General Relativity with *Mathematica*

Luigi E. Masciovecchio, M.Sc.
email: luigi.masciovecchio@virgilio.it

first published on http://mio.discoremoto.alice.it/luigimasciovecchio/, October 2011
available as notebook and PDF on
http://sites.google.com/site/luigimasciovecchio/

In[2]:=
```
Print["Revision ", IntegerPart[Date[]]]
```

```
Revision {2020, 9, 21, 8, 5, 3}
```

As soon as Wile E. Coyote has regained a more appropriate situational awareness,
he will begin to move on a geodesic (psycho-gravitational effect).
Fortunately, as usual, he will survive it without major problems.

Table of Content

General Relativity with Mathematica

Luigi E. Masciovecchio, M.Sc.
email: luigi.masciovecchio@virgilio.it
first published on http://mio.discoremoto.alice.it/luigimasciovecchio/, October 2011
available as notebook and PDF on http://sites.google.com/site/luigimasciovecchio/

A) INTRODUCTION

Dear Colleagues,

This is my personal *Mathematica* notebook on Albert Einstein's genial **general theory of relativity**. This document wasn't originally intended for publication, but a few formulas and tricks are maybe of interest to you, so here they are. The code seems to work well and I added some comments to make it more understandable. This is not an introduction to this field, so use it at your own risk!

The main point about this work is to show how to do the typical mathematics of general relativity easily and rigorously with *Mathematica*. In addition, I "streamlined" a little bit the derivation of some classical results (perihelion advance, bending of light etc.).

As main textbook I have chosen the excellent and brilliantly instructive "A short course in general relativity" by James Foster and J.David Nightingale. *Mathematica* together with the packages Tensorial and GeneralRelativity have been used by David Park to do all the derivations, examples and exercises of this textbook. Most of the present notebook is actually a rewrite of Park's very fine original work.

Once again, the combination of a good textbook and *Mathematica* provides a fun, easy and mathematical rigorous learning environment that stimulates greatly understanding and own experiments with the formulas. Don't miss it!

* * *

General relativity is a metric theory of gravitation. At its core are Einstein's field equations, which describe the relation between the geometry of a four-dimensional, pseudo-Riemannian manifold representing spacetime, and the energy-momentum contained in that spacetime. First published by Albert Einstein in 1915 as a tensor equation, the Einstein's field equations equate spacetime curvature (expressed by the Einstein tensor) with the energy and momentum within that spacetime (expressed by the energy-momentum-stress tensor). General relativity's predictions have been confirmed in all observations and experiments to date. Although general relativity is not the only relativistic theory of gravity, it is the simplest theory that is consistent with experimental data. (*Wikipedia*, 2011)

General relativity is a geometric theory and incorporates special relativity in the sense that locally the spacetime of the general theory is like that of the special theory. So it's important for the sake of conceptual cleanness to derive in your course first special relativity from the basic geometrical spacetime symmetries without using the postulate of constant speed of light or any other "unneeded physics" (see for example Jean-Marc Lévy-Leblond, "One more derivation of the Lorentz transformation", American Journal of Physics **44**, 271-277 (1976); visit http://o.castera.free.fr for more information).

Valuable web resources on general relativity:
- David Park, *Mathematica* notebooks (2005) based on "A short course in general relativity" (Foster/Nightingale)
- See books.google.com or the Springer editor web site for a preview of the above-mentioned textbook.
- Florian Schrack "Gravitation - Theorien, Effekte und Simulation am Computer" (2002)

• Gerard ’t Hooft "Introduction to General Relativity" (2007)
• Matt Visser "Math 464: Notes on Differential Geometry" (2009)
• Matt Visser "Math 465: Notes on General Relativity and Cosmology" (2009)
• Norbert Dragon "Geometrie der Relativitätstheorie" (2011)
• Sean Carroll "Lecture Notes on General Relativity" (1997)
• Tom Marsh "Notes for PX436, General Relativity" (2009)
• Clifford M. Will "The Confrontation between General Relativity and Experiment", Living Rev. Relativity, 9, (2006)
• Neil Ashby "Relativity in the Global Positioning System", Living Rev. Relativity, 6, (2003)
• Wikipedia: "General relativity", "Allgemeine Relativitätstheorie" and links
• General relativity video courses (Charles Bailyn, Alexander Maloney, Lenny Susskind)

Note:
• ***Mathematica*** by Wolfram Research is a (fabulous) computer algebra system.
• A **notebook** is an interactive *Mathematica* document (extension .nb).
• **Tensorial 3.0** (R. Cabrera, D. Park, J.-F. Gouyet, August 2005) is a general-purpose tensor calculus package for *Mathematica* Version 4.1 or later.
• **TGeneralRelativity1`GeneralRelativity`** (D. Park, 29 January 2005) is a subpackage for the Tensorial package that adds routines useful in special and general relativity. (This also automatically loads the regular Tensorial package.)

```
Print["This system is:"]
{"ProductIDName", "ProductVersion"} /. $ProductInformation
ReadList["!ver", String][[2]]
{$MachineType, $ProcessorType, $ByteOrdering, $SystemCharacterEncoding}
```

```
This system is:

{Mathematica, 5.2 for Microsoft Windows (June 20, 2005)}

Windows 98 [versione 4.10.1998]

{PC, x86, -1, WindowsANSI}
```

B) HELP

(Extracted from the Tensorial package help.)

■ **{x,δ,g,Γ}** are the standard set of tensor labels used in all Tensorial derivative routines. They tell the routines which labels will be considered to represent the coordinates x, Kronecker δ, metric tensor g and Christoffel symbol Γ.

■ **DeclareBaseIndices[{index..}]** declares the base indices for the underlying linear space.

■ **DeclareIndexFlavor[{flavorname,flavorform}...]** will add the index flavors to the IndexFlavors list and establish the Format for displaying indices with the given flavor name.

■ **ToArrayValues[baseindices][expr]** will convert the expression to a vector, matrix or array by expansion and substitution of any stored values.

■ **EvaluateDotProducts[e,g,metricsimplify:True][expr]** expands Dot products of vectors expressed in a given basis e using the metric tensor g. Metric simplification is performed if the default argument metricsimplify is True.

■ **LinearBreakout[f1,f2,...][v1,v2,...][expr]** will break out the linear terms of any expressions within expr that have heads matching the patterns fi over variables matching the patterns vj.

■ **SetMetricValues[g,metricmatrix,flavor:Identity]** creates value definitions for the up and down forms of the metric tensor using the label g and a metric matrix.

■ **CoordinateToTensors[{r,θ,ϕ...},coord:x,flavor:Identity][expr]** will convert the coordinate symbols in the expression to the corresponding indexed tensors. The optional arguments coord and flavor give the coordinate label and index flavor to use. Their default values are x and plain.

■ **SetChristoffelValueRules[xu[i,metricmatrix,Γ,simplification:Identity]** calculates and stores substitution rules for the Christoffel values of Γudd[i,j,k] and Γddd[i,j,k] from the values of metricmatrix and the xu[i] vector pattern.

■ **SelectedTensorRules[label,pattern]** will select the rules for label whose right hand sides are nonzero and whose left hand sides match the pattern.

■ **SimplifyTensorSum[expr]** will check that all terms in a tensor sum have valid indices,that the free indices are the same in all terms,and will simplify the sum by matching dummy indices in all terms that have the same index structure.

■ **ExpandCovariantD[{x,δ,g,Γ},a][expr]** will expand first order covariant derivatives of tensors using x as the label for the coordinates, δ as the label for the Kronecker, g as the label for the metric tensor and Γ as the label for Christoffel symbols. The introduced dummy index will be a.

■ **MapLevelParts[function,{topposition,levelpositions}][expr]** will map the function onto the selected level positions in an expression. The function is applied to them as a group and they are replaced with a single new expression. Other parts not specified on the list are left unchanged.

C) PHYSICAL CONSTANTS

Some physical constants as given by *Mathematica*.

```
Print["Miscellaneous`PhysicalConstants`:"]
<< Miscellaneous`PhysicalConstants`
<< Miscellaneous`Units`
{SpeedOfLight, GravitationalConstant, CosmicBackgroundTemperature, HubbleConstant}
{HubbleConstant^-1, AgeOfUniverse / HubbleConstant^-1, Convert[AgeOfUniverse, Year]}
{"Earth:", EarthMass, EarthRadius, "Sun:",
 Convert[SolarSchwarzschildRadius SpeedOfLight^2 / (2 GravitationalConstant), Kilogram],
 SolarRadius, SolarSchwarzschildRadius}
```

Miscellaneous`PhysicalConstants`:

$\left\{\frac{299\,792\,458\ \text{Meter}}{\text{Second}},\ \frac{6.673\times10^{-11}\ \text{Meter}^2\ \text{Newton}}{\text{Kilogram}^2},\ 2.726\ \text{Kelvin},\ \frac{3.2\times10^{-18}}{\text{Second}}\right\}$

$\{3.125\times10^{17}\ \text{Second},\ 1.504,\ 1.49036\times10^{10}\ \text{Year}\}$

$\{\text{Earth:},\ 5.9742\times10^{24}\ \text{Kilogram},\ 6\,378\,140\ \text{Meter},$
$\text{Sun:},\ 1.9888\times10^{30}\ \text{Kilogram},\ 6.9599\times10^{8}\ \text{Meter},\ 2953.25\ \text{Meter}\}$

```
Print["Gravitational constant\nG = ",
 Convert[GravitationalConstant, Kilogram^-1 Meter^3 Second^-2], " = ",
 Convert[GravitationalConstant, Gram^-1 Centimeter^3 Second^-2]]
Convert[8 π GravitationalConstant / SpeedOfLight^2, Meter / Kilogram];
Print[
 "Einsteinsche Gravitationskonstante in Sexl/Urbantke S.69\nκ = 8 π G c^-2 = ", %, " = ",
 Convert[%, Gram^-1 Centimeter ]]
Print["coupling constant in Foster/Nightingale p.113\nκ = - 8 π G c^-4 = ",
 - %% / SpeedOfLight^2]
```

Gravitational constant

$$G = \frac{6.673\times10^{-11}\ \text{Meter}^3}{\text{Kilogram Second}^2} = \frac{6.673\times10^{-8}\ \text{Centimeter}^3}{\text{Gram Second}^2}$$

Einsteinsche Gravitationskonstante in Sexl/Urbantke S.69

$$\kappa = 8\ \pi\ G\ c^{-2} = \frac{1.86603\times10^{-26}\ \text{Meter}}{\text{Kilogram}} = \frac{1.86603\times10^{-27}\ \text{Centimeter}}{\text{Gram}}$$

coupling constant in Foster/Nightingale p.113

$$\kappa = -\ 8\ \pi\ G\ c^{-4} = -\frac{2.07624\times10^{-43}\ \text{Second}^2}{\text{Kilogram Meter}}$$

I will use the CODATA 2010 values. (See http://physics.nist.gov/ for updates.)

```
Print["CODATA 2010: G = 6.673 84(80)×10^-11 m^3 kg^-1 s^-2"]
Print["κ = - 8 π G c^-4 = ",
 NumberForm[-8 π 6.67384 × 10^-11 / (299 792 458^4), 7], " m^-1 kg^-1 s^2"]
```

CODATA 2010: $G = 6.673\ 84(80)\times10^{-11}\ m^3\ kg^{-1}\ s^{-2}$

$\kappa = -\ 8\ \pi\ G\ c^{-4} = -2.076504\times10^{-43}\ m^{-1}\ kg^{-1}\ s^2$

D) OWN (?) CONSIDERATIONS

Special relativity teaches us how spacetime dictates the behavior of matter-energy and general relativity teaches us how matter-energy influences the behavior of spacetime. We could say that this two entities, spacetime and matter-energy, are in some kind of interaction. Starting from a heuristic principle that states that entities who can interact can not be *completely* different "in essence", we could tentatively postulate a symmetry between spacetime and matter-energy, implying the possibility of a transformation of spacetime into matter-energy and vice-versa. So it's maybe sensible to ask:

• How much spacetime can we get from a given quantity of matter-energy or vice-versa? What is the conversion factor λ between (geometrized) spacetime and matter-energy ($1\ m^4\ \hat{=}\ \lambda\ \cdot\ 1\ J$)? Is λ a universal constant?

• What are the observable signatures of spacetime ⇄ matter-energy transformations?

• How "expands" newly created spacetime in some finite region into the rest of the Universe? How works the local "collapse" of the universe caused by the destruction of a finite piece of spacetime?

• How works the spacetime - matter-energy - transformation at a fundamental level?

Since wild speculations don't cost a thing, we can go further and postulate the existence of a substance called "Essenz" in which matter-energy and spacetime are not separated entities and which constitutes all of the Universe at some point. We can assume that the Big Bang represents the moment of the evolution of the Universe where the Essenz undergoes a phase transition separating into the two components spacetime and matter-energy. Since then we have "matter-energy acting on the stage of spacetime" and we can define (at most locally) a metric to measure space and time. We said that the Essenz undergoes a phase transition at some point: this means that this substance is not static. But the time coordinate that we need to catalogue events in the pre-Big-Bang era has to be interpreted as an *intrinsic* parameter of the Essenz. This intrinsic time parameter must not necessarily be a measurable quantity (if there is no metric) but may defines only an order relation between events, the evolution of the Essenz proceeding by "leaps" much like today quantum systems evolves (e.g. successive decays in a radioactive series). Perhaps this analogy is not accidental and points to some connection between quantum mechanics and spacetime physics!

Well, as I said, wild speculations don't cost a thing...

E) CALCULATIONS FROM

James Foster, J.David Nightingale
A SHORT COURSE IN GENERAL RELATIVITY
(3.ed., 2006)

with *Mathematica* code by David Park (2005, for the 2. ed. [1995])
partially modified, corrected and simplified by Luigi E. Masciovecchio (2011)

Utilization note: Every of the following *Mathematica* subsections should be evaluated by its own! The initialization code for a subsection ends with a horizontal line. I present here only my limited set of calculations from the textbook, for a complete (!) and extensively commented set see the huge work by David Park.

Chapter 1: Vector and tensor fields

1.0 Introduction p. 7

FN: "Our starting point is a consideration of vector fields in the familiar setting of three-dimensional Euclidean space and how they can be handled using arbitrary curvilinear coordinate systems. We then go on to extend and generalize these ideas in two different ways, first by admitting tensor fields, and second by allowing the dimension of the space to be arbitrary and its geometry to be non-Euclidean."

1.1 Coordinate systems in Euclidean space p. 7 - 13 *(nonsuffix notation)*

Keywords: Cartesian and non-Cartesian coordinate systems, coordinate surfaces and curves, position vector of points in space, natural and dual basis in 3D Euclidean space.

Mathematica resources on coordinate systems (see the standard add-on package `Calculus`VectorAnalysis``)

```
<< Calculus`VectorAnalysis`
```

```
CoordinatesToCartesian[Coordinates[Spherical], Spherical]
{Coordinates[Spherical], CoordinateRanges[Spherical]}
CoordinatesToCartesian[{u, v, w}, Spherical]
CoordinatesFromCartesian[{x, y, z}, Spherical]
```

```
{Rr Cos[Pphi] Sin[Ttheta], Rr Sin[Pphi] Sin[Ttheta], Rr Cos[Ttheta]}
```

```
{{Rr, Ttheta, Pphi}, {0 ≤ Rr < ∞, 0 ≤ Ttheta ≤ π, -π < Pphi ≤ π}}
```

```
{u Cos[w] Sin[v], u Sin[v] Sin[w], u Cos[v]}
```

$$\left\{\sqrt{x^2+y^2+z^2},\ \text{ArcCos}\left[\frac{z}{\sqrt{x^2+y^2+z^2}}\right],\ \text{ArcTan}[x, y]\right\}$$

```
CoordinatesToCartesian[{u, v, w}, Paraboloidal]
```

$\left\{u\,v\,\text{Cos}[w],\ u\,v\,\text{Sin}[w],\ \frac{1}{2}\left(u^2-v^2\right)\right\}$

```
CoordinatesToCartesian[{u, v, w}, ParabolicCylindrical]
```

$\left\{\frac{1}{2}\left(u^2-v^2\right),\ u\,v,\ w\right\}$

■ Example 1.1.2, p.10. "spherical coordinates - natural basis/dual basis"

```
x[u_, v_, w_] := u Cos[w] Sin[v];
y[u_, v_, w_] := u Sin[v] Sin[w];
z[u_, v_, w_] := u Cos[v];
r = {x[u, v, w], y[u, v, w], z[u, v, w]};
(* inverted equations *)
uu[x_, y_, z_] := Sqrt[x^2 + y^2 + z^2];
vv[x_, y_, z_] := ArcCos[z / Sqrt[x^2 + y^2 + z^2]];
ww[x_, y_, z_] := ArcTan[x, y];
$Assumptions = {0 ≤ u < ∞, 0 ≤ v ≤ π, -π < w ≤ π, {x, y, z} ∈ Reals};
grad_xyz[scalarfield_] := {∂_x scalarfield, ∂_y scalarfield, ∂_z scalarfield}

Print["Position and natural basis {e_u,e_v,e_w} in spherical coordinates u,v,w"]
r // MatrixForm
naturalbasis_uvw = {∂_u r, ∂_v r, ∂_w r};
MatrixForm /@ naturalbasis_uvw
(naturalbasis_uvw.Transpose[naturalbasis_uvw]) // Simplify // MatrixForm
Print["{ê_u,ê_v,ê_w} in u,v,w"]
normnaturalbasis_uvw = naturalbasis_uvw / (Simplify[Sqrt[Abs[#.#]]] & /@ naturalbasis_uvw);
MatrixForm /@ normnaturalbasis_uvw
normnaturalbasis_uvw.Transpose[normnaturalbasis_uvw] // Simplify // MatrixForm
```

Position and natural basis $\{e_u,e_v,e_w\}$ in spherical coordinates u,v,w

$$\begin{pmatrix} u\,\text{Cos}[w]\,\text{Sin}[v] \\ u\,\text{Sin}[v]\,\text{Sin}[w] \\ u\,\text{Cos}[v] \end{pmatrix}$$

$$\left\{\begin{pmatrix} \text{Cos}[w]\,\text{Sin}[v] \\ \text{Sin}[v]\,\text{Sin}[w] \\ \text{Cos}[v] \end{pmatrix},\ \begin{pmatrix} u\,\text{Cos}[v]\,\text{Cos}[w] \\ u\,\text{Cos}[v]\,\text{Sin}[w] \\ -u\,\text{Sin}[v] \end{pmatrix},\ \begin{pmatrix} -u\,\text{Sin}[v]\,\text{Sin}[w] \\ u\,\text{Cos}[w]\,\text{Sin}[v] \\ 0 \end{pmatrix}\right\}$$

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & u^2 & 0 \\ 0 & 0 & u^2\,\text{Sin}[v]^2 \end{pmatrix}$$

$\{\hat{e}_u,\hat{e}_v,\hat{e}_w\}$ in u,v,w

$$\left\{\begin{pmatrix} \text{Cos}[w]\,\text{Sin}[v] \\ \text{Sin}[v]\,\text{Sin}[w] \\ \text{Cos}[v] \end{pmatrix},\ \begin{pmatrix} \text{Cos}[v]\,\text{Cos}[w] \\ \text{Cos}[v]\,\text{Sin}[w] \\ -\text{Sin}[v] \end{pmatrix},\ \begin{pmatrix} -\text{Sin}[w] \\ \text{Cos}[w] \\ 0 \end{pmatrix}\right\}$$

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

```
Print["Position and dual basis {e^u,e^v,e^w} in x,y,z"]
{uu[x, y, z], vv[x, y, z], ww[x, y, z]} // MatrixForm
dualbasis_xyz =
```

```
  {grad_xyz[uu[x, y, z]], grad_xyz[vv[x, y, z]], grad_xyz[ww[x, y, z]]} // Simplify;
MatrixForm /@ dualbasis_xyz
dualbasis_xyz.Transpose[dualbasis_xyz] // Simplify // MatrixForm
Print["{e^u,e^v,e^w} in u,v,w"]
dualbasis_uvw = dualbasis_xyz /. {x → x[u, v, w], y → y[u, v, w], z → z[u, v, w]};
MatrixForm /@ dualbasis_uvw // Simplify
dualbasis_uvw.Transpose[dualbasis_uvw] // Simplify // MatrixForm
Print["{ê^u,ê^v,ê^w} in u,v,w"]
normdualbasis_uvw = dualbasis_uvw / (Simplify[Sqrt[Abs[#.#]]] & /@ dualbasis_uvw);
normdualbasis_uvw = % // Simplify;
MatrixForm /@ %
%%%.Transpose[%%%] // Simplify // MatrixForm
```

Position and dual basis $\{e^u, e^v, e^w\}$ in x,y,z

$$\begin{pmatrix} \sqrt{x^2+y^2+z^2} \\ \text{ArcCos}\left[\frac{z}{\sqrt{x^2+y^2+z^2}}\right] \\ \text{ArcTan}[x, y] \end{pmatrix}$$

$$\left\{ \begin{pmatrix} \frac{x}{\sqrt{x^2+y^2+z^2}} \\ \frac{y}{\sqrt{x^2+y^2+z^2}} \\ \frac{z}{\sqrt{x^2+y^2+z^2}} \end{pmatrix}, \begin{pmatrix} \frac{x z}{\sqrt{x^2+y^2}\,(x^2+y^2+z^2)} \\ \frac{y z}{\sqrt{x^2+y^2}\,(x^2+y^2+z^2)} \\ -\frac{\sqrt{x^2+y^2}}{x^2+y^2+z^2} \end{pmatrix}, \begin{pmatrix} -\frac{y}{x^2+y^2} \\ \frac{x}{x^2+y^2} \\ 0 \end{pmatrix} \right\}$$

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{x^2+y^2+z^2} & 0 \\ 0 & 0 & \frac{1}{x^2+y^2} \end{pmatrix}$$

$\{e^u, e^v, e^w\}$ in u,v,w

$$\left\{ \begin{pmatrix} \text{Cos}[w]\,\text{Sin}[v] \\ \text{Sin}[v]\,\text{Sin}[w] \\ \text{Cos}[v] \end{pmatrix}, \begin{pmatrix} \frac{\text{Cos}[v]\,\text{Cos}[w]}{u} \\ \frac{\text{Cos}[v]\,\text{Sin}[w]}{u} \\ -\frac{1}{u\,\text{Abs}[\text{Csc}[v]]} \end{pmatrix}, \begin{pmatrix} -\frac{\text{Csc}[v]\,\text{Sin}[w]}{u} \\ \frac{\text{Cos}[w]\,\text{Csc}[v]}{u} \\ 0 \end{pmatrix} \right\}$$

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{u^2} & 0 \\ 0 & 0 & \frac{\text{Csc}[v]^2}{u^2} \end{pmatrix}$$

$\{\hat{e}^u, \hat{e}^v, \hat{e}^w\}$ in u,v,w

$$\left\{ \begin{pmatrix} \text{Cos}[w]\,\text{Sin}[v] \\ \text{Sin}[v]\,\text{Sin}[w] \\ \text{Cos}[v] \end{pmatrix}, \begin{pmatrix} \text{Cos}[v]\,\text{Cos}[w] \\ \text{Cos}[v]\,\text{Sin}[w] \\ -\frac{1}{\text{Abs}[\text{Csc}[v]]} \end{pmatrix}, \begin{pmatrix} -\frac{\text{Csc}[v]\,\text{Sin}[w]}{\text{Abs}[\text{Csc}[v]]} \\ \frac{\text{Cos}[w]\,\text{Csc}[v]}{\text{Abs}[\text{Csc}[v]]} \\ 0 \end{pmatrix} \right\}$$

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

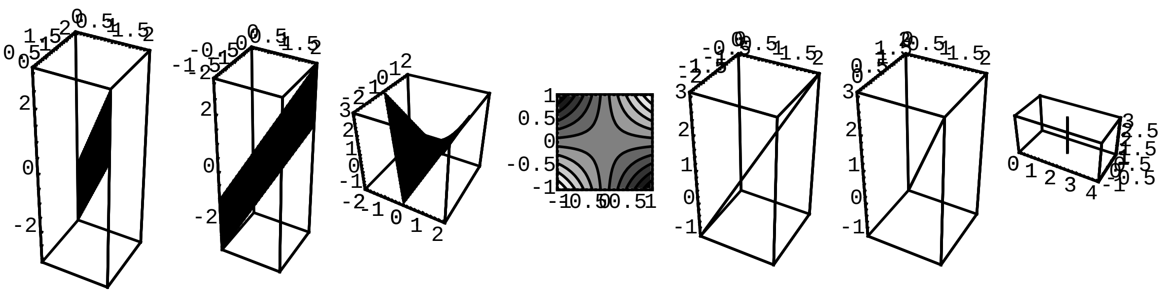
```
$Assumptions = {0 < u, 0 < v < π};
Print["{ê_u,ê_v,ê_w} in u,v,w = {ê^u,ê^v,ê^w} in u,v,w if ", %, " ?"]
Simplify /@ (normnaturalbasis_uvw == normdualbasis_uvw)
```

$\{\hat{e}_u, \hat{e}_v, \hat{e}_w\}$ in u,v,w = $\{\hat{e}^u, \hat{e}^v, \hat{e}^w\}$ in u,v,w if $\{0 < u, 0 < v < \pi\}$?

True

■ Example 1.1.3, p.11. "paraboloidal coordinates - natural basis"

```
x[u_, v_, w_] := u + v;
y[u_, v_, w_] := u - v;
z[u_, v_, w_] := 2 u v + w;
r = {x[u, v, w], y[u, v, w], z[u, v, w]};
{u0, v0, w0} = {1, 1, 1} (* working point *);

(* coordinate surfaces *)
CSu0 = ParametricPlot3D[{x[u0, v, w], y[u0, v, w], z[u0, v, w]},
   {v, -1, 1}, {w, -1, 1}, DisplayFunction → Identity];
CSv0 = ParametricPlot3D[{x[u, v0, w], y[u, v0, w], z[u, v0, w]},
   {u, -1, 1}, {w, -1, 1}, DisplayFunction → Identity];
CSw0 = ParametricPlot3D[{x[u, v, w0], y[u, v, w0], z[u, v, w0]},
   {u, -1, 1}, {v, -1, 1}, DisplayFunction → Identity];
CScontourzw0 = ContourPlot[z[u, v, w0], {u, -1, 1}, {v, -1, 1}, DisplayFunction → Identity];
(* coordinate lines *)
CLv0w0 = ParametricPlot3D[
   {x[u, v0, w0], y[u, v0, w0], z[u, v0, w0]}, {u, -1, 1}, DisplayFunction → Identity];
CLu0w0 = ParametricPlot3D[{x[u0, v, w0], y[u0, v, w0], z[u0, v, w0]},
   {v, -1, 1}, DisplayFunction → Identity];
CLu0v0 = ParametricPlot3D[{x[u0, v0, w], y[u0, v0, w], z[u0, v0, w]},
   {w, -1, 1}, DisplayFunction → Identity];
Show[GraphicsArray[{{CSu0, CSv0, CSw0, CScontourzw0, CLv0w0, CLu0w0, CLu0v0}}],
  ImageSize → 72 × 8];
```



⋆ Coordinate surfaces, contour plot z[u, v, w0], coordinate lines. ⋆

```
naturalbasis = {eu, ev, ew} = {∂u r, ∂v r, ∂w r};
Print["Position and natural basis {eu,ev,ew} in paraboloidal coordinates u,v,w"]
r
naturalbasis
(naturalbasis.Transpose[naturalbasis]) // Simplify // MatrixForm
Norm /@ naturalbasis // FullSimplify;
norms = Sqrt /@ Abs /@ {eu.eu, ev.ev, ew.ew} // Simplify
%% == %
normnaturalbasis = {êu, êv, êw} = naturalbasis / norms;
normnaturalbasis
normnaturalbasis.Transpose[normnaturalbasis] // Simplify // MatrixForm
```

Position and natural basis $\{e_u, e_v, e_w\}$ in paraboloidal coordinates u,v,w

{u + v, u - v, 2 u v + w}

{{1, 1, 2 v}, {1, -1, 2 u}, {0, 0, 1}}

$$\begin{pmatrix} 2+4v^2 & 4uv & 2v \\ 4uv & 2+4u^2 & 2u \\ 2v & 2u & 1 \end{pmatrix}$$

$$\left\{\sqrt{2+4v^2},\ \sqrt{2+4u^2},\ 1\right\}$$

True

$$\left\{\left\{\frac{1}{\sqrt{2+4v^2}}, \frac{1}{\sqrt{2+4v^2}}, \frac{2v}{\sqrt{2+4v^2}}\right\}, \left\{\frac{1}{\sqrt{2+4u^2}}, -\frac{1}{\sqrt{2+4u^2}}, \frac{2u}{\sqrt{2+4u^2}}\right\}, \{0, 0, 1\}\right\}$$

$$\begin{pmatrix} 1 & \frac{2uv}{\sqrt{(1+2u^2)(1+2v^2)}} & \frac{2v}{\sqrt{2+4v^2}} \\ \frac{2uv}{\sqrt{(1+2u^2)(1+2v^2)}} & 1 & \frac{2u}{\sqrt{2+4u^2}} \\ \frac{2v}{\sqrt{2+4v^2}} & \frac{2u}{\sqrt{2+4u^2}} & 1 \end{pmatrix}$$

Exercise 1.1.2 c), p.12-13 "paraboloidal coordinates (- dual basis)"

Evaluate first Example 1.1.3!

```
{e_u, e_v, e_w}
{1, 0, 0} == HoldForm[e_u / 2 + e_v / 2 - (u + v) e_w]
% // ReleaseHold
```

{{1, 1, 2 v}, {1, -1, 2 u}, {0, 0, 1}}

$$\{1, 0, 0\} == \frac{1}{2}\{1, 1, 2v\} + \frac{1}{2}\{1, -1, 2u\} - (u+v)\{0, 0, 1\}$$

True

Addendum

```
Print["Inverted equations"]
r == {x, y, z}
Solve[%, {u, v, w}];
{u, v, w} = {u, v, w} /. %[[1]]
grad_xyz[scalarfield_] := {∂_x scalarfield, ∂_y scalarfield, ∂_z scalarfield}
Print["Dual basis {e^u,e^v,e^w} in x,y,z and in paraboloidal coordinates u,v,w"]
dualbasis_xyz = {eu_xyz, ev_xyz, ew_xyz} = {grad_xyz[u], grad_xyz[v], grad_xyz[w]};
{dualbasis_xyz,
 dualbasis_xyz.Transpose[dualbasis_xyz] // Simplify // MatrixForm}
Clear[u, v, w]
dualbasis_uvw = (* {e^u,e^v,e^w}= *)
  {eu_uvw, ev_uvw, ew_uvw} = dualbasis_xyz /. {x → x[u, v, w], y → y[u, v, w], z → z[u, v, w]};
{dualbasis_uvw,
 dualbasis_uvw.Transpose[dualbasis_uvw] // Simplify // MatrixForm}
```

Inverted equations

$$\{u+v, u-v, 2uv+w\} == \{x, y, z\}$$

$$\left\{\frac{x+y}{2}, \frac{x-y}{2}, \frac{1}{2}\left(-x^2+y^2+2z\right)\right\}$$

Dual basis $\{e^u, e^v, e^w\}$ in x,y,z and in paraboloidal coordinates u,v,w

$$\left\{\left\{\left\{\frac{1}{2}, \frac{1}{2}, 0\right\}, \left\{\frac{1}{2}, -\frac{1}{2}, 0\right\}, \{-x, y, 1\}\right\}, \begin{pmatrix} \frac{1}{2} & 0 & \frac{1}{2}(-x+y) \\ 0 & \frac{1}{2} & \frac{1}{2}(-x-y) \\ \frac{1}{2}(-x+y) & \frac{1}{2}(-x-y) & 1+x^2+y^2 \end{pmatrix}\right\}$$

$$\left\{\left\{\left\{\frac{1}{2}, \frac{1}{2}, 0\right\}, \left\{\frac{1}{2}, -\frac{1}{2}, 0\right\}, \{-u-v, u-v, 1\}\right\}, \begin{pmatrix} \frac{1}{2} & 0 & -v \\ 0 & \frac{1}{2} & -u \\ -v & -u & 1+2u^2+2v^2 \end{pmatrix}\right\}$$

1.2 Suffix notation p. 13 - 19

FN: "The suffix notation provides a way of handling collections of related quantities that otherwise might be represented by arrays. The coordinates of a point constitute such a collection, as do the components of a vector, and the vectors on a basis. The basic idea is to represent the members of such a collection by means of a kernel letter to which is attached a literal suffix (or suffixes) representing numbers that serve to label the quantities in the collection."

```
Needs["TensorCalculus3`Tensorial`"]
$PrePrint =.
labs = {x, δ, g, Γ};
DeclareBaseIndices[{1, 2, 3}];
Print["Space dimension: ", NDim]
DefineTensorShortcuts[
 {{x, e}, 1},
 {{δ}, 2}]
SetTensorValues[δud[i, j], IdentityMatrix[NDim]]
δud[i, j] == (ToArrayValues[][δud[i, j]] // MatrixForm)
MyRed = StyleForm[Superscript[#, "′"], FontColor → RGBColor[1, 0, 0]] &;
DeclareIndexFlavor[{red, MyRed}]
```

Space dimension: 3

$$\delta^{i}{}_{j} == \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

```
SetTensorValueRules[xu[i], {x[u, v, w], y[u, v, w], z[u, v, w]}]
{xu[i], ToArrayValues[][xu[i]]}
{ed[i], ToArrayValues[][ed[i]]}
{xu[i] ed[i], EinsteinSum[][xu[i] ed[i]], ToArrayValues[][xu[i] ed[i]]}
(%[[2]] /. TensorValueRules[x]) == %[[3]]
```

$\{x^i, \{x[u, v, w], y[u, v, w], z[u, v, w]\}\}$

$\{e_i, \{e_1, e_2, e_3\}\}$

$\{e_i x^i, e_1 x^1 + e_2 x^2 + e_3 x^3, e_1 x[u, v, w] + e_2 y[u, v, w] + e_3 z[u, v, w]\}$

True

```
SetTensorValueRules[ed[i], IdentityMatrix[NDim]]
TensorValueRules[e]
{xu[i] ed[i], EinsteinSum[][xu[i] ed[i]], ToArrayValues[][xu[i] ed[i]]}
```

$\{e_1 \to \{1, 0, 0\}, e_2 \to \{0, 1, 0\}, e_3 \to \{0, 0, 1\}\}$

$\{e_i\, x^i,\ e_1\, x^1 + e_2\, x^2 + e_3\, x^3,\ \{x[u, v, w], y[u, v, w], z[u, v, w]\}\}$

```
a = xu[i] ed[i] // ToArrayValues[];
MatrixForm /@
 {"P", a /. {u → u0, v → v0, w → w0}, "CL", a /. {v → v0, w → w0}, a /. {u → u0, w → w0},
  a /. {u → u0, v → v0}, "CS",
  a /. u → u0, a /. v → v0, a /. w → w0, "s", a}
```

$$\left\{P,\ \begin{pmatrix} x[u_0, v_0, w_0] \\ y[u_0, v_0, w_0] \\ z[u_0, v_0, w_0] \end{pmatrix},\ CL,\ \begin{pmatrix} x[u, v_0, w_0] \\ y[u, v_0, w_0] \\ z[u, v_0, w_0] \end{pmatrix},\ \begin{pmatrix} x[u_0, v, w_0] \\ y[u_0, v, w_0] \\ z[u_0, v, w_0] \end{pmatrix},\ \begin{pmatrix} x[u_0, v_0, w] \\ y[u_0, v_0, w] \\ z[u_0, v_0, w] \end{pmatrix},\ CS,\ \begin{pmatrix} x[u_0, v, w] \\ y[u_0, v, w] \\ z[u_0, v, w] \end{pmatrix},\ \begin{pmatrix} x[u, v_0, w] \\ y[u, v_0, w] \\ z[u, v_0, w] \end{pmatrix},\ \begin{pmatrix} x[u, v, w_0] \\ y[u, v, w_0] \\ z[u, v, w_0] \end{pmatrix},\ s,\ \begin{pmatrix} x[u, v, w] \\ y[u, v, w] \\ z[u, v, w] \end{pmatrix}\right\}$$

```
SetTensorValueRules[xu[red@i], {u, v, w}]
{xu[i], xu[red@j]}
ToArrayValues[] /@ %
a = PartialD[xu[j], red@i];
b = a // ExpandPartialD[labs];
{a, b, b // TraditionalForm, MatrixForm /@ (b // ToArrayValues[])}
```

$\{x^i, x^{j'}\}$

$\{\{x[u, v, w], y[u, v, w], z[u, v, w]\}, \{u, v, w\}\}$

$$\left\{x^j{}_{,i'},\ \partial_{x^{i'}} x^j,\ \frac{\partial x^j}{\partial x^{i'}},\ \left\{\begin{pmatrix} x^{(1,0,0)}[u, v, w] \\ y^{(1,0,0)}[u, v, w] \\ z^{(1,0,0)}[u, v, w] \end{pmatrix},\ \begin{pmatrix} x^{(0,1,0)}[u, v, w] \\ y^{(0,1,0)}[u, v, w] \\ z^{(0,1,0)}[u, v, w] \end{pmatrix},\ \begin{pmatrix} x^{(0,0,1)}[u, v, w] \\ y^{(0,0,1)}[u, v, w] \\ z^{(0,0,1)}[u, v, w] \end{pmatrix}\right\}\right\}$$

1.3 Tangents and gradients p. 19 - 23

Keywords: tangent vector to a curve, length of a curve, line element $(ds)^2$ in general coordinates, partial differential operator ∂_i or $_{,i}$.

Sometimes it is more natural to work with the natural basis (example: tangents to curves) and sometimes it is more natural to work with the dual basis (example: gradients of scalar fields).

```
<< Calculus`VectorAnalysis`
```

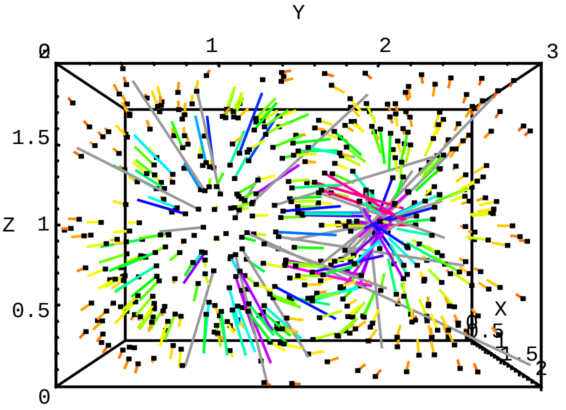
$$\text{ScalarField}[x\_, y\_, z\_] = -\frac{+1}{\sqrt{(x-1)^2 + (y-1)^2 + (z-1)^2}} - \frac{-1}{\sqrt{(x-1)^2 + (y-2)^2 + (z-1)^2}};$$

```
gSF[x_, y_, z_] = Grad[ScalarField[x, y, z], Cartesian[x, y, z]];
% // MatrixForm
```

$$\begin{pmatrix} -\frac{-1+x}{\left((-1+x)^2+(-2+y)^2+(-1+z)^2\right)^{3/2}} + \frac{-1+x}{\left((-1+x)^2+(-1+y)^2+(-1+z)^2\right)^{3/2}} \\ -\frac{-2+y}{\left((-1+x)^2+(-2+y)^2+(-1+z)^2\right)^{3/2}} + \frac{-1+y}{\left((-1+x)^2+(-1+y)^2+(-1+z)^2\right)^{3/2}} \\ -\frac{-1+z}{\left((-1+x)^2+(-2+y)^2+(-1+z)^2\right)^{3/2}} + \frac{-1+z}{\left((-1+x)^2+(-1+y)^2+(-1+z)^2\right)^{3/2}} \end{pmatrix}$$

```
nmax = 500; s = .15;
rp = Table[2 {Random[], 3 / 2 Random[], Random[]}, {n, 1, nmax}];
gSFp = gSF[Sequence @@ #1] & /@ rp;
Table[Graphics3D[
    {Point[rp[[n]]],
```

```
        {If[(a = Norm[ gSFp[[n]]]) < (b = 5), Hue[a / b], GrayLevel[.6]],
         Line[{rp[[n]], rp[[n]] + s gSFp[[n]]}]}}],
      {n, 1, nmax}];;
   Show[%, Axes → True, AxesLabel → {"X", "Y", "Z"},
     PlotRange → 2 {{0, 1}, {0, 3 / 2}, {0, 1}}, ViewPoint → {2, 0, 0}, ImageSize → 72 × 4];
```



⋆ Gradient of the 3D scalar field at random points. ⋆

Exercise 1.3.3, p.23.

Show that if the arc-length s (measured along a curve from some base point) is used as a parameter, then at each point of the curve given by $\xi(s)$ the tangent vector $d\xi(s)/ds$ has unit length.

Solution by David Park:

$$(ds)^2 = g_{ij}\, dx^i\, dx^j, \quad 1 = g_{ij} \frac{dx^i}{ds} \frac{dx^j}{ds};$$

$$\xi = e_i\, x^i, \quad \frac{d\xi}{ds} = e_i \frac{dx^i}{ds}$$

$$\left|\frac{d\xi}{ds}\right|^2 = \left|\left(e_i \frac{dx^i}{ds}\right) . \left(e_j \frac{dx^j}{ds}\right)\right|$$

$$\left|\frac{d\xi}{ds}\right|^2 = \left|g_{ij} \frac{dx^i}{ds} \frac{dx^j}{ds}\right|$$

$$\left|\frac{d\xi}{ds}\right|^2 = 1, \quad \left|\frac{d\xi}{ds}\right| = 1 \qquad q.e.d.$$

1.4 Coordinate transformations in Euclidean space p. 23 - 27

FN: "The purpose of this section is to explain how such things as the components of vectors relative to the bases defined by the coordinate systems transform, when we pass from the unprimed to the primed coordinate system (or *vice versa*). To this end, we shall use the suffix notation [...]."

Be careful:

```
{a √b , √(a² b)} /. {a → -1, b → +1}
{Dt[s²], (Dt[s])², Dt[s]², d s², ds²} // TraditionalForm
```

{-1, 1}

$\{2\, s\, d s, (d s)^2, (d s)^2, d\, s^2, \mathrm{ds}^2\}$

Example 1.4.1, p.25.

***Mathematica*:**

spherical coordinates: {Rr Cos[Pphi] Sin[Ttheta], Rr Sin[Pphi] Sin[Ttheta], Rr Cos[Ttheta]}, {{Rr, Ttheta, Pphi}, {0 ≤ Rr < ∞, 0 ≤ Ttheta ≤ π, -π < Pphi ≤ π}};

cylindrical coordinates: {Rr Cos[Ttheta], Rr Sin[Ttheta], Zz}, {{Rr, Ttheta, Zz}, {0 ≤ Rr < ∞, -π < Ttheta ≤ π, -∞ < Zz < ∞}}.

```
{x, y, z} ==
  {r Sin[θ] Cos[ϕ], r Sin[θ] Sin[ϕ], r Cos[θ]} == {ρ Cos[ϕ], ρ Sin[ϕ], z};
Drop[%, 1] // Thread
```

{r Cos[ϕ] Sin[θ] == ρ Cos[ϕ], r Sin[θ] Sin[ϕ] == ρ Sin[ϕ], r Cos[θ] == z}

Transformation between spherical coordinates and cylindrical coordinates :

$$\text{Hin} = \{r \to \sqrt{\rho^2 + z^2}, \theta \to \text{ArcTan}[\rho/z], \phi \to \phi\};$$
$$\text{Her} = \{\rho \to r\, \text{Sin}[\theta], \phi \to \phi, z \to r\, \text{Cos}[\theta]\};$$

```
{{HerVar, HerTrans}, {HinVar, HinTrans}} = {Table[(# /. Rule → List)[[i, 1]], {i, 3}],
    Table[(# /. Rule → List)[[i, 2]], {i, 3}]} & /@ {Hin, Her}
Us = Outer[D[#1, #2] &, HinTrans, HerVar, 1];
Uc = Simplify[Us /. Hin, {z ≥ 0}];
Ûc = Outer[D[#1, #2] &, HerTrans, HinVar, 1] // Simplify;
Ûs = Simplify[Uc /. Her, r ≥ 0];
MatrixForm /@ {Us, Uc, Ûc, Ûs}
MatrixForm /@ {Us.Ûs, Uc.Ûc} // Simplify
```

$$\{\{\{r, \theta, \phi\}, \{\sqrt{z^2+\rho^2}, \text{ArcTan}[\tfrac{\rho}{z}], \phi\}\}, \{\{\rho, \phi, z\}, \{r\,\text{Sin}[\theta], \phi, r\,\text{Cos}[\theta]\}\}\}$$

$$\left\{\begin{pmatrix} \text{Sin}[\theta] & r\,\text{Cos}[\theta] & 0 \\ 0 & 0 & 1 \\ \text{Cos}[\theta] & -r\,\text{Sin}[\theta] & 0 \end{pmatrix}, \begin{pmatrix} \frac{\rho}{\sqrt{z^2+\rho^2}} & z & 0 \\ 0 & 0 & 1 \\ \frac{z}{\sqrt{z^2+\rho^2}} & -\rho & 0 \end{pmatrix}, \begin{pmatrix} \frac{\rho}{\sqrt{z^2+\rho^2}} & 0 & \frac{z}{\sqrt{z^2+\rho^2}} \\ \frac{z}{z^2+\rho^2} & 0 & -\frac{\rho}{z^2+\rho^2} \\ 0 & 1 & 0 \end{pmatrix}, \begin{pmatrix} \text{Sin}[\theta] & 0 & \text{Cos}[\theta] \\ \frac{\text{Cos}[\theta]}{r} & 0 & -\frac{\text{Sin}[\theta]}{r} \\ 0 & 1 & 0 \end{pmatrix}\right\}$$

$$\left\{\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}\right\}$$

From Exercise 1.1.2 the unit vector field $\boldsymbol{\lambda} = \mathbf{i}$ has the spherical contravariant components:

```
Lss = {Sin[θ] Cos[ϕ], Cos[θ] Cos[ϕ] / r, -Csc[θ] Sin[ϕ] / r};
Lsc = Simplify[Lss /. Hin, {z ≥ 0}];
Lcs = Us.Lss // Simplify;
Lcc = Simplify[Lcs /. Hin, {z ≥ 0}];
MatrixForm /@ {Lss, Lsc, Lcs, Lcc}
{Lsc == Ûc.Lcc, Lss == Simplify[Lsc /. Her, r ≥ 0],
 Lcc == Simplify[Uc.Lsc], Lss == Ûs.Lcs, Lcs == Lcc /. Her}
```

$$\left\{\begin{pmatrix}\cos[\phi]\sin[\theta]\\ \frac{\cos[\theta]\cos[\phi]}{r}\\ -\frac{\csc[\theta]\sin[\phi]}{r}\end{pmatrix},\ \begin{pmatrix}\frac{\rho\cos[\phi]}{\sqrt{z^2+\rho^2}}\\ \frac{z\cos[\phi]}{z^2+\rho^2}\\ -\frac{\sin[\phi]}{\rho}\end{pmatrix},\ \begin{pmatrix}\cos[\phi]\\ -\frac{\csc[\theta]\sin[\phi]}{r}\\ 0\end{pmatrix},\ \begin{pmatrix}\cos[\phi]\\ -\frac{\sin[\phi]}{\rho}\\ 0\end{pmatrix}\right\}$$

```
{True, True, True, True, True}
```

Exercise 1.4.3, p.27.

[...] Hence, using G from Example 1.3.1 and $\hat{U}$ from Example 1.4.1, obtain the line element for Euclidean space in cylindrical coordinates.

The contracted dummy indices have to be brought adjacent. But where they are not adjacent we must use a transpose.

```
Gss = DiagonalMatrix[{1, r^2, r^2 Sin[θ]^2}];
Gsc = Gss /. Hin // Simplify;
MatrixForm /@ {Gss, Gsc}
Dt[s]^2 == Dt[HerVar].Gss.Dt[HerVar] // TraditionalForm
Print["line element for Euclidean space in spherical coordinates"]
```

$$\left\{\begin{pmatrix}1&0&0\\0&r^2&0\\0&0&r^2\sin[\theta]^2\end{pmatrix},\ \begin{pmatrix}1&0&0\\0&z^2+\rho^2&0\\0&0&\rho^2\end{pmatrix}\right\}$$

$$(ds)^2 = (dr)^2 + r^2\,(d\theta)^2 + r^2\,(d\phi)^2\,\sin^2(\theta)$$

```
line element for Euclidean space in spherical coordinates
```

```
Gcs = Transpose[Ûs].Gss.Ûs // Simplify;
Gcc = Gcs /. Hin // Simplify;
MatrixForm /@ {Gcs, Gcc}
Dt[s]^2 == Dt[HinVar].Gcc.Dt[HinVar] // TraditionalForm
Print["line element for Euclidean space in cylindrical coordinates"]
```

$$\left\{\begin{pmatrix}1&0&0\\0&r^2\sin[\theta]^2&0\\0&0&1\end{pmatrix},\ \begin{pmatrix}1&0&0\\0&\rho^2&0\\0&0&1\end{pmatrix}\right\}$$

$$(ds)^2 = (dz)^2 + (d\rho)^2 + \rho^2\,(d\phi)^2$$

```
line element for Euclidean space in cylindrical coordinates
```

1.5 Tensor fields in Euclidean space p. 27 - 30

FN: "While scalar and vector fields are sufficient to formulate Newton's theory of gravitation, tensor fields are an additional requirement for Einstein's theory."

Note: Often the label Λ instead of the Foster and Nightingale U is used to represent the transformation matrices.

The general rule for transforming tensors is just to use a transformation matrix Λ (Jacobian matrix) for each index in the tensor. The Λ matrix always has the *ud* configuration and has indices of different flavors. The dummy indices must be matched to the old indices and the other indices must correspond to the new free indices.

Example: $\tau^{i'}{}_{j'} == \Lambda^{l}{}_{j'} \Lambda^{i'}{}_{k} \tau^{k}{}_{l}$

```
Needs["TGeneralRelativity1`GeneralRelativity`"]
$PrePrint =.
DeclareBaseIndices[{1, 2, 3}]
DefineTensorShortcuts[
 {{e, f, n}, 1},
 {{g, Λ, τ}, 2}]
MyRed = StyleForm[Superscript[#, "′"], FontColor → RGBColor[1, 0, 0]] &;
DeclareIndexFlavor[{red, MyRed}]
```

The force-stress-relation $\vec{f} = \tau(\vec{n})$ in component form:

```
Print["Vector representation of forces"]
fu[i] ed[i] == τ[nu[j] ed[j]]
Print["τ is linear on the basis vectors"]
%% // LinearBreakout[τ][ed[_]]
Print["Expand τ on the basis vectors"]
%% /. τ[ed[j]] → τud[i, j] ed[i]
Print["We obtain the force components:"]
(# / ed[i] & /@ %%) // FrameBox // DisplayForm
```

Vector representation of forces

$e_i f^i == \tau[e_j n^j]$

τ is linear on the basis vectors

$e_i f^i == n^j \tau[e_j]$

Expand τ on the basis vectors

$e_i f^i == e_i n^j \tau^i{}_j$

We obtain the force components:

$f^i == n^j \tau^i{}_j$

Exercise 1.5.1, p.30.

Show that the components $\tau^i{}_j$ of the stress tensor τ are given by $\tau^i{}_j == e^i.\tau(e_j)$ and use this result to re-establish the transformation formula (1.58) for the components.

```
(step1 = τud[i, j] == eu[i].τ[ed[j]]) // FrameBox // DisplayForm
Print["Expand τ"]
step1 /. τ[ed[j]] → τud[k, j] ed[k]
Print["Linearity of dot product"]
%% // LinearBreakout[Dot][ed[_]]
Print["Basis, dual basis relation and using g as a Kronecker"]
%% /. BasisDotProductRules[e, g]
% // KroneckerAbsorb[g]
```

$\boxed{\tau^i{}_j == e^i.\tau[e_j]}$

Expand τ

$\tau^i{}_j == e^i.(e_k\,\tau^k{}_j)$

Linearity of dot product

$\tau^i{}_j == e^i.e_k\,\tau^k{}_j$

Basis, dual basis relation and using g as a Kronecker

$\tau^i{}_j == g^i{}_k\,\tau^k{}_j$

True

We can now use this to establish the transformation relation:

```
Print["We turn step1 into a rule."]
rule1 = Rule @@ Reverse@step1 // LHSSymbolsToPatterns[{i, j}]
step1 // ToFlavor[red]
Print["Express red basis vectors in terms of plain coordinates"]
%% /. eu[red@i] → Λud[red@i, k] eu[k] /. ed[red@j] → Λud[l, red@j] ed[l]
Print["Use linearity of τ and dot product"]
%% // LinearBreakout[Dot, τ][ed[_], eu[_], τ[_]]
Print["Use previous relation to substitute τ components"]
%% /. rule1
```

We turn step1 into a rule.

$e^{i\_}.\tau[e_{j\_}] \to \tau^i{}_j$

$\tau^{i'}{}_{j'} == e^{i'}.\tau[e_{j'}]$

Express red basis vectors in terms of plain coordinates

$\tau^{i'}{}_{j'} == (e^k\,\Lambda^{i'}{}_k).\tau[e_l\,\Lambda^l{}_{j'}]$

Use linearity of τ and dot product

$\tau^{i'}{}_{j'} == e^k.\tau[e_l]\,\Lambda^l{}_{j'}\,\Lambda^{i'}{}_k$

Use previous relation to substitute τ components

$\tau^{i'}{}_{j'} == \Lambda^l{}_{j'}\,\Lambda^{i'}{}_k\,\tau^k{}_l$

This is the desired transformation relation.

1.6 Surfaces in Euclidean space p. 30 - 35

The Potatoid Project

- A project for doing geometry on potatoids -

It's nice to play around with all the new geometrical concepts (coordinate transformations, natural basis, metric, dual basis, geodetics, parallel transport, etc.) in a nontrivial context where a complete visual representation is still possible. Sufficiently smooth and well-behaved deformations of a spherical surface (2D) embedded in a regular 3D Euclidean space (which I call *potatoids*) provide such a "geometrical playground". The basic coordinate system (ϕ, θ) on potatoids is borrowed from spherical coordinates on the sphere.

Spherical coordinates in Mathematica
{Rr Cos[Pphi] Sin[Ttheta], Rr Sin[Pphi] Sin[Ttheta], Rr Cos[Ttheta]}, {{Rr, Ttheta, Pphi}, {0 ≤ Rr < ∞ 0 ≤ Ttheta ≤ π, -π < Pphi ≤ π}}.

```
$Assumptions = -π < ϕ ≤ π && 0 ≤ θ ≤ π;
rϕ = {ϕ, -π, π}; rθ = {θ, 0, π};
surface[ϕ_, θ_] = {
   Rx[ϕ, θ] Cos[ϕ] Sin[θ],
   Ry[ϕ, θ] Sin[ϕ] Sin[θ],
   Rz[ϕ, θ] Cos[θ]};
naturalBasis = {nb1[ϕ_, θ_], nb2[ϕ_, θ_]} = {D[surface[ϕ, θ], ϕ], D[surface[ϕ, θ], θ]};
(*'down' metric *)
 gd[ϕ_, θ_] = naturalBasis.Transpose[naturalBasis];
(*'up' metric *)
gu[ϕ_, θ_] = Inverse[gd[ϕ, θ]];
dualBasis = {db1[ϕ_, θ_], db2[ϕ_, θ_]} = gu[ϕ, θ].naturalBasis;
angleBetweenNaturalBasisVectors[ϕ_, θ_] =
  ArcCos[gd[ϕ, θ][[1,2]] / (Norm[nb1[ϕ, θ]] Norm[nb2[ϕ, θ]])] / Pi * 180;
(* a path on the surface parametrized by t *)
path[t_] = surface[fϕ[t], fθ[t]];
```

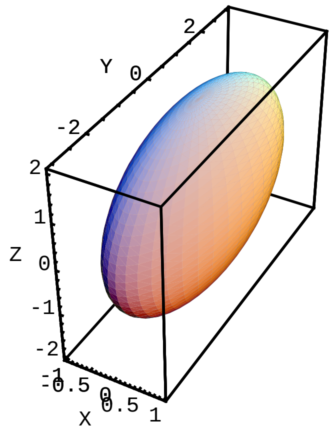
$$\text{velocity}[t\_] = \sqrt{\begin{array}{l}\text{gd}[f\phi[t], f\theta[t]]_{[[1,1]]}\ f\phi'[t]^2 + \\ 2\ \text{gd}[f\phi[t], f\theta[t]]_{[[1,2]]}\ f\phi'[t]\ f\theta'[t] + \\ \text{gd}[f\phi[t], f\theta[t]]_{[[2,2]]}\ f\theta'[t]^2\end{array}};$$

$$\text{length}[ti\_, tf\_] := \int_{ti}^{tf} \text{velocity}[t]\, dt;$$

```
Nlength[ti_, tf_] := NIntegrate[velocity[t], {t, ti, tf}];
```

```
(* A little collection of potatoids: *)

msg = "sphere";
pϕ[ϕ_] = pθ[θ_] = 0;
Rz[ϕ_, θ_] = Ry[ϕ_, θ_] = Rx[ϕ_, θ_] = 1 + pϕ[ϕ] pθ[θ];

msg = "shell potatoid";(* interesting, but NOT well behaved *)
pϕ[ϕ_] = (-π - ϕ)^2; pθ[θ_] = 1;
Rz[ϕ_, θ_] = Ry[ϕ_, θ_] = Rx[ϕ_, θ_] = 1 + pϕ[ϕ] pθ[θ];
```

```
msg = "X potatoid";
pϕ[ϕ_] = Sin[ϕ]^2; pθ[θ_] = Sin[2 θ]^2;
Rz[ϕ_, θ_] = Ry[ϕ_, θ_] = Rx[ϕ_, θ_] = 1 + pϕ[ϕ] pθ[θ];

msg = "cardio potatoid";
pϕ[ϕ_] = -(-π - ϕ)^2 (1/3 + 3 ϕ) (1/2 - ϕ) (π - ϕ)^2 / 550;
pθ[θ_] = θ^2 (2 - θ) (π - θ)^2 / 4;
Rz[ϕ_, θ_] = Ry[ϕ_, θ_] = Rx[ϕ_, θ_] = 1 + pϕ[ϕ] pθ[θ];

msg = "Ellipsoid"; (* gives reasonable short but nontrivial results *)
pϕ[ϕ_] = pθ[θ_] = Null;
Rx[ϕ_, θ_] = 1; Ry[ϕ_, θ_] = 3; Rz[ϕ_, θ_] = 2;

                                    (* --- *)

Potatoid = ParametricPlot3D[surface[ϕ, θ], Evaluate[rϕ],
   Evaluate[rθ], PlotPoints → 40, DisplayFunction → Identity];
Show[Graphics3D[EdgeForm[], Axes → True, ImageSize → {8 × 72, 3 × 72},
   AxesLabel → {"X", "Y", "Z"}, PlotLabel → msg], Potatoid];
```



★ Plot of the chosen surface. ★

```
ts[expr_] := TimeConstrained[FullSimplify[expr], 15, Print["(not simplified)"]; expr]
Print["Example: ", msg,
  " \nCartesian coordinates (x,y,z) in the embedding 3D space of a point
   with coordinates (ϕ,θ) on the chosen surface:"]
MatrixForm /@ ({x[ϕ, θ], y[ϕ, θ], z[ϕ, θ]} == ts[surface[ϕ, θ]])
Print["Natural basis vectors e_i = ∂_{x_i} f(x_j)"]
MatrixForm /@ ts[naturalBasis]
Print["Dual basis vectors e^i = g^ij e_j"]
MatrixForm /@ ts[dualBasis]
Print["Orthogonality and normalization e^i.e_j"]
ts[naturalBasis.Transpose[dualBasis]] // MatrixForm
Print["'Down' metric g_ij"]
ts[gd[ϕ, θ]] // MatrixForm
Print["'Up' metric g^ij"]
ts[gu[ϕ, θ]] // MatrixForm
Print["g_ij g^jk = δ_ik"]
ts[gd[ϕ, θ].gu[ϕ, θ]] // MatrixForm
```

```
Example: Ellipsoid
Cartesian coordinates (x,y,z) in the embedding
   3D space of a point with coordinates (ϕ,θ) on the chosen surface:
```

$$\begin{pmatrix} x[\phi,\theta] \\ y[\phi,\theta] \\ z[\phi,\theta] \end{pmatrix} == \begin{pmatrix} \mathrm{Cos}[\phi]\,\mathrm{Sin}[\theta] \\ 3\,\mathrm{Sin}[\theta]\,\mathrm{Sin}[\phi] \\ 2\,\mathrm{Cos}[\theta] \end{pmatrix}$$

Natural basis vectors $\underline{e}_i = \partial_{x_i} \underline{f}(x_j)$

$$\left\{ \begin{pmatrix} -\mathrm{Sin}[\theta]\,\mathrm{Sin}[\phi] \\ 3\,\mathrm{Cos}[\phi]\,\mathrm{Sin}[\theta] \\ 0 \end{pmatrix}, \begin{pmatrix} \mathrm{Cos}[\theta]\,\mathrm{Cos}[\phi] \\ 3\,\mathrm{Cos}[\theta]\,\mathrm{Sin}[\phi] \\ -2\,\mathrm{Sin}[\theta] \end{pmatrix} \right\}$$

Dual basis vectors $\underline{e}^i = g^{ij}\ \underline{e}_j$

$$\left\{ \begin{pmatrix} \frac{(13+5\,\mathrm{Cos}[2\theta])\,\mathrm{Csc}[\theta]\,\mathrm{Sin}[\phi]}{-29+11\,\mathrm{Cos}[2\theta]-32\,\mathrm{Cos}[2\phi]\,\mathrm{Sin}[\theta]^2} \\ \frac{3\,(-5+3\,\mathrm{Cos}[2\theta])\,\mathrm{Cos}[\phi]\,\mathrm{Csc}[\theta]}{-29+11\,\mathrm{Cos}[2\theta]-32\,\mathrm{Cos}[2\phi]\,\mathrm{Sin}[\theta]^2} \\ -\frac{16\,\mathrm{Cos}[\theta]\,\mathrm{Sin}[2\phi]}{-29+11\,\mathrm{Cos}[2\theta]-32\,\mathrm{Cos}[2\phi]\,\mathrm{Sin}[\theta]^2} \end{pmatrix}, \begin{pmatrix} \frac{18\,\mathrm{Cos}[\theta]\,\mathrm{Cos}[\phi]}{29-11\,\mathrm{Cos}[2\theta]+32\,\mathrm{Cos}[2\phi]\,\mathrm{Sin}[\theta]^2} \\ -\frac{6\,\mathrm{Cos}[\theta]\,\mathrm{Sin}[\phi]}{-29+11\,\mathrm{Cos}[2\theta]-32\,\mathrm{Cos}[2\phi]\,\mathrm{Sin}[\theta]^2} \\ -\frac{4\,(5+4\,\mathrm{Cos}[2\phi])\,\mathrm{Sin}[\theta]}{29-11\,\mathrm{Cos}[2\theta]+32\,\mathrm{Cos}[2\phi]\,\mathrm{Sin}[\theta]^2} \end{pmatrix} \right\}$$

Orthogonality and normalization $\underline{e}^i . \underline{e}_j$

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

'Down' metric g_{ij}

$$\begin{pmatrix} (5+4\,\mathrm{Cos}[2\phi])\,\mathrm{Sin}[\theta]^2 & 8\,\mathrm{Cos}[\theta]\,\mathrm{Cos}[\phi]\,\mathrm{Sin}[\theta]\,\mathrm{Sin}[\phi] \\ 8\,\mathrm{Cos}[\theta]\,\mathrm{Cos}[\phi]\,\mathrm{Sin}[\theta]\,\mathrm{Sin}[\phi] & \mathrm{Cos}[\theta]^2\,(5-4\,\mathrm{Cos}[2\phi])+4\,\mathrm{Sin}[\theta]^2 \end{pmatrix}$$

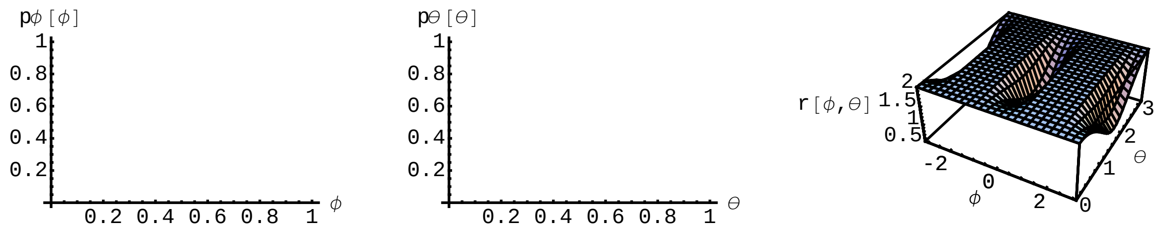
'Up' metric g^{ij}

$$\begin{pmatrix} \frac{8+(10-8\,\mathrm{Cos}[2\phi])\,\mathrm{Cot}[\theta]^2}{29-11\,\mathrm{Cos}[2\theta]+32\,\mathrm{Cos}[2\phi]\,\mathrm{Sin}[\theta]^2} & \frac{8\,\mathrm{Cot}[\theta]\,\mathrm{Sin}[2\phi]}{-29+11\,\mathrm{Cos}[2\theta]-32\,\mathrm{Cos}[2\phi]\,\mathrm{Sin}[\theta]^2} \\ \frac{8\,\mathrm{Cot}[\theta]\,\mathrm{Sin}[2\phi]}{-29+11\,\mathrm{Cos}[2\theta]-32\,\mathrm{Cos}[2\phi]\,\mathrm{Sin}[\theta]^2} & \frac{2\,(5+4\,\mathrm{Cos}[2\phi])}{29-11\,\mathrm{Cos}[2\theta]+32\,\mathrm{Cos}[2\phi]\,\mathrm{Sin}[\theta]^2} \end{pmatrix}$$

$g_{ij}\ g^{jk} = \delta_{ik}$

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

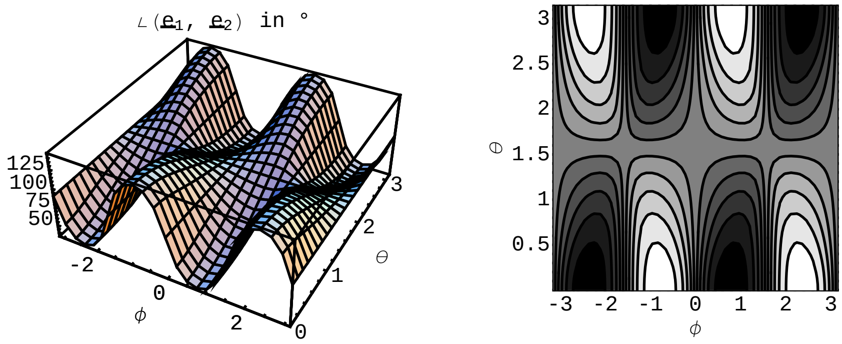
```
p1 = Plot[pϕ[ϕ], Evaluate[rϕ], AxesLabel → {"ϕ", "pϕ[ϕ]"},
    PlotStyle → Hue[0], DisplayFunction → Identity];
p2 = Plot[pθ[θ], Evaluate[rθ], AxesLabel → {"θ", "pθ[θ]"},
    PlotStyle → Hue[0], DisplayFunction → Identity];
p3 = Plot3D[Norm[surface[ϕ, θ]], Evaluate[rϕ], Evaluate[rθ],
    Mesh → True, PlotRange → {Rest[rϕ], Rest[rθ], {0.5, 2}},
    AxesLabel → {"ϕ", "θ", "r[ϕ,θ]   "}, DisplayFunction → Identity];
Show[GraphicsArray[{p1, p2, p3}], ImageSize → 72 × 8];
```



★ Functions pϕ(ϕ) and pθ(θ) if defined. Radial coordinate r(ϕ, θ) of the point (ϕ, θ). ★

```
ϵ = {0, 10^-6, -10^-6};
p1 = Plot3D[angleBetweenNaturalBasisVectors[ϕ, θ], Evaluate[rϕ + ϵ], Evaluate[rθ + ϵ],
    Mesh → True, PlotRange → {Rest[rϕ], Rest[rθ], Automatic}, AxesLabel → {"ϕ", "θ", ""},
    PlotLabel → "∠(e1, e2) in °", DisplayFunction → Identity];
p2 = ContourPlot[angleBetweenNaturalBasisVectors[ϕ, θ],
```

```
    Evaluate[rϕ + ϵ], Evaluate[rθ + ϵ], PlotRange → {Rest[rϕ], Rest[rθ], Automatic},
    PlotPoints → 50, FrameLabel → {"ϕ", "θ"}, DisplayFunction → Identity];
Show[GraphicsArray[{p1, p2}], ImageSize → 72 × 6];
```



★ Angle between natural basis vectors at point (ϕ, θ). ★

```
(* working point p *)
ϕp = 2.5; θp = 0.5 ;
p = surface[ϕp, θp] // N
```

```
{-0.384089, 0.860768, 1.75517}
```

```
(* natural basis and properties at point p *)
a = nb1[ϕp, θp] // N; na = Graphics3D[Line[{p, p + a}]];
b = nb2[ϕp, θp] // N; nb = Graphics3D[Line[{p, p + b}]];
{a // MatrixForm, b // MatrixForm, Norm[a], Norm[b], a.b,
 a.b / (Norm[a] Norm[b]), angleBetweenNaturalBasisVectors[ϕp, θp] "°"}
```

$$\left\{\begin{pmatrix}-0.286923\\-1.15227\\0.\end{pmatrix}, \begin{pmatrix}-0.70307\\1.57563\\-0.958851\end{pmatrix}, 1.18745, 1.9739, -1.61381, -0.688512, 133.512\,°\right\}$$

```
(* down metric, up metric and their product at point p *)
MatrixForm /@ N[{gd[ϕp, θp], gu[ϕp, θp], gd[ϕp, θp].gu[ϕp, θp]}]
```

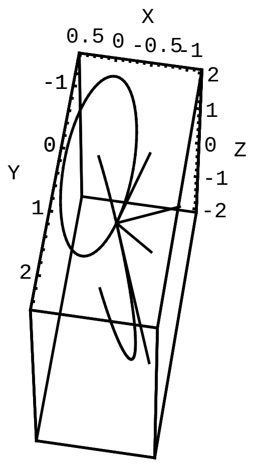
$$\left\{\begin{pmatrix}1.41004 & -1.61381\\-1.61381 & 3.8963\end{pmatrix}, \begin{pmatrix}1.34841 & 0.5585\\0.5585 & 0.48798\end{pmatrix}, \begin{pmatrix}1. & 3.17671\times10^{-17}\\9.21572\times10^{-17} & 1.\end{pmatrix}\right\}$$

```
(* dual basis and properties at point p *)
c = db1[ϕp, θp] // N; dc = Graphics3D[Line[{p, p + c}]];
d = db2[ϕp, θp] // N; dd = Graphics3D[Line[{p, p + d}]];
{c // MatrixForm, d // MatrixForm, Norm[c], Norm[d], c.d,
 c.d / (Norm[c] Norm[d]), (180 - ArcCos[c.d / (Norm[c] Norm[d])] / Pi * 180) "°"}
{{a.c, a.d}, {b.c, b.d}} // MatrixForm
```

$$\left\{\begin{pmatrix}-0.779554\\-0.67374\\-0.535518\end{pmatrix}, \begin{pmatrix}-0.50333\\0.125333\\-0.4679\end{pmatrix}, 1.16121, 0.698555, 0.5585, 0.688512, 133.512\,°\right\}$$

$$\begin{pmatrix}1. & -1.38791\times10^{-16}\\2.17491\times10^{-16} & 1.\end{pmatrix}$$

```
CLϕp = ParametricPlot3D[surface[ϕp, θ], Evaluate[rθ], DisplayFunction → Identity];
CLθp = ParametricPlot3D[surface[ϕ, θp], Evaluate[rϕ], DisplayFunction → Identity];
Show[CLϕp, CLθp, na, nb, dc, dd, DisplayFunction → $DisplayFunction,
  AxesLabel → {"X", "Y", "Z"}, ImageSize → {8 × 72, 3 × 72}, ViewPoint → 5 (p + {.2, .3, .5})];
```



★ Coordinate lines, natural basis and dual basis at point p. ★

```
(* Straight path from north pole to south pole, 0≤t≤1 *)
fϕ[t_] := π
fθ[t_] := π t
nr = 1;

(* Spiral path from north pole to south pole with nr spires, 0≤t≤1 *)
nr = 7;
fϕ[t_] := -π + Mod[2 π nr t, 2 π]
fθ[t_] := π t

(* Path length analytical/numerical *)
TimeConstrained[length[0, 1], 10]
TimeConstrained[Nlength[0, 1], 20]
```

$Aborted

75.2417

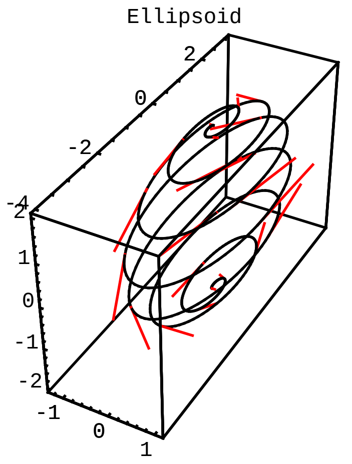
```
(* At a particular point of the path *)
pt = 0.4;
MatrixForm /@ {
  {ϕpt = fϕ[pt], θpt = fθ[pt]},
  {apt = fϕ'[pt], bpt = fθ'[pt]}, apt nb1[ϕpt, θpt] + bpt nb2[ϕpt, θpt]}
MatrixForm /@ {path[pt], path'[pt]}
%%[[3]] == %[[2]]
```

$$\left\{\begin{pmatrix}1.88496\\1.25664\end{pmatrix}, \begin{pmatrix}43.9823\\\pi\end{pmatrix}, \begin{pmatrix}-40.0824\\-36.0083\\-5.97566\end{pmatrix}\right\}$$

$$\left\{\begin{pmatrix}-0.293893\\2.71353\\0.618034\end{pmatrix}, \begin{pmatrix}-40.0824\\-36.0083\\-5.97566\end{pmatrix}\right\}$$

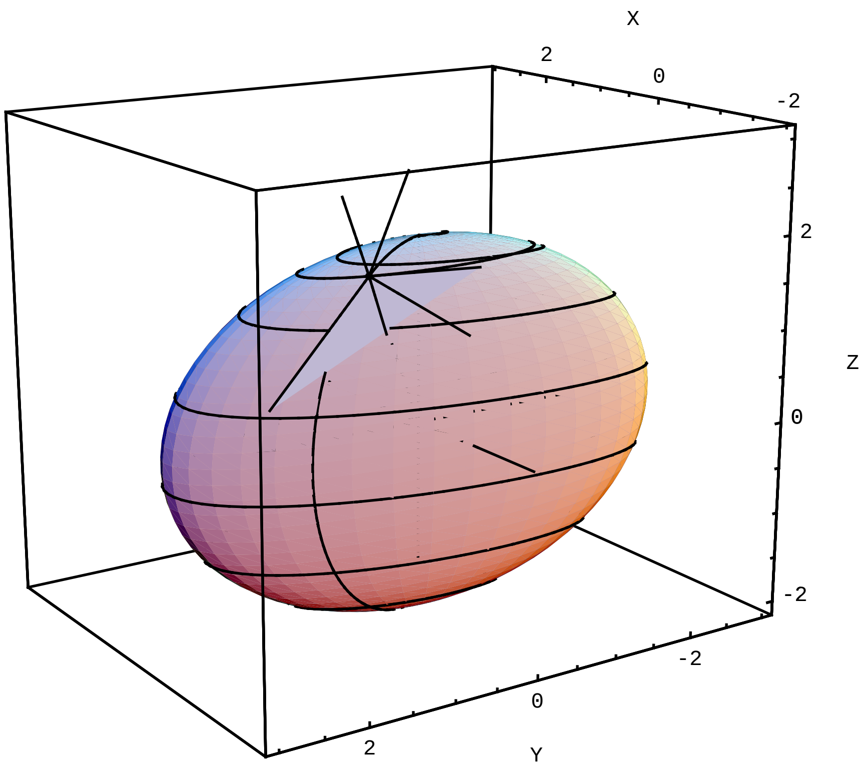
True

```
pWeg = ParametricPlot3D[path[t], {t, 0, 1}, PlotPoints → nr 75, DisplayFunction → Identity];
pvt[t_] = Graphics3D[{Hue[0], Line[{path[t], path[t] + path'[t] / 50}]}];
Show[pWeg, N[Table[pvt[1 / 20 n], {n, 0, 20}]],
  DisplayFunction → $DisplayFunction, ImageSize → {8 × 72, 3 × 72}, PlotLabel → msg];
```



★ A path with velocity vectors on a potatoid. ★

```
Show[
  Graphics3D[EdgeForm[], Axes → True, ImageSize → {8 × 72, 5 × 72},
   AxesLabel → {"X", "Y", "Z"}, ViewPoint → {-3, 2, 1} ],
  Potatoid,
  Graphics3D[Point[p]],
  CLϕp, CLθp,
  Graphics3D[Line[{{0, 0, 0}, 1.5 p}]],
  Graphics3D[Line[{{-2, 0, 0}, {2, 0, 0}}]],
  Graphics3D[Line[{{0, -2, 0}, {0, 2, 0}}]],
  Graphics3D[Line[{{0, 0, -2}, {0, 0, 2}}]],
  na, nb, dc, dd,
  Graphics3D[Polygon[{p, p + a, p + b}]],
  Graphics3D[Line[{p - a × b, p + 3 a × b}]],
  pWeg
 ];
```



★ Potatoid with natural basis, dual basis, radial line and orthogonal line at point p; a path on the surface; axes. ★

```
Export[
 "C:\\USERS\\LUMA\\Mathematica\\_Tensorrechnung\\1.6 – Potatoid_plus.dxf", %, "DXF"]
```

```
C:\USERS\LUMA\Mathematica\_Tensorrechnung\1.6 – Potatoid_plus.dxf
```

"Well, this was instructive but at some point we will have to learn how to do this without the embedding space."

1.7 Manifolds p. 35-37

coming soon...

1.8 Tensor Fields on manifolds p. 38 - 43
"We can create new tensors from old tensors by a number of methods."

coming soon...

1.9 Metric properties p. 43 - 46 *(pseudo-Riemannian manifolds)*

coming soon...

1.10 What and where are the bases? p. 46 - 49

coming soon...

Chapter 2: The spacetime of general relativity and paths of particles

2.0 Introduction p. 53 - 56

FN: "The mathematics of geodesics is covered in the next few sections, along with the related concepts of parallelism and absolute and covariant differentiation [...]. Note that in the present chapter we are concerned only with the motion of particles in a *given* spacetime [...]. How that field arises is answered in the next chapter, where we relate the curvature of spacetime to the sources of the gravitational field."

"The surface of a cylinder does not have curvature. Therefore, initially parallel geodesics would not deviate from one another. The ants could not decide if they were on a cylinder or a flat plane simply by examining their geodesics." (David Park)

2.1 Geodesics p. 56 - 64

```
Needs["TGeneralRelativity1`GeneralRelativity`"]
$PrePrint = .
DeclareBaseIndices[{1, 2, 3, 4}]
labs = {x, δ, g, Γ};
DefineTensorShortcuts[
 {{x, e, λ, u, zero}, 1},
 {{δ, g}, 2},
 {{Γ}, 3}]
DeclareZeroTensor[zero]
$Assumptions = -π < ϕ ≤ π && 0 ≤ θ ≤ π;
```

---

1) Derivation of the *affinely parametrized geodesic equation in Euclidean space* using the *straightness* concept. General parametrization.

```
Print[
 "For a straight line x(u) all the tangent vectors λ(u) point in the same direction.
   If we use the arc-length s as a parameter u, then the tangent vector is"]
λ == TotalD[x, s] // TraditionalForm
Print["Constant direction of tangent vector implies"]
TotalD[NestedTensor[λ], s] == 0
Print["Substitute component expression for λ"]
%% /. λ → λu[i] ed[i]
Print["Expand the total derivative by Unnesting the tensor"]
%% // UnnestTensor
Print["Expand the total derivative of the basis vectors in terms of the coordinates"]
MapAt[ExpandTotalD[labs, a], %%, {{1, 1}}] // TraditionalForm
Print["The partial derivatives of e are expanded
   in terms of e and connection coefficients Γ (to be defined), "]
PartialD[labs][ed[i_], xu[j_]] → Γudd[k, i, j] ed[k]
%%% /. PartialD[labs][ed[i_], xu[j_]] → Γudd[k, i, j] ed[k]
Print["Reindex the first term and factor"]
MapAt[IndexChange[{{k, i}, {i, j}, {a, k}}], %%, 1]
MapAt[Factor, %, 1]
Print["Therefore the bracketed expression must be zero"]
MapAt[Rest, %%, 1]
Print["Substitute λ in terms of total derivative of x to
   obtain the geodesic equation with the arc-length s as parameter"]
GeodEqArcLenght = %% /. λu[i_] → TotalD[xu[i], s];
% // FrameBox // DisplayForm
Print["Compare with eqn[2, 4], p.57."]
```

For a straight line x(u) all the tangent vectors λ(u) point in the same direction. If we use the arc-length s as a parameter u, then the tangent vector is

$$\lambda = \frac{dx}{ds}$$

Constant direction of tangent vector implies

$$\frac{d\lambda}{ds} == 0$$

Substitute component expression for λ

$$\frac{d(e_i \lambda^i)}{ds} == 0$$

Expand the total derivative by Unnesting the tensor

$$\lambda^i \frac{de_i}{ds} + e_i \frac{d\lambda^i}{ds} == 0$$

Expand the total derivative of the basis vectors in terms of the coordinates

$$e_i \frac{d\lambda^i}{ds} + \lambda^i \frac{dx^a}{ds} \frac{\partial e_i}{\partial x^a} = 0$$

The partial derivatives of e are expanded
in terms of e and connection coefficients Γ (to be defined),

$$\partial_{x^{j\_}} e_{i\_} \to e_k \Gamma^k{}_{ij}$$

$$e_k \Gamma^k{}_{ia} \lambda^i \frac{dx^a}{ds} + e_i \frac{d\lambda^i}{ds} == 0$$

Reindex the first term and factor

$$e_i \Gamma^i{}_{jk} \lambda^j \frac{dx^k}{ds} + e_i \frac{d\lambda^i}{ds} == 0$$

$$e_i \left(\Gamma^i{}_{jk} \lambda^j \frac{dx^k}{ds} + \frac{d\lambda^i}{ds}\right) == 0$$

Therefore the bracketed expression must be zero

$$\Gamma^i{}_{jk} \lambda^j \frac{dx^k}{ds} + \frac{d\lambda^i}{ds} == 0$$

Substitute λ in terms of total derivative of x
to obtain the geodesic equation with the arc-length s as parameter

$$\boxed{\frac{d^2 x^i}{ds\,ds} + \Gamma^i{}_{jk} \frac{dx^j}{ds} \frac{dx^k}{ds} == 0}$$

Compare with eqn[2, 4], p.57.

```
Print["If a general parameter u=u[s] (inverse transformation s=s[u]) is used to
   parameterize a straight line,then the geodesic equation takes the form..."]
GeodEqArcLenght
x[u[s]];
{%, D[%, s], D[%, {s, 2}]}
{rule1 = TotalD[xu[i_], s] → TotalD[xu[i], u] D[u[s], s],
 rule2 = TotalD[xu[i_], {s, s}] →
   TotalD[xu[i], {u, u}] D[u[s], s]^2 + TotalD[xu[i], u] D[u[s], {s, 2}]}
GeodEqGeneralParameter = GeodEqArcLenght /. {rule1, rule2} // FullSimplify
# - %[[1, 2]] & /@ %;
# / %[[1, 2]] & /@ %
-u''[s] / u'[s]^2 → h[s[u]]
%% /. % // FrameBox // DisplayForm
Print["See Exercise 2.1.1, p. 63."]
```

If a general parameter u=u[s] (inverse transformation s=s[u]) is used
to parameterize a straight line,then the geodesic equation takes the form...

$$\frac{d^2x^i}{ds\,ds}+\Gamma^i{}_{jk}\frac{dx^j}{ds}\frac{dx^k}{ds}==0$$

$$\{x[u[s]],\ u'[s]\,x'[u[s]],\ x'[u[s]]\,u''[s]+u'[s]^2\,x''[u[s]]\}$$

$$\left\{\frac{dx^{i}{}_{-}}{ds}\to\frac{dx^i}{du}u'[s],\ \frac{d^2x^{i}{}_{-}}{ds\,ds}\to\frac{d^2x^i}{du\,du}u'[s]^2+\frac{dx^i}{du}u''[s]\right\}$$

$$\left(\frac{d^2x^i}{du\,du}+\Gamma^i{}_{jk}\frac{dx^j}{du}\frac{dx^k}{du}\right)u'[s]^2+\frac{dx^i}{du}u''[s]==0$$

$$\frac{d^2x^i}{du\,du}+\Gamma^i{}_{jk}\frac{dx^j}{du}\frac{dx^k}{du}==-\frac{\frac{dx^i}{du}u''[s]}{u'[s]^2}$$

$$-\frac{u''[s]}{u'[s]^2}\to h[s[u]]$$

$$\boxed{\frac{d^2x^i}{du\,du}+\Gamma^i{}_{jk}\frac{dx^j}{du}\frac{dx^k}{du}==h[s[u]]\frac{dx^i}{du}}$$

See Exercise 2.1.1, p. 63.

```
Print["If we use an affine parameter w=u[s]=As+B where A≠0 and B are
   constants, then the affinely parametrized geodesic equation is..."]
u[s_] := A s + B
GeodEqAffineParameter =
  (MapAt[#1 / A^2 &, GeodEqGeneralParameter, 1] // Simplify // IndexChange[{i, a}] //
     IndexChange[{{j, b}, {k, c}}]) /. u → w;
% // FrameBox // DisplayForm
Print["Compare with eqn[2, 11], p.58."]
```

If we use an affine parameter w=u[s]=As+B where A≠0 and
 B are constants, then the affinely parametrized geodesic equation is...

$$\boxed{\frac{d^2x^a}{dw\,dw}+\Gamma^a{}_{bc}\frac{dx^b}{dw}\frac{dx^c}{dw}==0}$$

Compare with eqn[2, 11], p.58.

■ 2) Derivation of the connection coefficients in terms of the metric in Euclidean space (*Christoffel symbols*).

WARNING: *Mathematica* supposes partial differentiation to be commutative!

```
{D[f[x, y], x, y], D[f[x, y], y, x]}
{∂x,y f[x, y], ∂y,x f[x, y]}
HoldComplete[∂x,y f[x, y] == ∂y,x f[x, y]]
% // ReleaseHold
{PartialD[NestedTensor[f], {i, j}], PartialD[NestedTensor[f], {j, i}]};
Equal @@ %
%% // ExpandPartialD[labs] // TraditionalForm
Equal @@ %
```

$$\{f^{(1,1)}[x,y],\ f^{(1,1)}[x,y]\}$$

$$\{f^{(1,1)}[x,y],\ f^{(1,1)}[x,y]\}$$

```
HoldComplete[∂x,y f[x, y] == ∂y,x f[x, y]]
```

True

$f_{,i,j} == f_{,j,i}$

$$\left\{\frac{\partial^2 f}{\partial x^i \partial x^j}, \frac{\partial^2 f}{\partial x^i \partial x^j}\right\}$$

True

```
Print["The order of partial differentiation is commutative
   with smooth functions (continuous second partial derivatives)."]
{PartialD[ed[i], j], PartialD[ed[j], i]}
% // ExpandPartialD[labs] // TraditionalForm
% /. ed[k_] → PartialD[NestedTensor[r], k]
% // ExpandPartialD[labs] // TraditionalForm
Print["Definition of metric tensor"]
gdd[i, j] == ed[i].ed[j]
Print["Taking the partial derivative of each side"]
PartialD[#, k] & /@ %%
Print["Substituting the expansion in terms of Christoffel symbols"]
%% /. PartialD[ed[i_], j_] → Γudd[m, i, j] ed[m]
Print["Evaluating the dot products"]
(eqn[1] = %% // EvaluateDotProducts[e, g, False]) // FrameBox // DisplayForm
```

The order of partial differentiation is commutative
with smooth functions (continuous second partial derivatives).

$\{e_{i,j}, e_{j,i}\}$

$$\left\{\frac{\partial e_i}{\partial x^j}, \frac{\partial e_j}{\partial x^i}\right\}$$

$\{\partial_{x^j} r_{,i}, \partial_{x^i} r_{,j}\}$

$$\left\{\frac{\partial^2 r}{\partial x^i \partial x^j}, \frac{\partial^2 r}{\partial x^i \partial x^j}\right\}$$

Definition of metric tensor

$g_{ij} == e_i . e_j$

Taking the partial derivative of each side

$g_{ij,k} == e_{i,k} . e_j + e_{j,k} . e_i$

Substituting the expansion in terms of Christoffel symbols

$g_{ij,k} == (e_m \Gamma^m{}_{ik}) . e_j + (e_m \Gamma^m{}_{jk}) . e_i$

Evaluating the dot products

$$\boxed{g_{ij,k} == g_{mj} \Gamma^m{}_{ik} + g_{mi} \Gamma^m{}_{jk}}$$

```
eqn[1]
eqn[2] = eqn[1] // IndexChange[Transpose[{{i, j, k}, {j, k, i}}]]
eqn[3] = eqn[1] // IndexChange[Transpose[{{i, j, k}, {k, i, j}}]]
```

```
Print["Add the first two equations"]
Inner[Plus, eqn[1], eqn[2], Equal]
Print["Subtract the third equation"]
Inner[Subtract, %%, eqn[3], Equal]
Print["Apply the symmetries"]
DeclareTensorSymmetries[g, 2, {1, {1, 2}}]
DeclareTensorSymmetries[Γ, 3, {1, {2, 3}}]
%%%% // SymmetrizeSlots[]
Print["Reverse, multiply by the inverse metric and simplify"]
guu[l, j] / 2 # & /@ Reverse[%%]
(eqn[4] = MapAt[MetricSimplify[g], %, 1]) // FrameBox // DisplayForm
Print["Compare with eqn[2, 9], p.58."]
Print["Lower the first index to obtain an expression for the down components of Γ"]
gdd[l, m] # & /@ eqn[4] // MetricSimplify[g]
Print["Reindex"]
%% // IndexChange[Transpose[{{m, i, k}, {a, b, c}}]];
MapAt[Factor, %, 2] // FrameBox // DisplayForm
```

$$g_{ij,k} == g_{mj}\,\Gamma^{m}{}_{ik} + g_{mi}\,\Gamma^{m}{}_{jk}$$

$$g_{jk,i} == g_{mk}\,\Gamma^{m}{}_{ji} + g_{mj}\,\Gamma^{m}{}_{ki}$$

$$g_{ki,j} == g_{mk}\,\Gamma^{m}{}_{ij} + g_{mi}\,\Gamma^{m}{}_{kj}$$

Add the first two equations

$$g_{ij,k} + g_{jk,i} == g_{mj}\,\Gamma^{m}{}_{ik} + g_{mk}\,\Gamma^{m}{}_{ji} + g_{mi}\,\Gamma^{m}{}_{jk} + g_{mj}\,\Gamma^{m}{}_{ki}$$

Subtract the third equation

$$g_{ij,k} + g_{jk,i} - g_{ki,j} == -g_{mk}\,\Gamma^{m}{}_{ij} + g_{mj}\,\Gamma^{m}{}_{ik} + g_{mk}\,\Gamma^{m}{}_{ji} + g_{mi}\,\Gamma^{m}{}_{jk} + g_{mj}\,\Gamma^{m}{}_{ki} - g_{mi}\,\Gamma^{m}{}_{kj}$$

Apply the symmetries

$$g_{ij,k} + g_{jk,i} - g_{ki,j} == 2\,g_{jm}\,\Gamma^{m}{}_{ik}$$

Reverse, multiply by the inverse metric and simplify

$$g^{lj}\,g_{jm}\,\Gamma^{m}{}_{ik} == \frac{1}{2}\,g^{lj}\,(g_{ij,k} + g_{jk,i} - g_{ki,j})$$

$$\boxed{\Gamma^{l}{}_{ik} == \frac{1}{2}\,g^{lj}\,(g_{ij,k} + g_{jk,i} - g_{ki,j})}$$

Compare with eqn[2, 9], p.58.

Lower the first index to obtain an expression for the down components of Γ

$$\Gamma_{mik} == \frac{1}{2}\,g_{im,k} - \frac{1}{2}\,g_{ki,m} + \frac{1}{2}\,g_{mk,i}$$

Reindex

$$\boxed{\Gamma_{abc} == \frac{1}{2}\,(g_{ac,b} + g_{ba,c} - g_{cb,a})}$$

?? ChristoffelDownRule

ChristoffelDownRule gives the rule for the Γ Christoffel down elements in terms of the metric g.

ChristoffelDownRule $= \Gamma_{abc} \to \frac{1}{2}\,(g_{ac,b} + g_{ba,c} - g_{bc,a})$

So we have now expressions for the Christoffel symbols in terms of the metric and, given a specific metric, we can actually evaluate the geodesic equations.

■ Example 2.1.1 p. 59-60. geodesics on a sphere (in 3D space)

```
r[θ_, ϕ_] = {a Cos[ϕ] Sin[θ], a Sin[ϕ] Sin[θ], a Cos[θ]};
naturalbasis = {eθ[θ_, ϕ_] = ∂θ %, eϕ[θ_, ϕ_] = ∂ϕ %}
metricmatrix = naturalbasis.Transpose[naturalbasis] // Simplify;
MatrixForm /@
 {"3D-position:", r[θ, ϕ], "natural basis:", eθ[θ, ϕ], eϕ[θ, ϕ], "metric:", metricmatrix}
```

{{a Cos[θ] Cos[ϕ], a Cos[θ] Sin[ϕ], -a Sin[θ]}, {-a Sin[θ] Sin[ϕ], a Cos[ϕ] Sin[θ], 0}}

$$\left\{\text{3D-position:}, \begin{pmatrix} a\,\text{Cos}[\phi]\,\text{Sin}[\theta] \\ a\,\text{Sin}[\theta]\,\text{Sin}[\phi] \\ a\,\text{Cos}[\theta] \end{pmatrix}, \text{natural basis:}, \begin{pmatrix} a\,\text{Cos}[\theta]\,\text{Cos}[\phi] \\ a\,\text{Cos}[\theta]\,\text{Sin}[\phi] \\ -a\,\text{Sin}[\theta] \end{pmatrix}, \begin{pmatrix} -a\,\text{Sin}[\theta]\,\text{Sin}[\phi] \\ a\,\text{Cos}[\phi]\,\text{Sin}[\theta] \\ 0 \end{pmatrix}, \text{metric:}, \begin{pmatrix} a^2 & 0 \\ 0 & a^2\,\text{Sin}[\theta]^2 \end{pmatrix}\right\}$$

```
Print["Metric:"]
DeclareBaseIndices[{1, 2}]
metric = metricmatrix // CoordinatesToTensors[{θ, ϕ}, u];
SetMetricValueRules[g, metric]
NonzeroValueRules[g] // TableForm
Print["Christoffel symbols:"]
SetChristoffelValueRules[uu[i], metric, Γ]
NonzeroValueRules[Γ] // TableForm
SelectedTensorRules[Γ, Γudd[_, a_, b_] /; OrderedQ[{a, b}]] ⋃
 SelectedTensorRules[Γ, Γddd[_, a_, b_] /; OrderedQ[{a, b}]]
Print["Geodesic equation:"]
TotalD[uu[i], {s, s}] + Γudd[i, j, k] TotalD[uu[j], s] TotalD[uu[k], s] == zerou[i]
GeodesicEq = % // ToArrayValues[] // UseCoordinates[{θ[s], ϕ[s]}, u]
```

Metric:

$g_{11} \to a^2$

$g_{22} \to a^2\,\text{Sin}[u^1]^2$

$g^{11} \to \frac{1}{a^2}$

$g^{22} \to \frac{\text{Csc}[u^1]^2}{a^2}$

Christoffel symbols:

$\Gamma^1{}_{22} \to -\text{Cos}[u^1]\,\text{Sin}[u^1]$

$\Gamma^2{}_{12} \to \text{Cot}[u^1]$

$\Gamma^2{}_{21} \to \text{Cot}[u^1]$

$\Gamma_{122} \to -a^2\,\text{Cos}[u^1]\,\text{Sin}[u^1]$

$\Gamma_{212} \to a^2\,\text{Cos}[u^1]\,\text{Sin}[u^1]$

$\Gamma_{221} \to a^2\,\text{Cos}[u^1]\,\text{Sin}[u^1]$

$\{\Gamma^1{}_{22} \to -\text{Cos}[u^1]\,\text{Sin}[u^1],\ \Gamma^2{}_{12} \to \text{Cot}[u^1],\ \Gamma_{122} \to -a^2\,\text{Cos}[u^1]\,\text{Sin}[u^1],\ \Gamma_{212} \to a^2\,\text{Cos}[u^1]\,\text{Sin}[u^1]\}$

Geodesic equation:

$$\frac{d^2u^i}{ds\,ds} + \Gamma^i{}_{jk}\frac{du^j}{ds}\frac{du^k}{ds} == \text{zero}^i$$

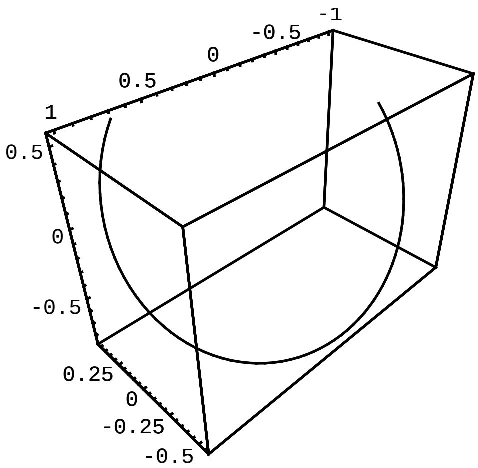
$$\left\{-\text{Cos}[\theta[s]]\,\text{Sin}[\theta[s]]\,\phi'[s]^2 + \theta''[s] == 0,\ 2\,\text{Cot}[\theta[s]]\,\theta'[s]\,\phi'[s] + \phi''[s] == 0\right\}$$

Try to solve the geodesic equation with initial conditions analytically and numerically.

```
eqs = GeodesicEq ⋃ {θ[0] == 1, θ'[0] == 1, ϕ[0] == 1, ϕ'[0] == 1};
DSolve[eqs, {θ[s], ϕ[s]}, s]
NDSolve[eqs, {θ[s], ϕ[s]}, {s, 0, 2 π}]
r[θ[s], ϕ[s]] /. %[[1]] /. a → 1;
ParametricPlot3D[%, {s, 0, π}, ViewPoint → {-1, 1, 1 }, ImageSize → {8 × 72, 3 × 72}];
```

```
DSolve[{θ[0] == 1, ϕ[0] == 1, θ′[0] == 1, ϕ′[0] == 1,
  -Cos[θ[s]] Sin[θ[s]] ϕ′[s]^2 + θ″[s] == 0, 2 Cot[θ[s]] θ′[s] ϕ′[s] + ϕ″[s] == 0}, {θ[s], ϕ[s]}, s]
```

```
{{θ[s] → InterpolatingFunction[{{0., 6.28319}}, <>][s],
  ϕ[s] → InterpolatingFunction[{{0., 6.28319}}, <>][s]}}
```



★ Plot of the numerically found geodesic. ★

For circles of latitude the geodesic equation is satisfied only by the equator.

```
Block[{θ, ϕ, θ0, a},
 (* circles of latitude *)
 θ[s_] := θ0; ϕ[s_] := s / (a Sin[θ0]);
 Print[GeodesicEq, ",  ", {θ[s], θ'[s], ϕ[s], ϕ'[s]} /. s → 0];
 θ0 = π / 2;
 Print[GeodesicEq, ",  ", {θ[s], θ'[s], ϕ[s], ϕ'[s]} /. s → 0];
]
```

$$\left\{-\frac{\text{Cot}[\theta 0]}{a^2} == 0,\ \text{True}\right\},\ \left\{\theta 0,\ 0,\ 0,\ \frac{\text{Csc}[\theta 0]}{a}\right\}$$

$$\{\text{True},\ \text{True}\},\ \left\{\frac{\pi}{2},\ 0,\ 0,\ \frac{1}{a}\right\}$$

■ Example 2.1.2 p. 61-63. geodesic equation for the Robertson-Walker spacetime

```
DeclareBaseIndices[{0, 1, 2, 3}]
Var = {t, r, θ, ϕ};
```

$$\text{cmetric} = \text{DiagonalMatrix}\left[\left\{1,\ -R[t]^2\,\frac{1}{1 - k\,r^2},\ -R[t]^2\,r^2,\ -R[t]^2\,r^2\,\text{Sin}[\theta]^2\right\}\right];$$

```
% // MatrixForm
Dt[s]^2 == Collect[Dt[Var].cmetric.Dt[Var], -R[t]^2] // TraditionalForm
```

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -\frac{R[t]^2}{1-k\,r^2} & 0 & 0 \\ 0 & 0 & -r^2\,R[t]^2 & 0 \\ 0 & 0 & 0 & -r^2\,R[t]^2\,\text{Sin}[\theta]^2 \end{pmatrix}$$

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

```
BeginTime = SessionTime[];
metric = cmetric // CoordinatesToTensors[Var]
SetMetricValues[g, metric]
SetChristoffelValueRules[xu[i], metric, Γ, Simplify[#, Trig → False] &]
SelectedTensorRules[Γ, Γudd[a_, b_, c_] /; OrderedQ[{b, c}]] //
  UseCoordinates[Var] // TableForm
Print["Geodesic equation for the Robertson-Walker spacetime"]
TotalD[xu[i], {u, u}] + Γudd[i, j, k] TotalD[xu[j], u] TotalD[xu[k], u] == zerou[i]
% // ToArrayValues[] // UseCoordinates[Var];
Collect[#, R[t] R'[t]] & //@ %;
Collect[#, r (k r^2 - 1)] & //@ %;
% // TableForm // TraditionalForm
Print["Compare with eqn[2, 20], p.62."]
Print["- - - - Time used: ", N[SessionTime[] - BeginTime, 3], " s"]
```

$$\left\{\{1, 0, 0, 0\}, \left\{0, -\frac{R[x^0]^2}{1-k\,(x^1)^2}, 0, 0\right\}, \left\{0, 0, -R[x^0]^2\,(x^1)^2, 0\right\}, \left\{0, 0, 0, -R[x^0]^2\,\text{Sin}[x^2]^2\,(x^1)^2\right\}\right\}$$

$\Gamma^0{}_{11} \to \frac{R[t]\,R'[t]}{1-k\,r^2}$

$\Gamma^0{}_{22} \to r^2\,R[t]\,R'[t]$

$\Gamma^0{}_{33} \to r^2\,R[t]\,\text{Sin}[\theta]^2\,R'[t]$

$\Gamma^1{}_{01} \to \frac{R'[t]}{R[t]}$

$\Gamma^1{}_{11} \to \frac{k\,r}{1-k\,r^2}$

$\Gamma^1{}_{22} \to r\,(-1+k\,r^2)$

$\Gamma^1{}_{33} \to r\,(-1+k\,r^2)\,\text{Sin}[\theta]^2$

$\Gamma^2{}_{02} \to \frac{R'[t]}{R[t]}$

$\Gamma^2{}_{12} \to \frac{1}{r}$

$\Gamma^2{}_{33} \to -\text{Cos}[\theta]\,\text{Sin}[\theta]$

$\Gamma^3{}_{03} \to \frac{R'[t]}{R[t]}$

$\Gamma^3{}_{13} \to \frac{1}{r}$

$\Gamma^3{}_{23} \to \text{Cot}[\theta]$

Geodesic equation for the Robertson-Walker spacetime

$$\frac{d^2x^i}{du\,du} + \Gamma^i{}_{jk}\,\frac{dx^j}{du}\,\frac{dx^k}{du} == \text{zero}^i$$

$$\frac{d^2t}{du^2}+R(t)\left(\frac{\left(\frac{dr}{du}\right)^2}{1-k\,r^2}+r^2\left(\frac{d\theta}{du}\right)^2+r^2\left(\frac{d\phi}{du}\right)^2\sin^2(\theta)\right)R'(t)=0$$

$$\frac{k\,r\left(\frac{dr}{du}\right)^2}{1-k\,r^2}+\frac{2\,\frac{dt}{du}\,R'(t)\,\frac{dr}{du}}{R(t)}+\frac{d^2r}{du^2}+r\left(k\,r^2-1\right)\left(\left(\frac{d\theta}{du}\right)^2+\left(\frac{d\phi}{du}\right)^2\sin^2(\theta)\right)=0$$

$$-\cos(\theta)\sin(\theta)\left(\frac{d\phi}{du}\right)^2+\frac{2\,\frac{dr}{du}\,\frac{d\theta}{du}}{r}+\frac{d^2\theta}{du^2}+\frac{2\,\frac{dt}{du}\,\frac{d\theta}{du}\,R'(t)}{R(t)}=0$$

$$\frac{2\,\frac{dr}{du}\,\frac{d\phi}{du}}{r}+2\cot(\theta)\,\frac{d\theta}{du}\,\frac{d\phi}{du}+\frac{2\,\frac{dt}{du}\,R'(t)\,\frac{d\phi}{du}}{R(t)}+\frac{d^2\phi}{du^2}=0$$

```
Compare with eqn[2, 20], p.62.

- - - - Time used: 10.4 s
```

■ More examples: geodesics on special surfaces (in 3D space).

```
msg = "trumpet ('mbuti, vuvuzela)";
Reduce[{ρ > 0, π / 2 < θ < π, Abs[z] / ρ == Tan[θ - π / 2],
   z == -1 / ρ, x == ρ Cos[ϕ], y == ρ Sin[ϕ]}, {ρ, x, y, z}, Reals]
r[ϕ_, θ_] = {%[[3, 2]], %[[4, 2]], %[[5, 2]]} /. θ → (θ + π) / 2;
```

$$\frac{\pi}{2}<\theta<\pi\;\&\&\;\rho==\sqrt{-\mathrm{Tan}[\theta]}\;\&\&\;x==\mathrm{Cos}[\phi]\sqrt{-\mathrm{Tan}[\theta]}\;\&\&\;y==\mathrm{Sin}[\phi]\sqrt{-\mathrm{Tan}[\theta]}\;\&\&\;z==-\frac{1}{\sqrt{-\mathrm{Tan}[\theta]}}$$

```
msg = "sphere";
r[ϕ_, θ_] = {Cos[ϕ] Sin[θ], Sin[ϕ] Sin[θ], Cos[θ]};

msg = "cardio potatoid";
pϕ[ϕ_] = -(-π - ϕ)^2 (1 / 3 + 3 ϕ) (1 / 2 - ϕ) (π - ϕ)^2 / 550;
pθ[θ_] = θ^2 (2 - θ) (π - θ)^2 / 4;
r[ϕ_, θ_] = (1 + pϕ[ϕ] pθ[θ]) {Cos[ϕ] Sin[θ], Sin[ϕ] Sin[θ], Cos[θ]};

Print[msg]
Print["3D-position:"]
r[ϕ, θ] // MatrixForm // Short
naturalbasis = {e_ϕ, e_θ} = {∂_ϕ r[ϕ, θ], ∂_θ r[ϕ, θ]};
Print["natural basis:"]
MatrixForm /@ {e_ϕ, e_θ} // Short
Print["metric:"]
metricmatrix = naturalbasis.Transpose[naturalbasis] (*//Simplify*);
% // Short // MatrixForm
Potatoid = ParametricPlot3D[r[ϕ, θ], {ϕ, -π, π},
  {θ, 0 + 2 $MachineEpsilon, π}, PlotPoints -> {40, 40}, DisplayFunction → Identity]

cardio potatoid

3D-position:
```

$$\begin{pmatrix} \left(1-\frac{(2-\theta)\ \ll 6\gg}{2200}\right)\mathrm{Cos}[\phi]\,\mathrm{Sin}[\theta] \\ \ll 1\gg \\ (\ll 1\gg)\ \ll 1\gg \end{pmatrix}$$

```
natural basis:
```

$$\left\{\begin{pmatrix} (\ll 1\gg)\,\mathrm{Cos}[\phi]\,\mathrm{Sin}[\theta]-\ll 1\gg \\ \ll 1\gg+\ll 1\gg \\ (\ll 1\gg)\ \ll 1\gg\ \ll 1\gg \end{pmatrix},\ \begin{pmatrix} \ll 1\gg+(\ll 1\gg)\ \ll 1\gg\ \mathrm{Sin}[\theta] \\ \ll 1\gg \\ \ll 1\gg \end{pmatrix}\right\}$$

```
metric:
```

$$\left\{\left\{(\ll 1\gg)^2\,\mathrm{Cos}[\theta]^2+(\ll 1\gg)^2+(\ll 1\gg-\ll 1\gg)^2,\ \ll 1\gg\right\},\ \ll 1\gg\right\}$$

```
- Graphics3D -
```

```
DeclareBaseIndices[{1, 2}]
metric = metricmatrix // CoordinatesToTensors[{ϕ, θ}, u];
Block[{SetMetricValueRules},
 SetMetricValueRules[g_, MT_, flavor_: Identity] :=
  Module[{i, j},
   SetTensorValueRules[Tensor[g, {Void, Void}, {i, j}] // ToFlavor[flavor], MT];
   SetTensorValueRules[Tensor[g, {i, j}, {Void, Void}] // ToFlavor[flavor],
    Identity[Inverse[MT]]]];
 SetMetricValueRules[g, metric]; SetChristoffelValueRules[uu[i], metric, Γ];
]
Print["g:"];
NonzeroValueRules[g] // Shallow // TableForm
Print["Γ:"];
SelectedTensorRules[Γ, Γudd[_, a_, b_] /;
    OrderedQ[{a, b}]] // Shallow // TableForm
Print["Geodesic equation:"]
TotalD[uu[i], {s, s}] + Γudd[i, j, k] TotalD[uu[j], s] TotalD[uu[k],
    s] == zerou[i]
GeodesicEq = % // ToArrayValues[] // UseCoordinates[{ϕ[s], θ[s]}, u];
GeodesicEq // Short
eqs := GeodesicEq ⋃ IBConds;
```

```
g:

{<<3>> → Plus[<<3>>], <<3>> → Plus[<<3>>], <<3>> → Plus[<<3>>], <<3>> → Plus[<<3>>],
 <<3>> → Times[<<2>>], <<3>> → Times[<<2>>], <<3>> → Times[<<2>>], <<3>> → Times[<<2>>]}

Γ:

{<<3>> → Times[<<2>>], <<3>> → Times[<<2>>], <<3>> → Times[<<2>>],
 <<3>> → Times[<<2>>], <<3>> → Times[<<2>>], <<3>> → Times[<<2>>]}

Geodesic equation:
```

$$\frac{d^2u^i}{ds\,ds} + \Gamma^i{}_{jk}\,\frac{du^j}{ds}\,\frac{du^k}{ds} == \text{zero}^i$$

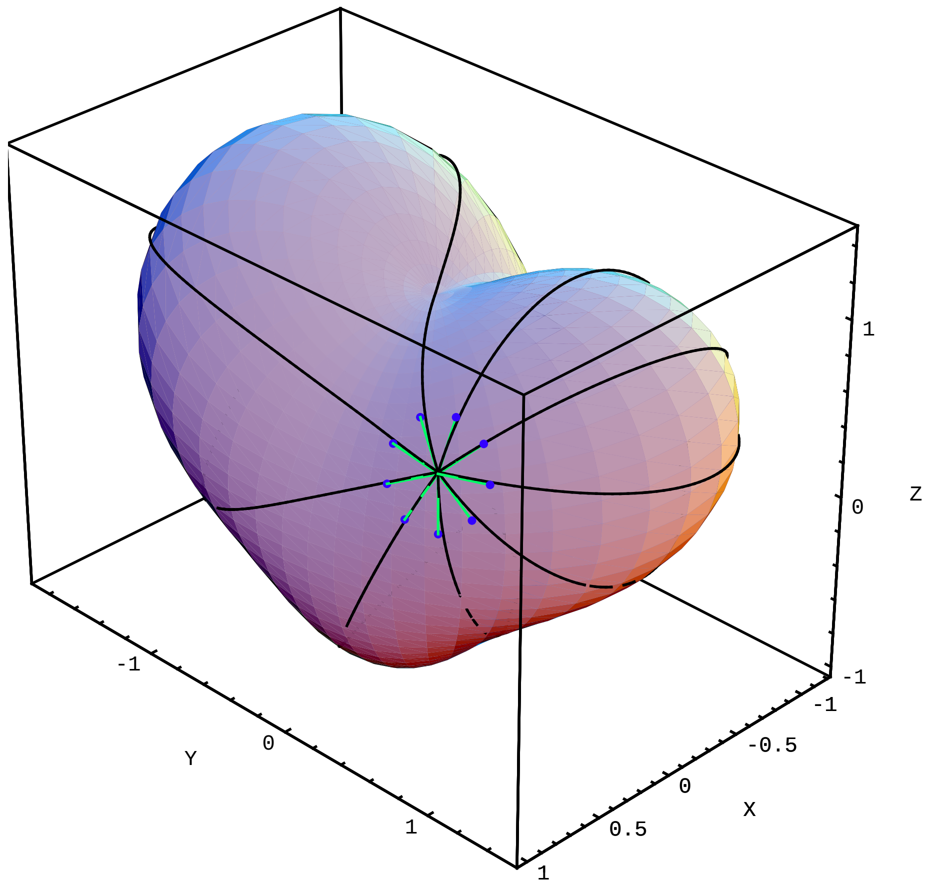
```
{<<1>> == 0, <<1>> == 0}
```

A collection of geodetics starting at the same point on the chosen surface:

```
ι = .4; sRange = {s, 0, ι 2 π };
geodplots = {}; Clear[funs, IBvecs]
Δs = .001; μ = 250; α = 1;
{ϕ0, θ0} = { π / 4, π / 3};
nmax = 9;
Do[(
  β = (n - 1) 2 π / nmax;
  IBConds = {θ[0] == θ0, θ'[0] == α Cos[β], ϕ[0] == ϕ0, ϕ'[0] == α Sin[β]};
  solrule = NDSolve[eqs, {ϕ, θ}, sRange];
  {ϕsol, θsol} = {solrule[[1, 1, 2]], solrule[[1, 2, 2]]};
  geodesic[s_] := r[ϕsol[s], θsol[s]];
  funs[n] = {ϕsol, θsol, geodesic};
  AppendTo [geodplots, ParametricPlot3D[
     1.001 geodesic[s] + {0, 0, .01}, Evaluate[sRange], DisplayFunction → Identity]];
  r0 = r[ϕsol[0], θsol[0]];
  rs = r[ϕsol[0] + ϕsol'[0] Δs, θsol[0] + θsol'[0] Δs];
  Δr = rs - r0;
  IBvecs[n] = {
```

```
    Graphics3D[{Hue[0], PointSize[0.02]}, Point[r[ϕsol[0]], θsol[0]]],
    Graphics3D[{Hue[0.4], Line[{r0, r0 + μ Δr}]}],
    Graphics3D[{Hue[0.7], PointSize[0.01], Point[r0 + μ Δr]}]};
  AppendTo[geodplots, IBvecs[n]];
 ),
 {n, 1, nmax}] (* End of Do *)
```

```
vP = 5 r[ϕ0, θ0];
Show[{
   Graphics3D[EdgeForm[]], Potatoid,
   geodplots, Graphics3D[{PointSize[0.005], Point[{0, 0, 0}]}]},
  Axes → True, AxesLabel → {"X", "Y", "Z"},
  DisplayFunction → $DisplayFunction, ViewPoint → vP, ImageSize → 72 × 6];
```



★ Plot of the chosen surface with a collection of geodetics starting at the same point. ★

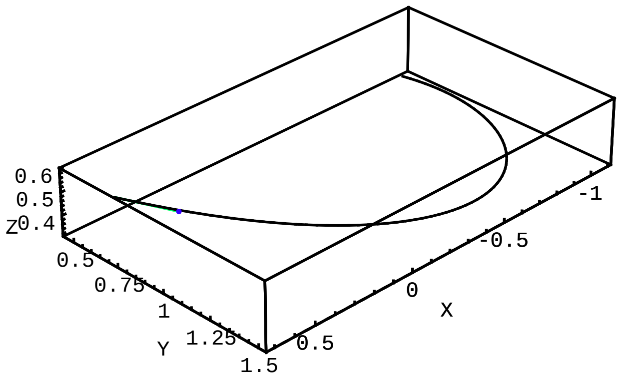
```
Export["C:\\USERS\\LUMA\\Mathematica\\_Tensorrechnung\\2.1
   - Geodesics_on_special_surfaces.dxf", %, "DXF"]
```

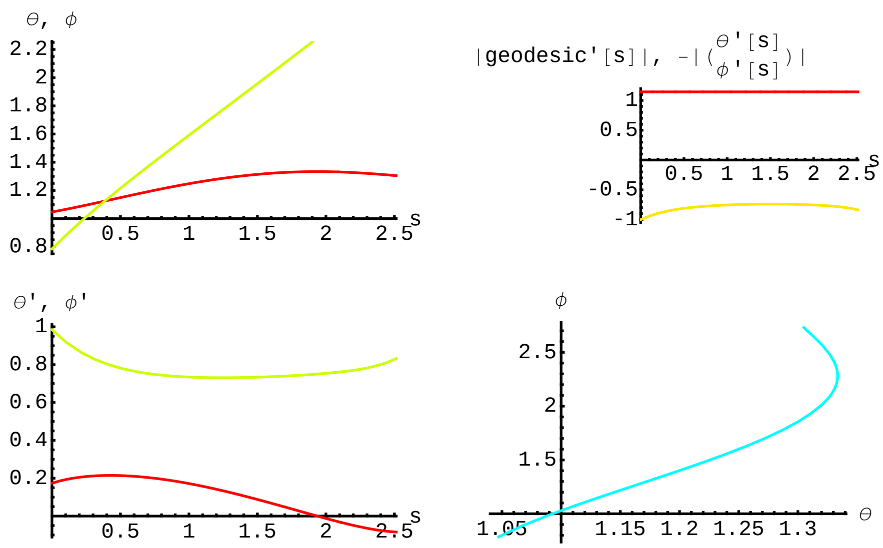
```
C:\USERS\LUMA\Mathematica\_Tensorrechnung\2.1 - Geodesics_on_special_surfaces.dxf
```

More details on a selected geodetics:

```
nselect = 3;
{ϕsol, θsol, geodesic} = funs[nselect];
pl0 = ParametricPlot3D[geodesic[s], Evaluate[sRange],
   AxesLabel → {"X", "Y", "Z"}, ViewPoint → vP, DisplayFunction → Identity];
pl1 = Plot[{θsol[s], ϕsol[s]}, Evaluate[sRange], PlotRange → {{0, sRange[[3]]}, Automatic},
   AxesLabel → {"s", "θ, ϕ"}, PlotStyle → {Hue[0], Hue[0.2]}, DisplayFunction → Identity];
pl2 = Plot[{θsol'[s], ϕsol'[s]}, Evaluate[sRange],
   PlotRange → {{0, sRange[[3]]}, Automatic}, AxesLabel → {"s", "θ', ϕ'"},
   PlotStyle → {Hue[0], Hue[0.2]}, DisplayFunction → Identity];
```

```
pl3 = Plot[
    {Norm[geodesic'[s]], -Norm[{θsol'[s], ϕsol'[s]}]}, Evaluate[sRange],
    PlotRange → {{0, sRange[[3]]}, All}, AxesLabel → {"s", "|geodesic'[s]|, -|(θ'[s]
                                                                              ϕ'[s])|"},
    PlotStyle → {Hue[0], Hue[0.15]}, DisplayFunction → Identity];
pl4 = ParametricPlot[{θsol[s], ϕsol[s]}, Evaluate[sRange], AxesLabel → {"θ", "ϕ"},
    PlotStyle → {Hue[0.5]}, DisplayFunction → Identity];
Show[pl0, IBvecs[nselect], DisplayFunction → $DisplayFunction, ImageSize → 72 × 4];
Show[GraphicsArray[{{pl1, pl3}, {pl2, pl4}}], ImageSize → 72 × 6];
```





★ Details on a selected geodetic. ★

"Geodesics are locally extremes of length. [...] Feynman has a cute illustration in a book of this. Suppose you want to arrive back where you are now in one hour of local time, but with a maximum of time having elapsed for you. Note that going uphill takes you to a place where, informally speaking, time goes faster. But moving fast causes time to go "slower" (informally speaking). What is the tradeoff between the two which leads to an optimum of wasted time? Geodesics in space-time are the *free-fall* paths of objects. So the right thing to do is to shoot yourself out of a cannon so that in free fall, you return to the same spot on the ground." (Keith Ramsay)

2.2 Parallel vectors along a curve p. 64 - 71

```
Needs["TGeneralRelativity1`GeneralRelativity`"]
$PrePrint =.
DeclareBaseIndices[{1, 2, 3, 4}]
labs = {x, δ, g, Γ};
DefineTensorShortcuts[
 {{x, e, λ, λ0, zero}, 1},
 {{δ, g}, 2},
 {{Γ}, 3}]
DeclareZeroTensor[zero]
DeclareTensorSymmetries[Γ, 3, {1, {2, 3}}]
(* A simple routine for formatted output of tensors: *)
ST[tensor_] :=
  Print["Tensor ", EinsteinArray[][tensor], " is ", ToArrayValues[][tensor]];
(* A victor is a graphical rappresentation of a vector as a pin. pin-
  head (green=START) ⟶ bloody pin-tip (red=STOP): *)
victor[a_, b_, linecolor_: Hue[.7]] := {
   Graphics3D[{Hue[.4], PointSize[0.02], Point[a]}],
   Graphics3D[{linecolor, Line[{a, a + b}]}],
   Graphics3D[{Hue[.0], PointSize[0.0075], Point[a + b]}]};
(* nbProj(v) are the components of a vector v in the basis {eθ, eϕ, eθ×eϕ}: *)
nbProj[v_] := {"v=", v // MatrixForm, "nbProj(v)=",
    {v.eθ[θ, ϕ], v.eϕ[θ, ϕ], v.eθ[θ, ϕ] × eϕ[θ, ϕ]} // MatrixForm} // Simplify;
```

Derivation of the vector parallel transport equation along a curve in an N-dimensional manifold

```
Print["Parallel transport of vector λ is intended as (λ̇)T = 0, with T = tangent space."]
Print["λ̇ = (λ̇)T + (λ̇)N"]
TotalD[Tensor[λ[u]], u] == 0;
MapAt[Subscript[#, T] &, %, 1] // FrameBox // DisplayForm
Print["Substituting tensor expression for λ[u] and evaluating"]
%%% /. Tensor[λ[u]] → NestedTensor[ed[b] λu[b]];
MapAt[Subscript[#, T] &, %, 1]
%% // UnnestTensor;
MapAt[Subscript[#, T] &, %, 1]
Print["Expanding the total derivative of the basis vector"]
MapAt[ExpandTotalD[labs, c], %%%, {1, 1}];
MapAt[Subscript[#, T] &, %, 1] // TraditionalForm
Print["Projection into T and expanding the partial
   derivative of the basis vector in terms of Christoffel symbols"]
%%% /. PartialD[labs][ed[i_], xu[j_]] → Γudd[a, i, j] ed[a]
Print["Reindex, factor and introduce zero vector"]
MapAt[(# // SimplifyTensorSum // Factor) &, %%, 1]
Part[%, 1, 2] == zerou[a] // FrameBox // DisplayForm
Print["Compare with eqn[2, 23], p.65."]
```

Parallel transport of vector λ is intended as $(\dot{\lambda})_T = 0$, with T = tangent space.

$\dot{\lambda} = (\dot{\lambda})_T + (\dot{\lambda})_N$

$$\boxed{\left(\frac{d\lambda[u]}{du}\right)_T == 0}$$

Substituting tensor expression for λ[u] and evaluating

$$\left(\frac{d\left(e_b\,\lambda^b\right)}{du}\right)_T == 0$$

$$\left(\lambda^b\,\frac{de_b}{du} + e_b\,\frac{d\lambda^b}{du}\right)_T == 0$$

Expanding the total derivative of the basis vector

$$\left(e_b\,\frac{d\lambda^b}{du} + \lambda^b\,\frac{dx^c}{du}\,\frac{\partial e_b}{\partial x^c}\right)_T = 0$$

Projection into T and expanding the partial
derivative of the basis vector in terms of Christoffel symbols

$$e_a\,\Gamma^a{}_{bc}\,\lambda^b\,\frac{dx^c}{du} + e_b\,\frac{d\lambda^b}{du} == 0$$

Reindex, factor and introduce zero vector

$$e_a\left(\Gamma^a{}_{bc}\,\lambda^b\,\frac{dx^c}{du} + \frac{d\lambda^a}{du}\right) == 0$$

$$\boxed{\Gamma^a{}_{bc}\,\lambda^b\,\frac{dx^c}{du} + \frac{d\lambda^a}{du} == \text{zero}^a}$$

Compare with eqn[2, 23], p.65.

■ Complete expansion of the vector parallel transport equation in 2D

```
DeclareBaseIndices[Range[2]]
```

```
AbsoluteD[λu[a], u] == zerou[a]
ExpandAbsoluteD[{x, δ, g, Γ}, {c, b}][%]
% /. Tensor[Γ, {a_, Void, Void}, {Void, b_, c_}] →
  1/2 guu[a, d] (PartialD[gdd[d, c], b] + PartialD[gdd[b, d], c] - PartialD[gdd[b, c], d])
EinsteinSum[][%%] // SymmetrizeSlots[] // ToArrayValues[];
Collect[#, {Tensor[Γ, {a_, Void, Void}, {Void, b_, c_}]}] & //@ % // TableForm
```

$$\frac{D\lambda^a}{du} == \text{zero}^a$$

$$\Gamma^a{}_{cb}\,\lambda^b\,\frac{dx^c}{du} + \frac{d\lambda^a}{du} == \text{zero}^a$$

$$\frac{1}{2}\,g^{ad}\left(-g_{cb,d} + g_{cd,b} + g_{db,c}\right)\lambda^b\,\frac{dx^c}{du} + \frac{d\lambda^a}{du} == \text{zero}^a$$

$$\Gamma^1{}_{11}\,\lambda^1\,\frac{dx^1}{du} + \Gamma^1{}_{22}\,\lambda^2\,\frac{dx^2}{du} + \Gamma^1{}_{12}\left(\lambda^2\,\frac{dx^1}{du} + \lambda^1\,\frac{dx^2}{du}\right) + \frac{d\lambda^1}{du} == 0$$

$$\Gamma^2{}_{11}\,\lambda^1\,\frac{dx^1}{du} + \Gamma^2{}_{22}\,\lambda^2\,\frac{dx^2}{du} + \Gamma^2{}_{12}\left(\lambda^2\,\frac{dx^1}{du} + \lambda^1\,\frac{dx^2}{du}\right) + \frac{d\lambda^2}{du} == 0$$

Definitions used in the next two sections:

```
DeclareBaseIndices[{1, 2}]
SetAttributes[{a}, Constant]
r[θ_, ϕ_] = {a Cos[ϕ] Sin[θ], a Sin[ϕ] Sin[θ], a Cos[θ]};
{eθ[θ_, ϕ_], eϕ[θ_, ϕ_]} = {∂θ%, ∂ϕ%};
%.Transpose[%] // Simplify;
```

```
metric = % // CoordinatesToTensors[{θ, ϕ}, x];
SetMetricValueRules[g, metric]
SetChristoffelValueRules[xu[i], metric, Γ]
```

Example 2.2.1 p. 66-68. vector parallel transport on a 2D-sphere

```
SetAttributes[{θ0, ω, α}, Constant]
```

```
Print["Vector parallel transport equation on a sphere with radius a"]
eqns = TotalD[λu[m], t] + Γudd[m, n, s] λu[n] TotalD[xu[s], t] == zerou[m]
% // ToArrayValues[] // TableForm
Print["...and along a circle of latitude θ0 for vector λ[t] with I.C. on λ[0]"]
SetTensorValueRules[xu[i], {θ0, t}]
xu[i] // ST
eqns = (eqns // ToArrayValues[]) /. λu[i_] → λu[i][t];
conds = {λu[1][0] == Cos[α] / a, λu[2][0] == Sin[α] / (a Sin[θ0])};
eqns ⋃ conds // TableForm // FrameBox // DisplayForm
sols = DSolve[eqns ⋃ conds, {λu[1], λu[2]}, t][[1]] /. Cos[θ0] → ω;
Print["Solution λ^i(t) with the given I.C. and ω=Cos[θ0] is"]
{λu[1][t_], λu[2][t_]} = {λu[1][t], λu[2][t]} /. sols // Simplify
```

Vector parallel transport equation on a sphere with radius a

$$\Gamma^{m}{}_{ns}\,\lambda^{n}\,\frac{dx^{s}}{dt}+\frac{d\lambda^{m}}{dt}==\text{zero}^{m}$$

$$-\text{Cos}[x^1]\,\text{Sin}[x^1]\,\lambda^2\,\frac{dx^2}{dt}+\frac{d\lambda^1}{dt}==0$$

$$\text{Cot}[x^1]\,\lambda^2\,\frac{dx^1}{dt}+\text{Cot}[x^1]\,\lambda^1\,\frac{dx^2}{dt}+\frac{d\lambda^2}{dt}==0$$

...and along a circle of latitude θ0 for vector λ[t] with I.C. on λ[0]

Tensor $\{x^1, x^2\}$ is {θ0, t}

$$\lambda^1[0]==\frac{\text{Cos}[\alpha]}{a}$$

$$\lambda^2[0]==\frac{\text{Csc}[\theta0]\,\text{Sin}[\alpha]}{a}$$

$$-\text{Cos}[\theta0]\,\text{Sin}[\theta0]\,\lambda^2[t]+\lambda^{1\prime}[t]==0$$

$$\text{Cot}[\theta0]\,\lambda^1[t]+\lambda^{2\prime}[t]==0$$

Solution $\lambda^i(t)$ with the given I.C. and ω=Cos[θ0] is

$$\left\{\frac{\text{Cos}[\alpha-t\,\omega]}{a},\ \frac{\text{Csc}[\theta0]\,\text{Sin}[\alpha-t\,\omega]}{a}\right\}$$

Let's look at λ[t], λ'[t] and λ''[t] in the Cartesian coordinates of the embedding 3D space and in the natural basis.

```
Print["λ[t] in 3D kartesischen Koordinaten und seine Länge"]
λfield[t_] = λu[1][t] e_θ[θ0, t] + λu[2][t] e_ϕ[θ0, t]
Sqrt[λfield[t].λfield[t] // Simplify]
```

λ[t] in 3D kartesischen Koordinaten und seine Länge

{Cos[t] Cos[θ0] Cos[α − t ω] − Sin[t] Sin[α − t ω],
 Cos[θ0] Cos[α − t ω] Sin[t] + Cos[t] Sin[α − t ω], −Cos[α − t ω] Sin[θ0]}

1

```
Print["λ'[t] in 3D kartesischen Koordinaten und in der lokalen Basis:"]
D1λfield[t_] = Dt[λfield[t], t] /. ω → Cos[θ0] // Simplify
```

```
{D1λfield[t].eθ[θ0, t], D1λfield[t].eϕ[θ0, t],
  D1λfield[t].eθ[θ0, t]×eϕ[θ0, t]} // Simplify
```

```
λ'[t] in 3D kartesischen Koordinaten und in der lokalen Basis:
```

$$\{-\mathrm{Cos}[t]\,\mathrm{Sin}[\theta0]^2\,\mathrm{Sin}[\alpha - t\,\mathrm{Cos}[\theta0]],\ -\mathrm{Sin}[t]\,\mathrm{Sin}[\theta0]^2\,\mathrm{Sin}[\alpha - t\,\mathrm{Cos}[\theta0]],\ -\mathrm{Cos}[\theta0]\,\mathrm{Sin}[\theta0]\,\mathrm{Sin}[\alpha - t\,\mathrm{Cos}[\theta0]]\}$$

$$\{0,\ 0,\ -a^2\,\mathrm{Sin}[\theta0]^2\,\mathrm{Sin}[\alpha - t\,\mathrm{Cos}[\theta0]]\}$$

```
Print["λ''[t] in 3D kartesischen Koordinaten und in der lokalen Basis:"]
D2λfield[t_] = Dt[λfield[t], {t, 2}] /. ω → Cos[θ0] // Simplify
{D2λfield[t].eθ[θ0, t], D2λfield[t].eϕ[θ0, t],
  D2λfield[t].eθ[θ0, t]×eϕ[θ0, t]} // Simplify
```

```
λ''[t] in 3D kartesischen Koordinaten und in der lokalen Basis:
```

$$\{\mathrm{Sin}[\theta0]^2\,(\mathrm{Cos}[t]\,\mathrm{Cos}[\theta0]\,\mathrm{Cos}[\alpha - t\,\mathrm{Cos}[\theta0]] + \mathrm{Sin}[t]\,\mathrm{Sin}[\alpha - t\,\mathrm{Cos}[\theta0]]),\ \mathrm{Sin}[\theta0]^2\,(\mathrm{Cos}[\theta0]\,\mathrm{Cos}[\alpha - t\,\mathrm{Cos}[\theta0]]\,\mathrm{Sin}[t] - \mathrm{Cos}[t]\,\mathrm{Sin}[\alpha - t\,\mathrm{Cos}[\theta0]]),\ \mathrm{Cos}[\theta0]^2\,\mathrm{Cos}[\alpha - t\,\mathrm{Cos}[\theta0]]\,\mathrm{Sin}[\theta0]\}$$

$$\{0,\ -a\,\mathrm{Sin}[\theta0]^3\,\mathrm{Sin}[\alpha - t\,\mathrm{Cos}[\theta0]],\ a^2\,\mathrm{Cos}[\theta0]\,\mathrm{Cos}[\alpha - t\,\mathrm{Cos}[\theta0]]\,\mathrm{Sin}[\theta0]^2\}$$

The series expansion of λ[t] − λ[0] reveals it's $O[t]^2$ behavior in T, as expected.

```
Print["Serie in t von λ[t] in 3D kartesischen Koordinaten:"]
Series[λfield[t], {t, 0, 1}] /. ω → Cos[θ0] // Simplify
Print["Serie in t von λ[t]-λ[0] in 3D kartesischen Koordinaten:"]
%% - λfield[0];
{%.eθ[θ0, t], %.eϕ[θ0, t], %.eθ[θ0, t]×eϕ[θ0, t]} // Simplify
Print["Serie in t von {λu[1][t],λu[2][t]}:"]
Series[{λu[1][t], λu[2][t]}, {t, 0, 1}] /. ω → Cos[θ0] // Simplify
```

```
Serie in t von λ[t] in 3D kartesischen Koordinaten:
```

$$\{\mathrm{Cos}[\alpha]\,\mathrm{Cos}[\theta0] - \mathrm{Sin}[\alpha]\,\mathrm{Sin}[\theta0]^2\,t + O[t]^2,\ \mathrm{Sin}[\alpha] + O[t]^2,\ -\mathrm{Cos}[\alpha]\,\mathrm{Sin}[\theta0] - \mathrm{Cos}[\theta0]\,\mathrm{Sin}[\alpha]\,\mathrm{Sin}[\theta0]\,t + O[t]^2\}$$

```
Serie in t von λ[t]-λ[0] in 3D kartesischen Koordinaten:
```

$$\{O[t]^2,\ O[t]^2,\ -a^2\,\mathrm{Sin}[\alpha]\,\mathrm{Sin}[\theta0]^2\,t + O[t]^2\}$$

```
Serie in t von {λu[1][t],λu[2][t]}:
```

$$\left\{\frac{\mathrm{Cos}[\alpha]}{a} + \frac{\mathrm{Cos}[\theta0]\,\mathrm{Sin}[\alpha]\,t}{a} + O[t]^2,\ \frac{\mathrm{Csc}[\theta0]\,\mathrm{Sin}[\alpha]}{a} - \frac{\mathrm{Cos}[\alpha]\,\mathrm{Cot}[\theta0]\,t}{a} + O[t]^2\right\}$$

Twist of the vector induced by the parallel transport at latitude 85° and 5°:

```
λu[i] // ST
SetTensorValueRules[λu[i], {λu[1][t], λu[2][t]}]
λu[i] // ST
SetTensorValueRules[λ0u[i], {λu[1][0], λu[2][0]}]
λ0u[i] // ST
gdd[i, j] λu[i] λu[j]
% // ToArrayValues[] // Simplify
Print["cos(Δα)=f(t,θ0)"]
gdd[i, j] λu[i] λ0u[j]
(% // ToArrayValues[] // Simplify) /. ω → Cos[θ0]
N[{2 π Cos[85 °], 2 π Cos[5 °]} / °, 5] "°"
360 "°" - %[[2]]
```

```
Tensor {λ^1, λ^2} is {λ^1, λ^2}
```

$$\text{Tensor } \{\lambda^1, \lambda^2\} \text{ is } \left\{\frac{\text{Cos}[\alpha - t\,\omega]}{a}, \frac{\text{Csc}[\theta 0]\,\text{Sin}[\alpha - t\,\omega]}{a}\right\}$$

$$\text{Tensor } \{\lambda 0^1, \lambda 0^2\} \text{ is } \left\{\frac{\text{Cos}[\alpha]}{a}, \frac{\text{Csc}[\theta 0]\,\text{Sin}[\alpha]}{a}\right\}$$

$g_{ij}\,\lambda^i\,\lambda^j$

1

cos(Δα)=f(t,θ0)

$g_{ij}\,\lambda^i\,\lambda 0^j$

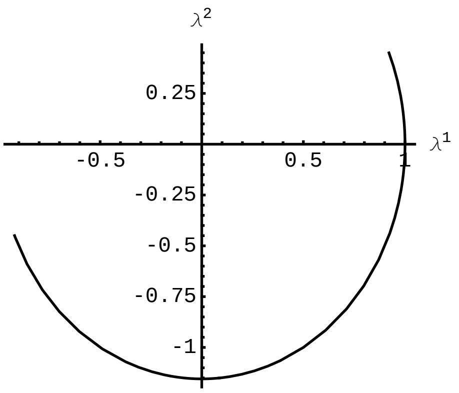
Cos[t Cos[θ0]]

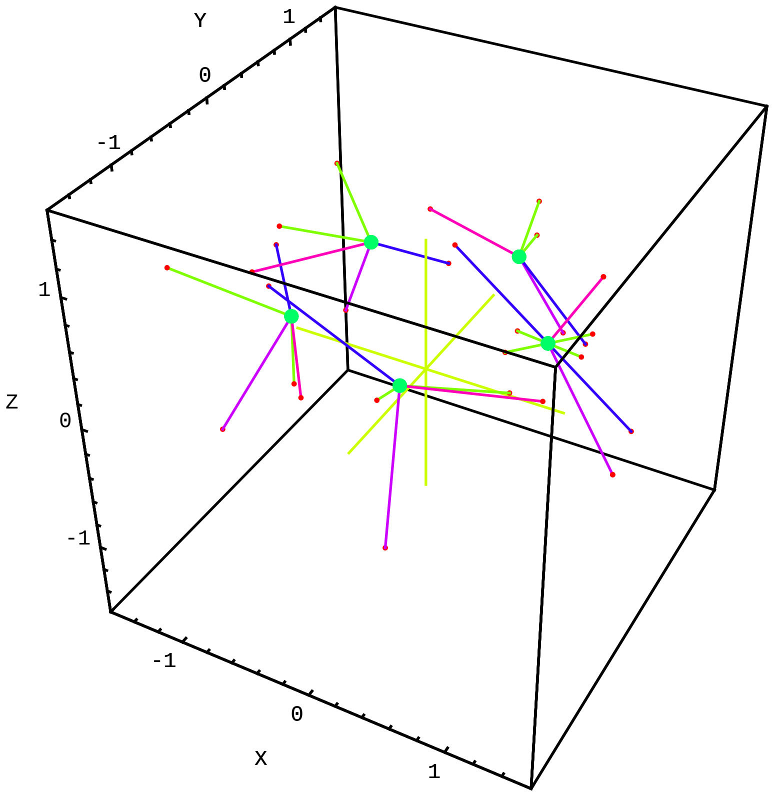
{31.376 °, 358.63 °}

1.37 °

A graphic representation of the vector parallel transport on a sphere.

```
Block[{α = .4, θ0 = π / 3, ω = Cos[θ0], a = 1.},
 range = {t, 0, 2 π, 2 π / 5};

 ParametricPlot[{λu[1][t], λu[2][t]}, Evaluate[Most[range]],
  AxesLabel → {"λ¹", "λ²"}, AspectRatio → Automatic, ImageSize → 72 × 3];

 λfieldPlot = Table[victor[r[θ0, t], λfield[t]], Evaluate[range]];
 D1λfieldPlot = Table[victor[r[θ0, t], D1λfield[t], Hue[.25]], Evaluate[range]];
 D2λfieldPlot = Table[victor[r[θ0, t], D2λfield[t], Hue[.25]], Evaluate[range]];
 naturalbasisPlot = Table[{
     victor[r[θ0, t], eθ[θ0, t], Hue[.80]], victor[r[θ0, t], eϕ[θ0, t], Hue[.88]]},
    Evaluate[range]]; sphereaxisPlot = Graphics3D[{Hue[.2], Line[a {{1, 0, 0}, {-1, 0, 0}}],
     Line[a {{0, 1, 0}, {0, -1, 0}}], Line[a {{0, 0, 1}, {0, 0, -1}}]}];
 Show[sphereaxisPlot, naturalbasisPlot,
  λfieldPlot, D1λfieldPlot, D2λfieldPlot,
  Axes → True, AxesLabel → {"X", "Y", "Z"},
  PlotRange → 1.6 {{-1, 1}, {-1, 1}, {-1, 1}}, ImageSize → 72 × 5];
]
```





★ Parallel transport on a sphere: λ, λ', λ'' and e_i.★

```
Export["C:\\USERS\\LUMA\\Mathematica\\_Tensorrechnung\\2.2
   - Parallel transport on a sphere.dxf", %, "DXF"]
```

```
C:\USERS\LUMA\Mathematica\_Tensorrechnung\2.2 - Parallel transport on a sphere.dxf
```

■ Connection coefficients (when metric-induced: Christoffel symbols) on a 2D-sphere

```
Print["Christoffel symbols of the first kind are defined by"]
Γddd[a, b, c] == ChristoffelDownRule[[2]]
Print["Christoffel symbols of the second kind"]
Γudd[a, b, c] ==
 1/2 guu[a, d] (PartialD[gdd[d, c], b] + PartialD[gdd[b, d], c] - PartialD[gdd[b, c], d])
gdd[a, d] Γudd[d, b, c];
% == (% // MetricSimplify[g])
```

Christoffel symbols of the first kind are defined by

$$\Gamma_{abc} == \frac{1}{2}\left(g_{ac,b} + g_{ba,c} - g_{bc,a}\right)$$

Christoffel symbols of the second kind

$$\Gamma^{a}{}_{bc} == \frac{1}{2}\, g^{ad}\left(-g_{bc,d} + g_{bd,c} + g_{dc,b}\right)$$

$$g_{ad}\,\Gamma^{d}{}_{bc} == \Gamma_{abc}$$

Let's test the geometrical meaning of the connection coefficients as the projection of $e_{i,j}$ into T: $\left(e_{i,j}\right)_T = \Gamma^{k}{}_{ij}\, e_k$.

```
d1e1[θ_, ϕ_] = ∂θ eθ[θ, ϕ]; d2e1[θ_, ϕ_] = ∂ϕ eθ[θ, ϕ];
d1e2[θ_, ϕ_] = ∂θ eϕ[θ, ϕ]; d2e2[θ_, ϕ_] = ∂ϕ eϕ[θ, ϕ];
Print["∂j ei and ∂j ei//nbProj"]
aa = d1e1[θ, ϕ] // nbProj
```

```
bb = d2e1[θ, ϕ] // nbProj
cc = d1e2[θ, ϕ] // nbProj
dd = d2e2[θ, ϕ] // nbProj
bb == cc
```

∂ⱼeᵢ and ∂ⱼeᵢ//nbProj

$\left\{v=, \begin{pmatrix} -a\,\mathrm{Cos}[\phi]\,\mathrm{Sin}[\theta] \\ -a\,\mathrm{Sin}[\theta]\,\mathrm{Sin}[\phi] \\ -a\,\mathrm{Cos}[\theta] \end{pmatrix}, \mathrm{nbProj}(v)=, \begin{pmatrix} 0 \\ 0 \\ -a^3\,\mathrm{Sin}[\theta] \end{pmatrix}\right\}$

$\left\{v=, \begin{pmatrix} -a\,\mathrm{Cos}[\theta]\,\mathrm{Sin}[\phi] \\ a\,\mathrm{Cos}[\theta]\,\mathrm{Cos}[\phi] \\ 0 \end{pmatrix}, \mathrm{nbProj}(v)=, \begin{pmatrix} 0 \\ a^2\,\mathrm{Cos}[\theta]\,\mathrm{Sin}[\theta] \\ 0 \end{pmatrix}\right\}$

$\left\{v=, \begin{pmatrix} -a\,\mathrm{Cos}[\theta]\,\mathrm{Sin}[\phi] \\ a\,\mathrm{Cos}[\theta]\,\mathrm{Cos}[\phi] \\ 0 \end{pmatrix}, \mathrm{nbProj}(v)=, \begin{pmatrix} 0 \\ a^2\,\mathrm{Cos}[\theta]\,\mathrm{Sin}[\theta] \\ 0 \end{pmatrix}\right\}$

$\left\{v=, \begin{pmatrix} -a\,\mathrm{Cos}[\phi]\,\mathrm{Sin}[\theta] \\ -a\,\mathrm{Sin}[\theta]\,\mathrm{Sin}[\phi] \\ 0 \end{pmatrix}, \mathrm{nbProj}(v)=, \begin{pmatrix} -a^2\,\mathrm{Cos}[\theta]\,\mathrm{Sin}[\theta] \\ 0 \\ -a^3\,\mathrm{Sin}[\theta]^3 \end{pmatrix}\right\}$

True

```
Print["Γ"]
Drop[NonzeroValueRules[Γ] // UseCoordinates[{θ, ϕ}, u], -3]
{Γ[1, 1, 1] = 0, Γ[1, 1, 2] = 0, Γ[1, 2, 1] = Γ[1, 1, 2], Γ[1, 2, 2] = -Cos[θ] Sin[θ],
   Γ[2, 1, 1] = 0, Γ[2, 1, 2] = Cot[θ], Γ[2, 2, 1] = Γ[2, 1, 2], Γ[2, 2, 2] = 0} // TableForm //
 HoldForm
Print["Γᵏᵢⱼeₖ and Γᵏᵢⱼeₖ//nbProj"]
Γ[1, 1, 1] eθ[θ, ϕ] + Γ[2, 1, 1] eϕ[θ, ϕ] // nbProj
Most[#[[4, 1]]] &/@ (aa == %)
Γ[1, 1, 2] eθ[θ, ϕ] + Γ[2, 1, 2] eϕ[θ, ϕ] // nbProj
Most[#[[4, 1]]] &/@ (bb == %)
Γ[1, 2, 1] eθ[θ, ϕ] + Γ[2, 2, 1] eϕ[θ, ϕ] // nbProj
Most[#[[4, 1]]] &/@ (cc == %)
Γ[1, 2, 2] eθ[θ, ϕ] + Γ[2, 2, 2] eϕ[θ, ϕ] // nbProj
Most[#[[4, 1]]] &/@ (dd == %)
```

Γ

$\{\Gamma^1{}_{22} \to -\mathrm{Cos}[x^1]\,\mathrm{Sin}[x^1], \Gamma^2{}_{12} \to \mathrm{Cot}[x^1], \Gamma^2{}_{21} \to \mathrm{Cot}[x^1]\}$

```
Γ[1, 1, 1] = 0
Γ[1, 1, 2] = 0
Γ[1, 2, 1] = Γ[1, 1, 2]
Γ[1, 2, 2] = -Cos[θ] Sin[θ]
Γ[2, 1, 1] = 0
Γ[2, 1, 2] = Cot[θ]
Γ[2, 2, 1] = Γ[2, 1, 2]
Γ[2, 2, 2] = 0
```

Γᵏᵢⱼeₖ and Γᵏᵢⱼeₖ//nbProj

$\left\{v=, \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \mathrm{nbProj}(v)=, \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}\right\}$

True

$\left\{v=, \begin{pmatrix} -a\,\mathrm{Cos}[\theta]\,\mathrm{Sin}[\phi] \\ a\,\mathrm{Cos}[\theta]\,\mathrm{Cos}[\phi] \\ 0 \end{pmatrix}, \mathrm{nbProj}(v)=, \begin{pmatrix} 0 \\ a^2\,\mathrm{Cos}[\theta]\,\mathrm{Sin}[\theta] \\ 0 \end{pmatrix}\right\}$

True

$$\left\{v=, \begin{pmatrix} -a \cos[\theta] \sin[\phi] \\ a \cos[\theta] \cos[\phi] \\ 0 \end{pmatrix}, \text{nbProj}(v)=, \begin{pmatrix} 0 \\ a^2 \cos[\theta] \sin[\theta] \\ 0 \end{pmatrix}\right\}$$

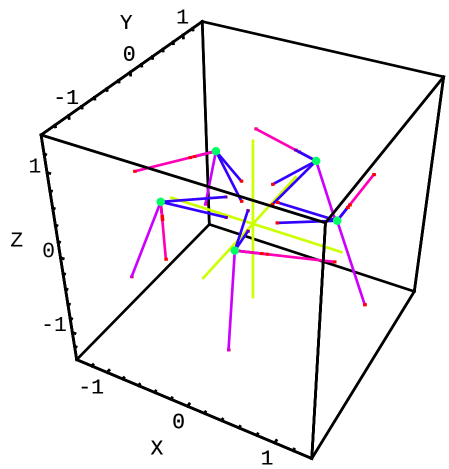
True

$$\left\{v=, \begin{pmatrix} -a \cos[\theta]^2 \cos[\phi] \sin[\theta] \\ -a \cos[\theta]^2 \sin[\theta] \sin[\phi] \\ a \cos[\theta] \sin[\theta]^2 \end{pmatrix}, \text{nbProj}(v)=, \begin{pmatrix} -a^2 \cos[\theta] \sin[\theta] \\ 0 \\ 0 \end{pmatrix}\right\}$$

True

A graphic representation of the natural basis e_i and the derivatives $e_{i,j}$.

```
Block[{α = .4, θ0 = 1.2, ω = Cos[θ0], a = 1., f = .7},
 range = {t, 0, 2 π, 2 π / 5};
 naturalbasisPlot = Table[{
    victor[r[θ0, t], eθ[θ0, t], Hue[.80]],
    victor[r[θ0, t], eϕ[θ0, t], Hue[.88]]}, Evaluate[range]];
 d1e1Plot = Table[victor[r[θ0, t], d1e1[θ0, t] f], Evaluate[range]];
 d2e1Plot = Table[victor[r[θ0, t], d2e1[θ0, t] f], Evaluate[range]];
 d1e2Plot = Table[victor[r[θ0, t], d2e1[θ0, t] f 1.2 ], Evaluate[range]];
 d2e2Plot = Table[victor[r[θ0, t], d2e2[θ0, t] f], Evaluate[range]];
 sphereaxisPlot = Graphics3D[{Hue[.2], Line[a {{1, 0, 0}, {-1, 0, 0}}],
    Line[a {{0, 1, 0}, {0, -1, 0}}], Line[a {{0, 0, 1}, {0, 0, -1}}]}];
 Show[sphereaxisPlot, naturalbasisPlot,
  d1e1Plot, d2e1Plot, d1e2Plot, d2e2Plot,
  Axes → True, AxesLabel → {"X", "Y", "Z"},
  PlotRange → 1.4 {{-1, 1}, {-1, 1}, {-1, 1}}, ImageSize → {8 × 72, 3 × 72}];
]
```



★ e_i and $e_{i,j}$ on a sphere. ★

```
Export[
 "C:\\USERS\\LUMA\\Mathematica\\_Tensorrechnung\\2.2 - Connection coefficients.dxf",
 %, "DXF"]
```

C:\USERS\LUMA\Mathematica_Tensorrechnung\2.2 - Connection coefficients.dxf

2.3 Absolute and covariant differentiation p. 71 - 79

```
Needs["TensorCalculus3`Tensorial`"]
$PrePrint =.
DeclareBaseIndices[{1, 2, 3, 4}]
labs = {x, δ, g, Γ};
DefineTensorShortcuts[
 {{x, λ, μ}, 1},
 {{δ, g, τ}, 2},
 {{Γ}, 3}]
SetTensorValues[δud[i, j], IdentityMatrix[NDim]]
DeclareTensorSymmetries[Γ, 3, {1, {2, 3}}]
TensorList = {Tensor[ϕ], λu[a], λd[a], τuu[a, b], τdd[a, b], τud[a, b], τdu[a, b]};
```

Absolute differentiation

```
(aa = AbsoluteD[NestedTensor[#], u];
   aa == (aa // UnnestTensor // ExpandAbsoluteD[labs, {i, j}])) & /@ TensorList // TableForm
Print["last expression fully expanded:"]
Collect[#, {TotalD[Tensor[x, List[_], List[Void]], u]}] & /@ EinsteinSum[][Last[%%]] //
 SymmetrizeSlots[]
```

$$\frac{D\phi}{du} == \frac{d\phi}{du}$$

$$\frac{D\lambda^a}{du} == \Gamma^a{}_{ij}\,\lambda^j\,\frac{dx^i}{du} + \frac{d\lambda^a}{du}$$

$$\frac{D\lambda_a}{du} == -\Gamma^j{}_{ia}\,\lambda_j\,\frac{dx^i}{du} + \frac{d\lambda_a}{du}$$

$$\frac{D\tau^{ab}}{du} == \left(\Gamma^b{}_{ij}\,\tau^{aj} + \Gamma^a{}_{ij}\,\tau^{jb}\right)\frac{dx^i}{du} + \frac{d\tau^{ab}}{du}$$

$$\frac{D\tau_{ab}}{du} == \left(-\Gamma^j{}_{ib}\,\tau_{aj} - \Gamma^j{}_{ia}\,\tau_{jb}\right)\frac{dx^i}{du} + \frac{d\tau_{ab}}{du}$$

$$\frac{D\tau^a{}_b}{du} == \left(-\Gamma^j{}_{ib}\,\tau^a{}_j + \Gamma^a{}_{ij}\,\tau^j{}_b\right)\frac{dx^i}{du} + \frac{d\tau^a{}_b}{du}$$

$$\frac{D\tau_a{}^b}{du} == \left(-\Gamma^j{}_{ia}\,\tau_j{}^b + \Gamma^b{}_{ij}\,\tau_a{}^j\right)\frac{dx^i}{du} + \frac{d\tau_a{}^b}{du}$$

last expression fully expanded:

$$\frac{D\tau_a{}^b}{du} == \left(\Gamma^b{}_{11}\,\tau_a{}^1 + \Gamma^b{}_{12}\,\tau_a{}^2 + \Gamma^b{}_{13}\,\tau_a{}^3 + \Gamma^b{}_{14}\,\tau_a{}^4 - \Gamma^1{}_{1a}\,\tau_1{}^b - \Gamma^2{}_{1a}\,\tau_2{}^b - \Gamma^3{}_{1a}\,\tau_3{}^b - \Gamma^4{}_{1a}\,\tau_4{}^b\right)\frac{dx^1}{du} +$$
$$\left(\Gamma^b{}_{12}\,\tau_a{}^1 + \Gamma^b{}_{22}\,\tau_a{}^2 + \Gamma^b{}_{23}\,\tau_a{}^3 + \Gamma^b{}_{24}\,\tau_a{}^4 - \Gamma^1{}_{2a}\,\tau_1{}^b - \Gamma^2{}_{2a}\,\tau_2{}^b - \Gamma^3{}_{2a}\,\tau_3{}^b - \Gamma^4{}_{2a}\,\tau_4{}^b\right)\frac{dx^2}{du} +$$
$$\left(\Gamma^b{}_{13}\,\tau_a{}^1 + \Gamma^b{}_{23}\,\tau_a{}^2 + \Gamma^b{}_{33}\,\tau_a{}^3 + \Gamma^b{}_{34}\,\tau_a{}^4 - \Gamma^1{}_{3a}\,\tau_1{}^b - \Gamma^2{}_{3a}\,\tau_2{}^b - \Gamma^3{}_{3a}\,\tau_3{}^b - \Gamma^4{}_{3a}\,\tau_4{}^b\right)\frac{dx^3}{du} +$$
$$\left(\Gamma^b{}_{14}\,\tau_a{}^1 + \Gamma^b{}_{24}\,\tau_a{}^2 + \Gamma^b{}_{34}\,\tau_a{}^3 + \Gamma^b{}_{44}\,\tau_a{}^4 - \Gamma^1{}_{4a}\,\tau_1{}^b - \Gamma^2{}_{4a}\,\tau_2{}^b - \Gamma^3{}_{4a}\,\tau_3{}^b - \Gamma^4{}_{4a}\,\tau_4{}^b\right)\frac{dx^4}{du} + \frac{d\tau_a{}^b}{du}$$

```
AbsoluteD[λu[a], u];
% == (% // ExpandAbsoluteD[labs, {i, j}])
MapAt[ExpandTotalD[labs, i], %, 2]
MapAt[Factor, %, 2]
Print["in FullForm:"]
%% // FullForm
```

$$\frac{D\lambda^a}{du} == \Gamma^a{}_{ij}\,\lambda^j\,\frac{dx^i}{du} + \frac{d\lambda^a}{du}$$

$$\frac{D\lambda^a}{du} == \Gamma^a{}_{ij}\,\lambda^j\,\frac{dx^i}{du} + \frac{dx^i}{du}\,\partial_{x^i}\lambda^a$$

$$\frac{D\lambda^a}{du} == \frac{dx^i}{du}\left(\Gamma^a{}_{ij}\,\lambda^j + \partial_{x^i}\lambda^a\right)$$

in FullForm:

```
Equal[AbsoluteD[Tensor[\[Lambda], List[a], List[Void]], u],
 Times[TotalD[Tensor[x, List[i], List[Void]], u],
  Plus[Times[Tensor[\[CapitalGamma], List[a,Void,Void], List[Void,i,j]], Tensor[
     \[Lambda], List[j], List[Void]]], PartialD[List[x, \[Delta], g, \[CapitalGamma]]][
    Tensor[\[Lambda], List[a], List[Void]], Tensor[x, List[i], List[Void]]]]]]
```

- If two vector fields λ and μ are parallel transported along a curve γ, then their inner product remains constant along the curve.

```
Print["The inner product λ·μ"]
gdd[a, b] λu[a] μu[b]
Print["is a scalar field, hence total derivative = absolute derivative:"]
TotalD[NestedTensor[%%], u]
AbsoluteD[NestedTensor[%%%], u]
UnnestTensor[%]
{AbsoluteD[gdd[_, _], u] → 0, AbsoluteD[λu[_], u] → 0, AbsoluteD[μu[_], u] → 0};
Print["Using the properties of metric and parallel transport ", %, ", we get:"]
%%% /. %%
Print["Hence λ·μ = const."]
```

The inner product λ·μ

$$g_{ab}\,\lambda^a\,\mu^b$$

is a scalar field, hence total derivative = absolute derivative:

$$\frac{d\left(g_{ab}\,\lambda^a\,\mu^b\right)}{du}$$

$$\frac{D\left(g_{ab}\,\lambda^a\,\mu^b\right)}{du}$$

$$\frac{Dg_{ab}}{du}\,\lambda^a\,\mu^b + g_{ab}\left(\frac{D\mu^b}{du}\,\lambda^a + \frac{D\lambda^a}{du}\,\mu^b\right)$$

Using the properties of metric and parallel transport $\left\{\frac{Dg\_\_}{du} \to 0, \frac{D\lambda\_}{du} \to 0, \frac{D\mu\_}{du} \to 0\right\}$, we get:

0

Hence λ·μ = const.

Covariant differentiation

```
{PartialD[λu[a], i], PartialD[λu[a], {i, j}], CovariantD[λu[a], i],
 CovariantD[λu[a], {i, j}], TotalD[λu[a], u], AbsoluteD[λu[a], u]}
CovariantD[λu[a], i];
% == (% // ExpandCovariantD[labs, j])
CovariantD[λu[a], {i, j}];
% == (% // ExpandCovariantD[labs, {m, n}])
```

$$\left\{\lambda^{a}{}_{,i},\ \lambda^{a}{}_{,i,j},\ \lambda^{a}{}_{;i},\ \lambda^{a}{}_{;i;j},\ \frac{d\lambda^{a}}{du},\ \frac{D\lambda^{a}}{du}\right\}$$

$$\lambda^{a}{}_{;i} == \Gamma^{a}{}_{ij}\,\lambda^{j} + \partial_{x^{i}}\lambda^{a}$$

$$\lambda^{a}{}_{;i;j} == \lambda^{m}\,\partial_{x^{j}}\Gamma^{a}{}_{im} + \partial_{x^{i},x^{j}}\lambda^{a} - \Gamma^{n}{}_{ji}\left(\Gamma^{a}{}_{nm}\,\lambda^{m} + \partial_{x^{n}}\lambda^{a}\right) + \Gamma^{a}{}_{im}\,\partial_{x^{j}}\lambda^{m} + \Gamma^{a}{}_{jn}\left(\Gamma^{n}{}_{im}\,\lambda^{m} + \partial_{x^{i}}\lambda^{n}\right)$$

```
(aa = CovariantD[NestedTensor[#], i];
    aa == (aa // UnnestTensor // ExpandCovariantD[labs, {j}])) & /@ TensorList // TableForm
Print["last expression fully expanded:"]
EinsteinSum[][Last[%%]]
```

$$\phi_{;i} == \phi_{,i}$$
$$\lambda^{a}{}_{;i} == \Gamma^{a}{}_{ij}\,\lambda^{j} + \partial_{x^{i}}\lambda^{a}$$
$$\lambda_{a;i} == -\Gamma^{j}{}_{ia}\,\lambda_{j} + \partial_{x^{i}}\lambda_{a}$$
$$\tau^{ab}{}_{;i} == \Gamma^{b}{}_{ij}\,\tau^{aj} + \Gamma^{a}{}_{ij}\,\tau^{jb} + \partial_{x^{i}}\tau^{ab}$$
$$\tau_{ab;i} == -\Gamma^{j}{}_{ib}\,\tau_{aj} - \Gamma^{j}{}_{ia}\,\tau_{jb} + \partial_{x^{i}}\tau_{ab}$$
$$\tau^{a}{}_{b;i} == -\Gamma^{j}{}_{ib}\,\tau^{a}{}_{j} + \Gamma^{a}{}_{ij}\,\tau^{j}{}_{b} + \partial_{x^{i}}\tau^{a}{}_{b}$$
$$\tau_{a}{}^{b}{}_{;i} == -\Gamma^{j}{}_{ia}\,\tau_{j}{}^{b} + \Gamma^{b}{}_{ij}\,\tau_{a}{}^{j} + \partial_{x^{i}}\tau_{a}{}^{b}$$

last expression fully expanded:

$$\tau_{a}{}^{b}{}_{;i} == \Gamma^{b}{}_{i1}\,\tau_{a}{}^{1} + \Gamma^{b}{}_{i2}\,\tau_{a}{}^{2} + \Gamma^{b}{}_{i3}\,\tau_{a}{}^{3} + \Gamma^{b}{}_{i4}\,\tau_{a}{}^{4} - \Gamma^{1}{}_{ia}\,\tau_{1}{}^{b} - \Gamma^{2}{}_{ia}\,\tau_{2}{}^{b} - \Gamma^{3}{}_{ia}\,\tau_{3}{}^{b} - \Gamma^{4}{}_{ia}\,\tau_{4}{}^{b} + \partial_{x^{i}}\tau_{a}{}^{b}$$

- In general relativity we define the *divergence* using covariant differentiation.

```
CovariantD[λu[a], a]
% // ExpandCovariantD[labs, i]
% // EinsteinSum[] // SymmetrizeSlots[] // FullSimplify
(aa = CovariantD[NestedTensor[#], a];
    aa == (aa // UnnestTensor // ExpandCovariantD[labs, {i}])) & /@
  TensorList[[{2, 4, 6}]] // TableForm
```

$$\lambda^{a}{}_{;a}$$

$$\Gamma^{a}{}_{ai}\,\lambda^{i} + \partial_{x^{a}}\lambda^{a}$$

$$\left(\Gamma^{1}{}_{11} + \Gamma^{2}{}_{12} + \Gamma^{3}{}_{13} + \Gamma^{4}{}_{14}\right)\lambda^{1} + \left(\Gamma^{1}{}_{12} + \Gamma^{2}{}_{22} + \Gamma^{3}{}_{23} + \Gamma^{4}{}_{24}\right)\lambda^{2} + \left(\Gamma^{1}{}_{13} + \Gamma^{2}{}_{23} + \Gamma^{3}{}_{33} + \Gamma^{4}{}_{34}\right)\lambda^{3} + \left(\Gamma^{1}{}_{14} + \Gamma^{2}{}_{24} + \Gamma^{3}{}_{34} + \Gamma^{4}{}_{44}\right)\lambda^{4} + \partial_{x^{1}}\lambda^{1} + \partial_{x^{2}}\lambda^{2} + \partial_{x^{3}}\lambda^{3} + \partial_{x^{4}}\lambda^{4}$$

$$\lambda^{a}{}_{;a} == \Gamma^{a}{}_{ai}\,\lambda^{i} + \partial_{x^{a}}\lambda^{a}$$
$$\tau^{ab}{}_{;a} == \Gamma^{b}{}_{ai}\,\tau^{ai} + \Gamma^{a}{}_{ai}\,\tau^{ib} + \partial_{x^{a}}\tau^{ab}$$
$$\tau^{a}{}_{b;a} == -\Gamma^{i}{}_{ab}\,\tau^{a}{}_{i} + \Gamma^{a}{}_{ai}\,\tau^{i}{}_{b} + \partial_{x^{a}}\tau^{a}{}_{b}$$

The divergence of a covariant vector field is defined to be that of the associated contravariant vector field.

```
μu[i] == guu[i, j] μd[j]
CovariantD[#, i] & /@ %
% /. Tensor[g, {_, _, Void}, {Void, Void, Cov[_]}] → 0
```

$$\mu^{i} == g^{ij}\,\mu_{j}$$

$$\mu^{i}{}_{;i} == g^{ij}{}_{;i}\,\mu_{j} + g^{ij}\,\mu_{j;i}$$

$\mu^{i}{}_{;i} == g^{ij}\,\mu_{j;i}$

Example 2.3.1 p. 79. Divergence of a radial field in Euclidean space

```
(* coordinates *)
coord = {r, θ, ϕ};
(* manifold *) {r Cos[ϕ] Sin[θ], r Sin[ϕ] Sin[θ], r Cos[θ]};
(* natural basis *) Table[∂coord[[n]] %, {n, 1, Length[coord]}];
(* metric *)
%.Transpose[%] // Simplify;
(* Christoffel symbols in tensor notation *)DeclareBaseIndices[Range[Length[coord]]]
SetChristoffelValueRules[xu[i], CoordinatesToTensors[coord, x][%%], Γ]

(* radial field λ *)
SetAttributes[{A, B}, Constant]
SetTensorValueRules[λu[i], {A xu[1] + B, 0, 0}]
CovariantD[λu[i], i]
% // ExpandCovariantD[labs, j]
% // ToArrayValues[] // Simplify // UseCoordinates[coord]
% /. A → 1 /. B → 0
```

$\lambda^{i}{}_{;i}$

$\Gamma^{i}{}_{ij}\,\lambda^{j} + \partial_{x^{i}}\lambda^{i}$

$3\,A + \frac{2\,B}{r}$

3

2.4 Geodesic coordinates p. 79 - 81

```
Needs["TensorCalculus3`Tensorial`"]
$PrePrint =.
labs = {x, δ, g, Γ};
DefineTensorShortcuts[
 {{x, x0}, 1},
 {{g, δ, X}, 2},
 {{Γ, Γ0, X}, 3}]
SetTensorValues[δud[a, b], IdentityMatrix[NDim]]
MyRed = StyleForm[Superscript[#, "′"], FontColor → RGBColor[1, 0, 0]] &;
MyBlue = StyleForm[Superscript[#, "″"], FontColor → RGBColor[0, 0, 1]] &;
DeclareIndexFlavor[{red, MyRed}, {blue, MyBlue}]
DeclareTensorSymmetries[Γ, 3, {1, {2, 3}}]
```

1.1) Geodesic coordinates at a point on a sphere

Set up the metric and the Christoffel symbols for a 2D-sphere of radius 1 in spherical coordinates in an embedding Euclidean space.

```
ma[u_, v_] = {Cos[u] Sin[v], Sin[u] Sin[v], Cos[v]};
ru = {u, -π, π}; rv = {v, 0, π};
(* P0 *){u0, v0} = {1 / 3, 1 / 2};
```

```
(* coordinates *)
co = {u, v};
(* manifold *)
ma[u, v];
(* naturalbasis *) Table[∂co[[n]] %, {n, 1, Length[co]}];
(* metric *)
g[u_, v_] = %.Transpose[%] // Simplify;
DeclareBaseIndices[Range[Length[co]]];
metric = %% // CoordinatesToTensors[co]; SetMetricValueRules[g, metric];
(* Christoffel symbols *)
SetTensorValueRules[Γudd[a, b, c], CalculateChristoffelu[xu[i], metric, Simplify]]
SelectedTensorRules[g, gdd[_, _]] // TableForm
SelectedTensorRules[Γ, Γudd[_, i_, j_] /; OrderedQ[{i, j}]] // TableForm
```

$g_{11} \to \mathrm{Sin}[x^2]^2$

$g_{22} \to 1$

$\Gamma^1{}_{12} \to \mathrm{Cot}[x^2]$

$\Gamma^2{}_{11} \to -\frac{1}{2}\,\mathrm{Sin}[2\,x^2]$

Set the point P0. We need the Christoffel symbols evaluated in P0.

```
SetTensorValueRules[x0u[i], {u0, v0}]
(Γudd[a, b, c] // ToArrayValues[]) /. xu[i_] → x0u[i] /. TensorValueRules[x0]
SetTensorValueRules[Γ0udd[a, b, c], %]
```

$\left\{\left\{\left\{0, \mathrm{Cot}\left[\frac{1}{2}\right]\right\}, \left\{\mathrm{Cot}\left[\frac{1}{2}\right], 0\right\}\right\}, \left\{\left\{-\frac{\mathrm{Sin}[1]}{2}, 0\right\}, \{0, 0\}\right\}\right\}$

Calculation of the Jacobian matrix $X^{a'}{}_d$ and the Jacobian determinant in P0.

```
Print[Xud[red@a, d], " = "]
δud[a, d] + Γ0udd[a, d, c] (xu[c] - x0u[c])
% // ToArrayValues[] // Simplify;
SetTensorValueRules[Xud[red@a, b], %]
SetTensorValueRules[Xud[a, red@b], Inverse[%%] // Simplify]
Xud[red@a, b] // ToArrayValues[] // UseCoordinates[co] // MatrixForm
Print[Xud[red@a, d], " in P0 = "]
%% /. u → u0 /. v → v0 // MatrixForm
Print["Jacobian determinant in P0:"]
Det[%%]
```

$X^{a'}{}_d$ =

$(x^c - x0^c)\,\Gamma 0^a{}_{dc} + \delta^a{}_d$

$$\begin{pmatrix} 1 - \frac{1}{2}\mathrm{Cot}\left[\frac{1}{2}\right] + v\,\mathrm{Cot}\left[\frac{1}{2}\right] & \frac{1}{3}(-1 + 3\,u)\,\mathrm{Cot}\left[\frac{1}{2}\right] \\ \frac{1}{6}(\mathrm{Sin}[1] - 3\,u\,\mathrm{Sin}[1]) & 1 \end{pmatrix}$$

$X^{a'}{}_d$ in P0 =

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Jacobian determinant in P0:

1

Let's check that the Christoffel symbols in the red coordinates are indeed 0 in P0 .

```
Print[Xudd[red@a, d, e], " = "]
PartialD[labs][Xud[red@a, d], xu[e]]
% // ToArrayValues[];
SetTensorValueRules[Xudd[red@a, b, c], %]
(Γudd[a, b, c] // ToFlavor[red]) == Γudd[d, e, f] Xud[red@a, d] Xud[e, red@b] Xud[f, red@c] -
  Xud[e, red@b] Xud[f, red@c] Xudd[red@a, e, f]
ToArrayValues[] /@ %;
% /. xu[i_] → x0u[i] /. TensorValueRules[x0]
```

$X^{a'}{}_{de}$ =

$\partial_{x^e} X^{a'}{}_d$

$\Gamma^{a'}{}_{b'c'} == -X^e{}_{b'} X^f{}_{c'} X^{a'}{}_{ef} + X^e{}_{b'} X^f{}_{c'} X^{a'}{}_d \Gamma^d{}_{ef}$

$\{\{\{\Gamma^{1'}{}_{1'1'}, \Gamma^{1'}{}_{1'2'}\}, \{\Gamma^{1'}{}_{2'1'}, \Gamma^{1'}{}_{2'2'}\}\}, \{\{\Gamma^{2'}{}_{1'1'}, \Gamma^{2'}{}_{1'2'}\}, \{\Gamma^{2'}{}_{2'1'}, \Gamma^{2'}{}_{2'2'}\}\}\} ==$
$\{\{\{0, 0\}, \{0, 0\}\}, \{\{0, 0\}, \{0, 0\}\}\}$

Calculation of the metric in the red coordinates.

```
Print[gdd[red@a, red@b], " = "]
Xud[c, red@a] Xud[d, red@b] gdd[c, d]
% // ToArrayValues[] // Simplify;
SetTensorValues[gdd[red@a, red@b], %]
%% // MatrixForm
```

$g_{a'b'}$ =

$g_{cd} X^c{}_{a'} X^d{}_{b'}$

$$\begin{pmatrix} \frac{9\left(\text{Sin}[1]^2+36\,\text{Sin}[x^2]^2-6\,\text{Sin}[1]^2 x^1+9\,\text{Sin}[1]^2 (x^1)^2\right)}{\left(18-9\,\text{Cot}[\frac{1}{2}]+\text{Cot}[\frac{1}{2}]\,\text{Sin}[1]-6\,\text{Cot}[\frac{1}{2}]\,\text{Sin}[1]\,x^1+9\,\text{Cot}[\frac{1}{2}]\,\text{Sin}[1]\,(x^1)^2+18\,\text{Cot}[\frac{1}{2}]\,x^2\right)^2} & -\frac{27\,\text{Cot}[\frac{1}{2}]\,(-1+3x^1)\,(2\,\text{Cos}[1]-2 \ldots}{\left(18-9\,\text{Cot}[\frac{1}{2}]+\text{Cot}[\frac{1}{2}]\,\text{Sin}[1]-6\,\text{Cot}[\frac{1}{2}]\,\text{Sin}[1 \ldots\right.} \\ -\frac{27\,\text{Cot}[\frac{1}{2}]\,(-1+3x^1)\,\left(2\,\text{Cos}[1]-2\,\text{Cos}[2x^2]+\text{Sin}[1]-2\,\text{Sin}[1]\,x^2\right)}{\left(18-9\,\text{Cot}[\frac{1}{2}]+\text{Cot}[\frac{1}{2}]\,\text{Sin}[1]-6\,\text{Cot}[\frac{1}{2}]\,\text{Sin}[1]\,x^1+9\,\text{Cot}[\frac{1}{2}]\,\text{Sin}[1]\,(x^1)^2+18\,\text{Cot}[\frac{1}{2}]\,x^2\right)^2} & \frac{9\left(4\,\text{Cot}[\frac{1}{2}]^2\,\text{Sin}[x^2]^2\,(1-3x^1)^2+9 \ldots\right.}{\left(18-9\,\text{Cot}[\frac{1}{2}]+\text{Cot}[\frac{1}{2}]\,\text{Sin}[1]-6\,\text{Cot}[\frac{1}{2}]\,\text{Sin}[1] \ldots\right.} \end{pmatrix}$$

The metric components are unchanged in P0.

```
Print[gdd[a, b], " and ", gdd[red@a, red@b], " in P0 ="]
ToArrayValues[] /@ {gdd[a, b], gdd[red@a, red@b]} /. xu[i_] → x0u[i] /.
  TensorValueRules[x0] // Simplify
MatrixForm /@ % // N
%% /. List → Equal
g0red = %%%[[2]] // Simplify;
```

g_{ab} and $g_{a'b'}$ in P0 =

$\{\{\{\text{Sin}[\frac{1}{2}]^2, 0\}, \{0, 1\}\}, \{\{\text{Sin}[\frac{1}{2}]^2, 0\}, \{0, 1\}\}\}$

$\left\{\begin{pmatrix} 0.229849 & 0. \\ 0. & 1. \end{pmatrix}, \begin{pmatrix} 0.229849 & 0. \\ 0. & 1. \end{pmatrix}\right\}$

True

Transformation to geodesic coordinates.

```
Print[xu[red@a], " = "]
xu[a] - x0u[a] + 1 / 2 Γ0udd[a, b, c] (xu[b] - x0u[b]) (xu[c] - x0u[c])
% // ToArrayValues[] // FullSimplify;
SetTensorValues[xu[red@a], %]
```

```
xu[red@a] // ToArrayValues[] // UseCoordinates[co]
Solve[{uu, vv} == %, {u, v}];
Select[%, FreeQ[#, Complex] &];
maRed[uu_, vv_] = ma[co /. %[[1]] /. List → Sequence];
```

$x^{a'}$ =

$$x^a - x0^a + \frac{1}{2}\left(x^b - x0^b\right)\left(x^c - x0^c\right)\Gamma0^a{}_{bc}$$

$$\left\{\frac{1}{6}(-1+3u)\left(2-\text{Cot}\left[\frac{1}{2}\right]+2\,v\,\text{Cot}\left[\frac{1}{2}\right]\right),\ -\frac{1}{2}+v-\frac{1}{36}(1-3u)^2\,\text{Sin}[1]\right\}$$

■ 1.2) Local Cartesian coordinates at a point on a sphere

```
{s, j} = JordanDecomposition[g0red];
snor = # / Sqrt[#.#] & /@ Transpose[s] // FullSimplify // Transpose;
jscale = FullSimplify[Inverse[Sqrt[Abs[j]]]];
MatrixForm /@ {"g0red", g0red, "s,j", s, j, "snor,jscale", snor, jscale, Transpose[jscale]}
Transpose[jscale].j.jscale // Simplify // MatrixForm
```

$$\left\{\text{g0red},\ \begin{pmatrix}\text{Sin}\left[\frac{1}{2}\right]^2 & 0\\ 0 & 1\end{pmatrix},\ \text{s,j},\ \begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix},\ \begin{pmatrix}1 & 0\\ 0 & \text{Sin}\left[\frac{1}{2}\right]^2\end{pmatrix},\right.$$
$$\left.\text{snor,jscale},\ \begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix},\ \begin{pmatrix}1 & 0\\ 0 & \text{Csc}\left[\frac{1}{2}\right]\end{pmatrix},\ \begin{pmatrix}1 & 0\\ 0 & \text{Csc}\left[\frac{1}{2}\right]\end{pmatrix}\right\}$$

$$\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}$$

```
perm = (1 0
        0 1);
Pmat = snor.jscale.perm // FullSimplify;
transPmat = Transpose[Pmat];
invPmat = Inverse[Pmat];
MatrixForm /@ {Pmat, transPmat, invPmat}
(transPmat.g0red.Pmat // FullSimplify) == DiagonalMatrix[{1, 1}]
```

$$\left\{\begin{pmatrix}0 & \text{Csc}\left[\frac{1}{2}\right]\\ 1 & 0\end{pmatrix},\ \begin{pmatrix}0 & 1\\ \text{Csc}\left[\frac{1}{2}\right] & 0\end{pmatrix},\ \begin{pmatrix}0 & 1\\ \text{Sin}\left[\frac{1}{2}\right] & 0\end{pmatrix}\right\}$$

True

```
SetTensorValues[Xud[red@a, blue@b], Pmat]
Print[gdd[blue@a, blue@b], " in P0  = "]
gdd[red@c, red@d] Xud[red@c, blue@a] Xud[red@d, blue@b]
ToArrayValues[][%] /. xu[i_] → x0u[i] /. TensorValueRules[x0] // Simplify
```

$g_{a''b''}$ in P0 =

$$g_{c'd'}\,X^{c'}{}_{a''}\,X^{d'}{}_{b''}$$

{{1, 0}, {0, 1}}

Transformation to local Cartesian coordinates.

```
Print[xu[blue@a], " = "]
invPmat.ToArrayValues[][xu[red@a]] // FullSimplify;
SetTensorValues[xu[blue@a], %]
xu[blue@a] // ToArrayValues[] // UseCoordinates[co]
```

```
Solve[{uuu, vvv} == %, {u, v}];
Select[%, FreeQ[#, Complex] &];
maBlue[uuu_, vvv_] = ma[co /. %[[1]] /. List → Sequence];
```

$x^{a''}$ =

$$\left\{-\frac{1}{2}+v-\frac{1}{36}(1-3u)^2\,\text{Sin}[1],\ \frac{1}{6}(-1+3u)\left(2-\text{Cot}\left[\frac{1}{2}\right]+2\,v\,\text{Cot}\left[\frac{1}{2}\right]\right)\text{Sin}\left[\frac{1}{2}\right]\right\}$$

■ 1.3) Coordinate mesh (spherical, geodesic and local Cartesian) at a point on a sphere

```
{P0 = ma[u0, v0], g[u0, v0] // MatrixForm}
% // N

(* coordinates *) coR = {uu, vv};
(* manifold *) maRed[uu, vv];
(* naturalbasis *) Table[∂coR[[n]] %, {n, 1, Length[coR]}];
(* metric *) gR[uu_, vv_] = %.Transpose[%] // MatrixForm;
{P0 == maRed[0, 0], gR[0, 0]} // Simplify

(* coordinates *) coB = {uuu, vvv};
(* manifold *) maBlue[uuu, vvv];
(* naturalbasis *) Table[∂coB[[n]] %, {n, 1, Length[coB]}];
(* metric *) gB[uuu_, vvv_] = %.Transpose[%] // MatrixForm;
{P0 == maBlue[0, 0], gB[0, 0]} // Simplify
```

$$\left\{\left\{\text{Cos}\left[\frac{1}{3}\right]\text{Sin}\left[\frac{1}{2}\right],\ \text{Sin}\left[\frac{1}{3}\right]\text{Sin}\left[\frac{1}{2}\right],\ \text{Cos}\left[\frac{1}{2}\right]\right\},\ \begin{pmatrix}\text{Sin}\left[\frac{1}{2}\right]^2 & 0\\ 0 & 1\end{pmatrix}\right\}$$

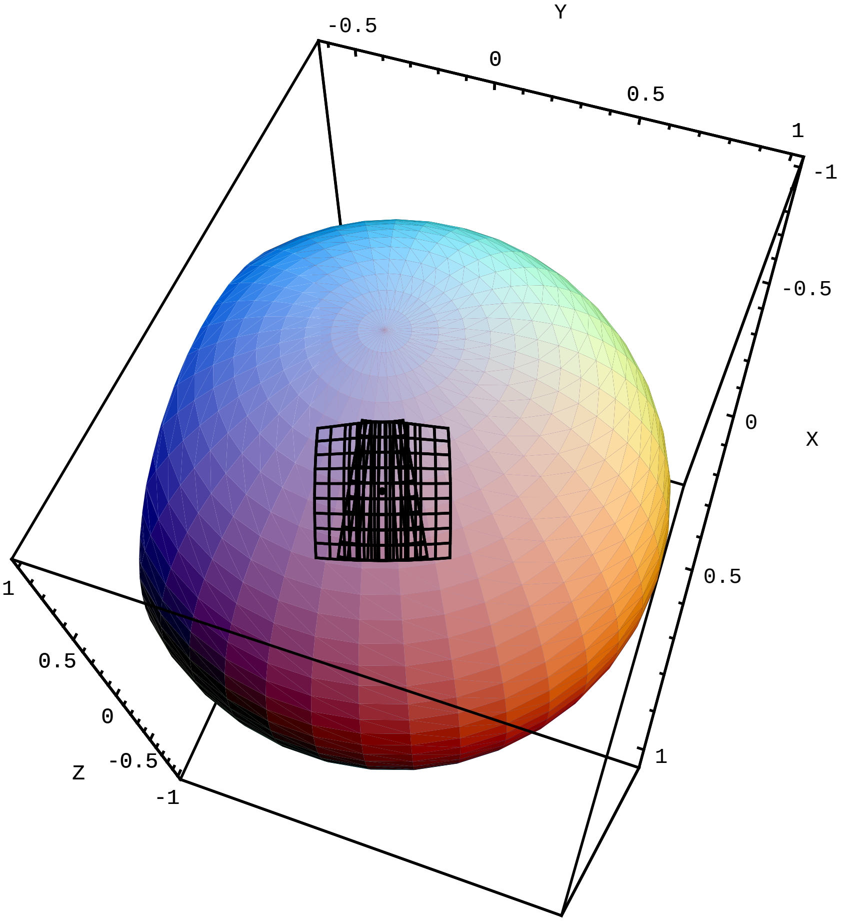
$$\left\{\{0.453036,\ 0.156865,\ 0.877583\},\ \begin{pmatrix}0.229849 & 0.\\ 0. & 1.\end{pmatrix}\right\}$$

$$\left\{\text{True},\ \begin{pmatrix}\text{Sin}\left[\frac{1}{2}\right]^2 & 0\\ 0 & 1\end{pmatrix}\right\}$$

$$\left\{\text{True},\ \begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}\right\}$$

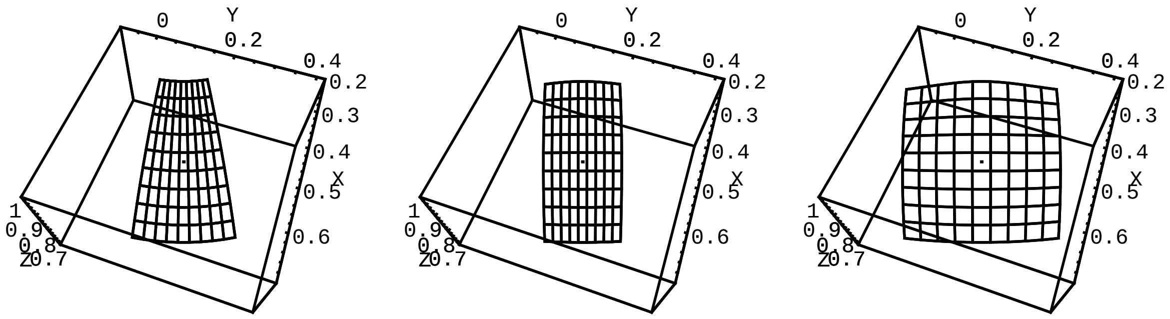
```
(* Graphics data *)
n = 10; Δu = .2; Δv = .2;
Table[ParametricPlot3D[ma[u, vn], {u, u0 - Δu, u0 + Δu}, DisplayFunction → Identity],
  {vn, v0 - Δv, v0 + Δv, 2 Δv / (n - 1)}];
Table[ParametricPlot3D[ma[un, v], {v, v0 - Δv, v0 + Δv}, DisplayFunction → Identity],
  {un, u0 - Δu, u0 + Δu, 2 Δu / (n - 1)}];
CoordinateMesh = {%, %%};

n = 10; Δu = .2; Δv = .2;
Table[ParametricPlot3D[maRed[u, vn], {u, -Δu, Δu}, DisplayFunction → Identity],
  {vn, -Δv, Δv, 2 Δv / (n - 1)}];
Table[ParametricPlot3D[maRed[un, v], {v, -Δv, Δv}, DisplayFunction → Identity],
  {un, -Δu, Δu, 2 Δu / (n - 1)}];
CoordinateMeshRed = {%, %%};

n = 10; Δu = .2; Δv = .2;
Table[ParametricPlot3D[maBlue[u, vn], {u, -Δu, Δu}, DisplayFunction → Identity],
  {vn, -Δv, Δv, 2 Δv / (n - 1)}];
Table[ParametricPlot3D[maBlue[un, v], {v, -Δv, Δv}, DisplayFunction → Identity],
  {un, -Δu, Δu, 2 Δu / (n - 1)}];
CoordinateMeshBlue = {%, %%};
```

```
pp = 40;
uvSphere = ParametricPlot3D[ma[u, v], Evaluate[ru],
    Evaluate[rv], PlotPoints -> {pp, pp}, DisplayFunction → Identity];

VP = ViewPoint → 2 P0;

Show[Graphics3D[EdgeForm[], Axes → True,
    AxesLabel → {"X", "Y", "Z"}, VP, ImageSize → {8 × 72, 6 × 72}],
  Graphics3D[Point[N[P0]]],
  CoordinateMesh,
  CoordinateMeshRed,
  CoordinateMeshBlue,
  uvSphere];
```



★ Spherical, geodesic and local Cartesian coordinates together on a sphere in P0. ★

```
Export["C:\\USERS\\LUMA\\Mathematica\\_Tensorrechnung\\2.4 - Spherical,
    geodesic and local Cartesian coordinates on a sphere.dxf", %, "DXF"]

C:\USERS\LUMA\Mathematica\_Tensorrechnung\2.4 -
  Spherical, geodesic and local Cartesian coordinates on a sphere.dxf

Show[Graphics3D[EdgeForm[], Axes → True,
      AxesLabel → {"X", "Y", "Z"}, VP, PlotRange → {{0.2, .7}, {-.1, .42}, {.7, 1}}],
     Graphics3D[Point[P0]], #, DisplayFunction → Identity] & /@
  {CoordinateMesh, CoordinateMeshRed, CoordinateMeshBlue};
Show[GraphicsArray[%], ImageSize → 72 × 8];
```



★ Spherical, geodesic and local Cartesian coordinates on a sphere in P0. ★

▪ 2) Local Cartesian coordinates in the Painlevé-Gullstrand metric (a generalized Schwarzschild metric with off-diagonal terms)

$$\text{g0red} = \begin{pmatrix} \frac{1}{3} & -\sqrt{\frac{2}{3}} & 0 & 0 \\ -\sqrt{\frac{2}{3}} & -1 & 0 & 0 \\ 0 & 0 & -9\,M^2 & 0 \\ 0 & 0 & 0 & -9\,M^2 \end{pmatrix}; \ \text{asmpt} = M > 0;$$

```
{s, j} = JordanDecomposition[g0red];
invs = Inverse[s] // FullSimplify;
snor = # / √(#.#) & /@ Transpose[s] // FullSimplify // Transpose;
invsnor = Inverse[snor] // FullSimplify;
jscale = FullSimplify[Inverse[Sqrt[Abs[j]]], asmpt];
```

$$\text{perm} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix};$$

1) Since the metric matrix is symmetrical a Jordan decomposition will give a diagonal matrix.

```
MatrixForm /@ {s, invs, j, s.j.invs} // Simplify
{(s.j.invs // Simplify) == g0red, (invs.g0red.s // Simplify) == j}
```

$$\left\{\begin{pmatrix} \frac{-2+\sqrt{10}}{\sqrt{6}} & -\frac{2+\sqrt{10}}{\sqrt{6}} & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \begin{pmatrix} \frac{\sqrt{\frac{3}{5}}}{2} & \frac{1}{2}+\frac{1}{\sqrt{10}} & 0 & 0 \\ -\frac{\sqrt{\frac{3}{5}}}{2} & \frac{1}{2}-\frac{1}{\sqrt{10}} & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix},\right.$$

$$\left.\begin{pmatrix} \frac{1}{3}\left(-1-\sqrt{10}\right) & 0 & 0 & 0 \\ 0 & \frac{1}{3}\left(-1+\sqrt{10}\right) & 0 & 0 \\ 0 & 0 & -9\,M^2 & 0 \\ 0 & 0 & 0 & -9\,M^2 \end{pmatrix}, \begin{pmatrix} \frac{1}{3} & -\sqrt{\frac{2}{3}} & 0 & 0 \\ -\sqrt{\frac{2}{3}} & -1 & 0 & 0 \\ 0 & 0 & -9\,M^2 & 0 \\ 0 & 0 & 0 & -9\,M^2 \end{pmatrix}\right\}$$

```
{True, True}
```

We calculate a new, orthogonal similarity matrix by normalizing the columns.

```
MatrixForm /@ {invsnor, invsnor.snor // FullSimplify}
invsnor == Transpose[snor]
{(snor.j.invsnor // FullSimplify) == g0red, (invsnor.g0red.snor // FullSimplify) == j}
```

$$\left\{\begin{pmatrix} \sqrt{\frac{1}{2}-\frac{1}{\sqrt{10}}} & \sqrt{\frac{1}{2}+\frac{1}{\sqrt{10}}} & 0 & 0 \\ -\sqrt{\frac{1}{2}+\frac{1}{\sqrt{10}}} & \sqrt{\frac{1}{2}-\frac{1}{\sqrt{10}}} & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}\right\}$$

```
True
```

```
{True, True}
```

2) Scaling matrix...

```
MatrixForm /@ {jscale, Inverse[jscale]}
jscale == Transpose[jscale]
jscale.j.jscale // Simplify // MatrixForm
```

$$\left\{\begin{pmatrix} \sqrt{\frac{1}{3}\left(-1+\sqrt{10}\right)} & 0 & 0 & 0 \\ 0 & \sqrt{\frac{1}{3}\left(1+\sqrt{10}\right)} & 0 & 0 \\ 0 & 0 & \frac{1}{3\,M} & 0 \\ 0 & 0 & 0 & \frac{1}{3\,M} \end{pmatrix}, \begin{pmatrix} \sqrt{\frac{3}{-1+\sqrt{10}}} & 0 & 0 & 0 \\ 0 & \sqrt{\frac{3}{1+\sqrt{10}}} & 0 & 0 \\ 0 & 0 & 3\,M & 0 \\ 0 & 0 & 0 & 3\,M \end{pmatrix}\right\}$$

```
True
```

$$\begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}$$

3) Permutation...

```
perm.(jscale.j.jscale) // Simplify // MatrixForm
%.perm // Simplify // MatrixForm
perm == Transpose[perm]
```

$$\begin{pmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & -1 & 0 \end{pmatrix}$$

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}$$

```
True
```

4) The FN P matrix is then...

```
(perm.jscale.invsnor).g0red.(snor.jscale.perm) // FullSimplify // MatrixForm
Pmat = snor.jscale.perm // FullSimplify;
transPmat = Transpose[Pmat] // FullSimplify;
invPmat = Inverse[Pmat] // FullSimplify // ToRadicals;
MatrixForm /@ {Pmat, transPmat, invPmat}
Print["Have we transPmat.g0red.Pmat==η ?"]
(transPmat.g0red.Pmat // FullSimplify) == DiagonalMatrix[{1, -1, -1, -1}]
```

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}$$

$$\left\{\begin{pmatrix} -\sqrt{\frac{1}{2}+\sqrt{\frac{2}{5}}} & \sqrt{-\frac{1}{2}+\sqrt{\frac{2}{5}}} & 0 & 0 \\ \sqrt{\frac{1}{30}\left(-5+4\sqrt{10}\right)} & \sqrt{\frac{1}{30}\left(5+4\sqrt{10}\right)} & 0 & 0 \\ 0 & 0 & \frac{1}{3M} & 0 \\ 0 & 0 & 0 & \frac{1}{3M} \end{pmatrix},\right.$$

$$\left.\begin{pmatrix} -\sqrt{\frac{1}{2}+\sqrt{\frac{2}{5}}} & \sqrt{\frac{1}{30}\left(-5+4\sqrt{10}\right)} & 0 & 0 \\ \sqrt{-\frac{1}{2}+\sqrt{\frac{2}{5}}} & \sqrt{\frac{1}{30}\left(5+4\sqrt{10}\right)} & 0 & 0 \\ 0 & 0 & \frac{1}{3M} & 0 \\ 0 & 0 & 0 & \frac{1}{3M} \end{pmatrix}, \begin{pmatrix} -\sqrt{\frac{1}{6}+\frac{2\sqrt{\frac{2}{5}}}{3}} & \sqrt{-\frac{1}{2}+\sqrt{\frac{2}{5}}} & 0 & 0 \\ \sqrt{\frac{1}{30}\left(-5+4\sqrt{10}\right)} & \sqrt{\frac{1}{2}+\sqrt{\frac{2}{5}}} & 0 & 0 \\ 0 & 0 & 3M & 0 \\ 0 & 0 & 0 & 3M \end{pmatrix}\right\}$$

```
Have we transPmat.g0red.Pmat==η ?
```

```
True
```

FN: "The implication of this for general relativity is that about each point of spacetime we can introduce a coordinate system in which $\Gamma^{\mu}_{\nu\sigma} \approx 0$, $g_{\mu\nu} \approx \eta_{\mu\nu}$ [...] showing that locally the spacetime of general relativity looks like that of special relativity."

2.5 The spacetime of general relativity p. 82 - 85

```
Needs["TensorCalculus3`Tensorial`"]
$PrePrint =.
labs = {x, δ, g, Γ};
DefineTensorShortcuts[
 {{x, j, p, dx, f, u, zero}, 1},
 {{δ, g, η, zero, F}, 2},
 {{Γ, zero}, 3}]
DeclareZeroTensor[zero]
FlatToCurvedSpacetime = {η → g, TotalD → AbsoluteD, Dif → Cov};
```

Proper time $d\tau$ for a particle and equation of motion under a force f:

```
Print["Flat spacetime:"]; {dτ^2 == ηdd[μ, ν] dxu[μ] dxu[ν], TotalD[pu[μ], τ] == fu[μ]}
Print["Curved spacetime:"]; (% /. FlatToCurvedSpacetime)
```

Flat spacetime:

$$\left\{d\tau^2 == dx^\mu\, dx^\nu\, \eta_{\mu\nu},\ \frac{dp^\mu}{d\tau} == f^\mu\right\}$$

Curved spacetime:

$$\left\{d\tau^2 == dx^\mu\, dx^\nu\, g_{\mu\nu},\ \frac{Dp^\mu}{d\tau} == f^\mu\right\}$$

Maxwell equations in curved spacetime:

```
(eqn[2, 67] = {PartialD[Fuu[μ, ν], ν] == μ0 ju[μ],
   Fold[#1 + (PartialD[Fdd[#1, #2], #3] &@@ #2) &,
     0, Table[RotateLeft[{μ, ν, σ}, i], {i, 0, 2}]] == zeroddd[μ, ν, σ]})
(eqn[2, 68] = eqn[2, 67] /. FlatToCurvedSpacetime)
```

$$\left\{F^{\mu\nu}{}_{,\nu} == \mu_0\, j^\mu,\ F_{\mu\nu,\sigma} + F_{\nu\sigma,\mu} + F_{\sigma\mu,\nu} == zero_{\mu\nu\sigma}\right\}$$

$$\left\{F^{\mu\nu}{}_{;\nu} == \mu_0\, j^\mu,\ F_{\mu\nu;\sigma} + F_{\nu\sigma;\mu} + F_{\sigma\mu;\nu} == zero_{\mu\nu\sigma}\right\}$$

Definition of time/light/space-like vectors.

```
MapThread[{#1, #2[gdd[μ, ν] λu[μ] λu[ν], 0]} &,
    {{"vector λ timelike", "vector λ null (lightlike)", "vector λ spacelike"},
     {Greater, Equal, Less}}] // TableForm // FrameBox // DisplayForm
```

vector λ timelike	$g_{\mu\nu}$ λu[μ] λu[ν] > 0
vector λ null (lightlike)	$g_{\mu\nu}$ λu[μ] λu[ν] == 0
vector λ spacelike	$g_{\mu\nu}$ λu[μ] λu[ν] < 0

- **Exercise 2.5.1 p.85.** Is the world velocity of a stationary chair (in the lab) timelike or spacelike? Is its world line a geodesic?

- **Exercise 2.5.2 p.85.** Geodesic equation for a free (massive) particle.

```
Print["Equation of motion (2,70), p.84 for a free particle:"]
AbsoluteD[pu[μ], τ] == zerou[μ]
Print["Substitute for momentum in terms of velocity and  divide out mass"]
%% /. pu[i_] → m uu[i]
(# / m & /@ %) /. a_. zerou[b_] :→ zerou[b]
Print["Expand the absolute derivative"]
MapAt[ExpandAbsoluteD[labs, {a, b}], %%, 1]
Print["Substitute velocity"]
%% /. uu[i_] → TotalD[xu[i], τ] // FrameBox // DisplayForm
```

Equation of motion (2,70), p.84 for a free particle:

$$\frac{Dp^\mu}{d\tau} == zero^\mu$$

Substitute for momentum in terms of velocity and divide out mass

$$m\,\frac{Du^\mu}{d\tau} == zero^\mu$$

$$\frac{Du^{\mu}}{d\tau} == zero^{\mu}$$

```
Expand the absolute derivative
```

$$\frac{du^{\mu}}{d\tau} + u^{b}\,\Gamma^{\mu}{}_{ab}\,\frac{dx^{a}}{d\tau} == zero^{\mu}$$

```
Substitute velocity
```

$$\boxed{\Gamma^{\mu}{}_{ab}\,\frac{dx^{a}}{d\tau}\,\frac{dx^{b}}{d\tau} + \frac{d^{2}x^{\mu}}{d\tau\, d\tau} == zero^{\mu}}$$

Compare with equation (2.71), p. 84.

■ Intermezzo: The concept of force in special relativity. (Der Begriff der Kraft in der Massenpunktdynamik der speziellen Relativitätstheorie.)

A particle with rest mass $m_0>0$ is moving around subject to some force. This motion is observed by two inertial systems Σ and Σ' ("p" = prime) with relative velocity v (measured by Σ); each is endowed with a Cartesian coordinate system and they have overlapping axes at t=t'=0 (standard configuration). We determine here various quantities (position, velocity, acceleration, kinetic energy, impulse, force) which characterizes a particular motion as viewed by Σ or Σ'.

● 1. Some definitions (See my Special Relativity *Mathematica* notebook for an *ab initio* derivation *à la Lévy-Leblond* of the Lorentz transformation matrix Λ)

```
η = DiagonalMatrix[{1, -1, -1, -1}];
γ[v_] := 1/Sqrt[1 - (v/c)^2];
```

$$\Lambda[vx\_, vy\_, vz\_] = \begin{pmatrix} \gamma & -vx\,\gamma/c & -vy\,\gamma/c & -vz\,\gamma/c \\ -vx\,\gamma/c & 1+\frac{vx^2(-1+\gamma)}{v^2} & \frac{vx\,vy(-1+\gamma)}{v^2} & \frac{vx\,vz(-1+\gamma)}{v^2} \\ -vy\,\gamma/c & \frac{vx\,vy(-1+\gamma)}{v^2} & 1+\frac{vy^2(-1+\gamma)}{v^2} & \frac{vy\,vz(-1+\gamma)}{v^2} \\ -vz\,\gamma/c & \frac{vx\,vz(-1+\gamma)}{v^2} & \frac{vy\,vz(-1+\gamma)}{v^2} & 1+\frac{vz^2(-1+\gamma)}{v^2} \end{pmatrix} //.$$

```
    {γ → γ[v], v → Sqrt[vx^2 + vy^2 + vz^2]} // Simplify;
```

Test: The coordinate transformation in standard configuration and proof of the isometry of the Lorentz transformation expressed by Λ

```
Λ[vx, 0, 0];
{c t, x, y, z};
MatrixForm[{c tp, xp, yp, zp}] ==
  MatrixForm[%%].MatrixForm[%] == MatrixForm[Simplify[%%.%]]
Print["(c tp)^2-xp^2-yp^2-zp^2 ≟ (c t)^2-x^2-y^2-z^2"]
Λ[vx, vy, vz].{c t, x, y, z};
Simplify[%.η.%] == (c t)^2 - x^2 - y^2 - z^2
```

$$\begin{pmatrix} c\,tp \\ xp \\ yp \\ zp \end{pmatrix} == \begin{pmatrix} \frac{1}{\sqrt{1-\frac{vx^2}{c^2}}} & -\frac{vx}{c\sqrt{1-\frac{vx^2}{c^2}}} & 0 & 0 \\ -\frac{vx}{c\sqrt{1-\frac{vx^2}{c^2}}} & \frac{1}{\sqrt{1-\frac{vx^2}{c^2}}} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} . \begin{pmatrix} c\,t \\ x \\ y \\ z \end{pmatrix} == \begin{pmatrix} \frac{c^2\,t - vx\,x}{c\sqrt{1-\frac{vx^2}{c^2}}} \\ \frac{-t\,vx + x}{\sqrt{1-\frac{vx^2}{c^2}}} \\ y \\ z \end{pmatrix}$$

```
(c tp)²-xp²-yp²-zp² ≟ (c t)²-x²-y²-z²
True
```

- 2. A little collection of motions restrained to the x-y-plane.

```
msg = "Kräftefreie Bewegung";
v = 1 / 2;
x1[t_] := v t;
x2[t_] := v t; x3[t_] := 0

msg = "Hyperbolische Bewegung";
g = 1 / 4; α = 3 π / 4;
x1[t_] := Cos[α] c² / g (√(1 + (g t / c)²) - 1); x2[t_] := Sin[α] c² / g (√(1 + (g t / c)²) - 1);
x3[t_] := 0

msg = "Bewegung auf Lissajous-Orbit";
(* Caution: Problems with the inverse t =f(t'), see below !*)
x1[t_] := Cos[π t] / 4;
x2[t_] := Sin[t] / 4; x3[t_] := 0

msg = "Parabolic orbit (Bewegung auf Parabel-Orbit)";
x1[t_] := t / 2;
x2[t_] := t² / 20; x3[t_] := 0
```

For simplicity we assume c=1 and m_0>0 ; time t ∈ [0, 2π].

```
c = 1 ; m0 = 1;
{ti, tf} = {0, 2 π};
$Assumptions = {ti ≤ t ≤ tf};
```

- 3. Cinematic as seen by Σ

Ordinary position X(t), velocity V(t) and acceleration A(t):

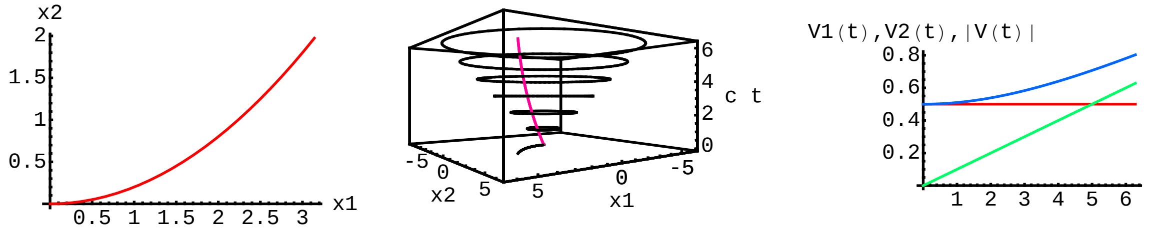
```
Print[" --- ", msg, " --- "]
X[t_] := {x1[t], x2[t], x3[t]}
V[t_] := D[X[t], t]
Vn[t_] := Simplify[Norm[V[t]]]
A[t_] := Simplify[D[X[t], {t, 2}]]
An[t_] := Simplify[Norm[A[t]]]
Print["{X(t), V(t), A(t), |V(t)|, |A(t)|} ="]
MatrixForm /@ {X[t], V[t], A[t], Vn[t], An[t]}

 --- Parabolic orbit (Bewegung auf Parabel-Orbit) ---

{X(t), V(t), A(t), |V(t)|, |A(t)|} =
```

$$\left\{\begin{pmatrix}\frac{t}{2}\\ \frac{t^2}{20}\\ 0\end{pmatrix}, \begin{pmatrix}\frac{1}{2}\\ \frac{t}{10}\\ 0\end{pmatrix}, \begin{pmatrix}0\\ \frac{1}{10}\\ 0\end{pmatrix}, \frac{\sqrt{25+t^2}}{10}, \frac{1}{10}\right\}$$

A graphical representation of the motion.

```
xyOrbitplot =
  ParametricPlot[{x1[x0 / c], x2[x0 / c]}, {x0, c ti, c tf}, AspectRatio → Automatic,
   AxesLabel → {"x1", "x2"}, PlotStyle → {Hue[0]}, DisplayFunction → Identity];
MinkowskiOrbit = ParametricPlot3D[{x1[x0 / c], x2[x0 / c], x0, Hue[0.9]},
   {x0, c ti, c tf}, DisplayFunction → Identity];
pMinkowskiOrbit = ParametricPlot3D[{x1[x0 / c], x2[x0 / c], 0},
```

```
    {x0, c ti, c tf}, DisplayFunction → Identity];
LightCone = Table[
    ParametricPlot3D[{Cos[α] x0, Sin[α] x0, x0}, {α, 0, 2 π}, DisplayFunction → Identity],
    {x0, c ti, c tf, (c tf - c ti) / (7 - 1)}];
vplot = Plot[Evaluate[{V[t][[1]], V[t][[2]], Vn[t]}],
    {t, ti, tf}, PlotStyle → {Hue[0], Hue[0.4], Hue[0.6]},
    AxesLabel → {"t", "V1(t),V2(t),|V(t)|"}, DisplayFunction → Identity];
Show[GraphicsArray[{xyOrbitplot, Show[MinkowskiOrbit, pMinkowskiOrbit, LightCone,
      ViewPoint → {1, 1.4, 0}, AxesLabel → {"x1", "x2", "c t"}], vplot}, ImageSize → 72 × 8]];
```



⋆ Orbit in the x-y-plane, orbit in the Minkowski space (with projection and light cone) and plot of {V1(t), V2(t), | V(t)| }⋆

4-position x(t), 4-velocity v(t) and 4-acceleration a(t):

```
x = {c t, x1[t], x2[t], x3[t]};
v = γ[Vn[t]] D[x, t] // Simplify;
a = γ[Vn[t]] D[v, t] // Simplify;
Print["{x(t),v(t),a(t)} ="]
MatrixForm /@ {x, v, a}
Print["{x(t)², v(t)², a(t)², v(t)·a(t)} ="]
Simplify /@ {x.η.x, v.η.v, a.η.a, v.η.a} // Simplify
```

{x(t),v(t),a(t)} =

$$\left\{\begin{pmatrix} t \\ \frac{t}{2} \\ \frac{t^2}{20} \\ 0 \end{pmatrix}, \begin{pmatrix} \frac{10}{\sqrt{75-t^2}} \\ \frac{5}{\sqrt{75-t^2}} \\ \frac{t}{\sqrt{75-t^2}} \\ 0 \end{pmatrix}, \begin{pmatrix} \frac{100\,t}{(-75+t^2)^2} \\ \frac{50\,t}{(-75+t^2)^2} \\ \frac{750}{(-75+t^2)^2} \\ 0 \end{pmatrix}\right\}$$

{x(t)², v(t)², a(t)², v(t)·a(t)} =

$$\left\{-\frac{1}{400} t^2 \left(-300+t^2\right), 1, \frac{7500}{\left(-75+t^2\right)^3}, 0\right\}$$

● 4. Dynamics as seen by Σ

Einstein-force $K_E = \frac{d}{dt}(\gamma\, m_0\, V)$ and 4-force f(t):

```
KE[t_] := D[m0 γ[Vn[t]] V[t], t] // Simplify
f = Flatten[{γ[Vn[t]] / c KE[t].V[t], γ[Vn[t]] KE[t]}] // Simplify;
Print["{KE(t),f(t)} = "]
MatrixForm /@ {KE[t], f}
Print["{f(t)²,v(t)·f(t)} = "]
{f.η.f, v.η.f} // Simplify
```

{KE(t),f(t)} =

$$\left\{\begin{pmatrix}\frac{5\,t}{(75-t^2)^{3/2}}\\ \frac{75}{(75-t^2)^{3/2}}\\ 0\end{pmatrix},\ \begin{pmatrix}\frac{100\,t}{(-75+t^2)^2}\\ \frac{50\,t}{(-75+t^2)^2}\\ \frac{750}{(-75+t^2)^2}\\ 0\end{pmatrix}\right\}$$

$\{f(t)^2, v(t)\cdot f(t)\}$ =

$$\left\{\frac{7500}{(-75+t^2)^3},\ 0\right\}$$

4-momentum p(t):

```
p = m0 v;
γ[Vn[t]] D[p, t] // Simplify;
Print["p(t), γ ṗ(t) and γ ṗ(t) =? f(t)"]
MatrixForm /@ {p, %%, %% == f}
Print["γ p⃗̇(t) and γ p⃗̇(t) =? \!\(K\_E\)(t)"]
D[Rest[p], t] // Simplify;
{MatrixForm[%], % == KE[t]}
```

p(t), γ ṗ(t) and γ ṗ(t) $\stackrel{?}{=}$ f(t)

$$\left\{\begin{pmatrix}\frac{10}{\sqrt{75-t^2}}\\ \frac{5}{\sqrt{75-t^2}}\\ \frac{t}{\sqrt{75-t^2}}\\ 0\end{pmatrix},\ \begin{pmatrix}\frac{100\,t}{(-75+t^2)^2}\\ \frac{50\,t}{(-75+t^2)^2}\\ \frac{750}{(-75+t^2)^2}\\ 0\end{pmatrix},\ \text{True}\right\}$$

γ $\dot{\vec{p}}$(t) and γ $\dot{\vec{p}}$(t) $\stackrel{?}{=}$ K_E(t)

$$\left\{\begin{pmatrix}\frac{5\,t}{(75-t^2)^{3/2}}\\ \frac{75}{(75-t^2)^{3/2}}\\ 0\end{pmatrix},\ \text{True}\right\}$$

Kinetic energy T(t):

```
Print[" T(t) and Ṫ(t) =? \!\(K\_E\)(t)·V(t)"]
T[t_] := (m0 γ[Vn[t]] c^2 - m0 c^2) // Simplify
{T[t],
 Simplify /@ (D[T[t], t] == KE[t].V[t])}
```

T(t) and Ṫ(t) $\stackrel{?}{=}$ K_E(t)·V(t)

$$\left\{-1+\frac{10}{\sqrt{75-t^2}},\ \text{True}\right\}$$

- 5. Comparisons of forces as seen by Σ and Σ'.
  (Vergleich der Kräfte, die verschiedene Inertialsysteme bezüglich derselben Bewegung eines Massenpunktes beobachten.)

a) First, we choose a particular relative velocity v of Σ' with respect to Σ and set up the corresponding Lorentz transformation matrix.

```
Λ = Λ[1/2, 1/2, 1/2] // Simplify;
% // MatrixForm
```

$$\begin{pmatrix} 2 & -1 & -1 & -1 \\ -1 & \frac{4}{3} & \frac{1}{3} & \frac{1}{3} \\ -1 & \frac{1}{3} & \frac{4}{3} & \frac{1}{3} \\ -1 & \frac{1}{3} & \frac{1}{3} & \frac{4}{3} \end{pmatrix}$$

b) 4-position `x'(t')` as seen by Σ'.

Warning: In general the inverse `t=f(t')`to be calculated by the Solve[] statement may cause serious problems!

```
xp = Λ.x // Simplify;
{tip, tfp} = {xp[[1]] /. t → ti, xp[[1]] /. t → tf};
$Assumptions = {tip ≤ tp ≤ tfp};
Solve[{c tp == xp[[1]]}, t]
trule = %[[1, 1]]; (* soluzione scelta (chosen solution): t=0 per tp=0 *)
xp = xp /. trule // Simplify;
xp // MatrixForm
```

$$\left\{\left\{t \to 15 - \sqrt{5}\ \sqrt{45 - 4\,tp}\right\}, \left\{t \to 15 + \sqrt{5}\ \sqrt{45 - 4\,tp}\right\}\right\}$$

$$\begin{pmatrix} tp \\ \frac{1}{6}\left(15 - \sqrt{5}\ \sqrt{45 - 4\,tp} - 2\,tp\right) \\ \frac{1}{6}\left(105 - 7\sqrt{5}\ \sqrt{45 - 4\,tp} - 8\,tp\right) \\ \frac{1}{3}\left(-15 + \sqrt{5}\ \sqrt{45 - 4\,tp} - tp\right) \end{pmatrix}$$

c) Ordinary velocity `V'(t')`and its norm as seen by Σ'.

```
Xp[tp_] = Rest[xp];
Vp[tp_] := Simplify[D[Xp[tp], tp]]
Vnp[tp_] := Simplify[Norm[Vp[tp]]]
MatrixForm /@ {Vp[tp], Vnp[tp]} // N // Simplify
```

$$\left\{\begin{pmatrix} -0.333333 + \frac{0.745356}{\sqrt{45.-4.\,tp}} \\ -1.33333 + \frac{5.21749}{\sqrt{45.-4.\,tp}} \\ -0.333333 - \frac{1.49071}{\sqrt{45.-4.\,tp}} \end{pmatrix}, 1.41421\sqrt{\frac{15. - 1.67705\sqrt{45. - 4.\,tp} - 1.\,tp}{11.25 - 1.\,tp}}\right\}$$

d) 4-velocity `v'(t')` and 4-acceleration `a'(t')`as seen by Σ'.

```
vp = γ[Vnp[tp]] D[xp, tp] // Simplify;
ap = γ[Vnp[tp]] D[vp, tp] // Simplify;
MatrixForm /@ {vp, ap}
{vp.η.vp, ap.η.vp} // Simplify
```

$$\left\{\begin{pmatrix} \sqrt{\frac{45-4\,tp}{-75+6\sqrt{5}\sqrt{45-4\,tp}+4\,tp}} \\ \frac{\sqrt{5}-\sqrt{45-4\,tp}}{3\sqrt{-75+6\sqrt{5}\sqrt{45-4\,tp}+4\,tp}} \\ \frac{7\sqrt{5}-4\sqrt{45-4\,tp}}{3\sqrt{-75+6\sqrt{5}\sqrt{45-4\,tp}+4\,tp}} \\ \frac{-2\sqrt{5}-\sqrt{45-4\,tp}}{3\sqrt{-75+6\sqrt{5}\sqrt{45-4\,tp}+4\,tp}} \end{pmatrix}, \begin{pmatrix} \frac{6\left(-45\sqrt{5}+10\sqrt{45-4\,tp}+4\sqrt{5}\,tp\right)}{\sqrt{45-4\,tp}\left(-75+6\sqrt{5}\sqrt{45-4\,tp}+4\,tp\right)^2} \\ \frac{2\left(-15+2\sqrt{5}\sqrt{45-4\,tp}\right)}{3\left(-75+6\sqrt{5}\sqrt{45-4\,tp}+4\,tp\right)^2} \\ \frac{10\left(-3+\sqrt{5}\sqrt{45-4\,tp}\right)}{3\left(-75+6\sqrt{5}\sqrt{45-4\,tp}+4\,tp\right)^2} \\ \frac{10\left(-12+\sqrt{5}\sqrt{45-4\,tp}\right)}{3\left(-75+6\sqrt{5}\sqrt{45-4\,tp}+4\,tp\right)^2} \end{pmatrix}\right\}$$

{1, 0}

e) Einstein-force $K_E'(t')$ and 4-force $f'(t')$ as seen by Σ'.

```
KsE[tp_] = D[m0 γ[Vnp[tp]] Vp[tp], tp] // Simplify;
fp[tp_] = Flatten[{γ[Vnp[tp]] / c KsE[tp].Vp[tp], γ[Vnp[tp]] KsE[tp]}] // Simplify;
MatrixForm /@ {KsE[tp], fp[tp]}
vp.η.fp[tp] // Simplify
```

$$\left\{\begin{pmatrix} \frac{2\left(15\left(-6\sqrt{5}+\sqrt{45-4\,tp}\right)+8\sqrt{5}\,tp\right)}{3\,(-45+4\,tp)\left(-75+6\sqrt{5}\sqrt{45-4\,tp}+4\,tp\right)^{3/2}} \\ \frac{10\left(-45\sqrt{5}+3\sqrt{45-4\,tp}+4\sqrt{5}\,tp\right)}{3\,(-45+4\,tp)\left(-75+6\sqrt{5}\sqrt{45-4\,tp}+4\,tp\right)^{3/2}} \\ \frac{10\left(-45\sqrt{5}+12\sqrt{45-4\,tp}+4\sqrt{5}\,tp\right)}{3\,(-45+4\,tp)\left(-75+6\sqrt{5}\sqrt{45-4\,tp}+4\,tp\right)^{3/2}} \end{pmatrix}, \begin{pmatrix} \frac{60-6\sqrt{5}\sqrt{45-4\,tp}}{\left(-75+6\sqrt{5}\sqrt{45-4\,tp}+4\,tp\right)^2} \\ -\frac{2\left(15\left(-6\sqrt{5}+\sqrt{45-4\,tp}\right)+8\sqrt{5}\,tp\right)}{3\sqrt{45-4\,tp}\left(-75+6\sqrt{5}\sqrt{45-4\,tp}+4\,tp\right)^2} \\ -\frac{10\left(-45\sqrt{5}+3\sqrt{45-4\,tp}+4\sqrt{5}\,tp\right)}{3\sqrt{45-4\,tp}\left(-75+6\sqrt{5}\sqrt{45-4\,tp}+4\,tp\right)^2} \\ -\frac{10\left(-45\sqrt{5}+12\sqrt{45-4\,tp}+4\sqrt{5}\,tp\right)}{3\sqrt{45-4\,tp}\left(-75+6\sqrt{5}\sqrt{45-4\,tp}+4\,tp\right)^2} \end{pmatrix}\right\}$$

0

f) Is the 4-force $f'(t')$ as seen by Σ' and calculated directly from the 4-position $x'(t')$ identical to the 4-force calculated with the Lorentz transformation of the 4-force $f(t)$ as seen by Σ with $t \to t'$?

```
Λ.f /. trule // Simplify;
% // MatrixForm
Simplify[fp[tp] == %%]
```

$$\begin{pmatrix} \frac{60-6\sqrt{5}\sqrt{45-4\,tp}}{\left(-75+6\sqrt{5}\sqrt{45-4\,tp}+4\,tp\right)^2} \\ \frac{2\left(-15+2\sqrt{5}\sqrt{45-4\,tp}\right)}{3\left(-75+6\sqrt{5}\sqrt{45-4\,tp}+4\,tp\right)^2} \\ \frac{10\left(-3+\sqrt{5}\sqrt{45-4\,tp}\right)}{3\left(-75+6\sqrt{5}\sqrt{45-4\,tp}+4\,tp\right)^2} \\ \frac{10\left(-12+\sqrt{5}\sqrt{45-4\,tp}\right)}{3\left(-75+6\sqrt{5}\sqrt{45-4\,tp}+4\,tp\right)^2} \end{pmatrix}$$

True

Yes, it is! As it should...

> "In moving from the flat spacetime of special relativity to the curved spacetime of general relativity we hope somehow to incorporate the effects of gravity, and the point of view we

are adopting is that gravity is not a force, and that gravitational effects may be explained in terms of the curvature of spacetime." (FN)

2.6 Newton's laws of motion p. 86 - 87

2.7 Gravitational potential and the geodesic p. 87 - 89

```
Needs["TGeneralRelativity1`GeneralRelativity`"]
$PrePrint =.
labs = {x, δ, g, Γ};
DeclareBaseIndices[{0, 1, 2, 3}]
DefineTensorShortcuts[
 {{x, v}, 1},
 {{δ, g, η, h}, 2},
 {{Γ, h}, 3}]
DeclareTensorSymmetries[Γ, 3, {1, {2, 3}}]
```

From the geodesic equation describing the motion of a free massive particle in GR we can recover the Newtonian equation of motion of a particle moving in a gravitational field by means of a weak&quasi-static field and a slow motion approximation. This is done here in three "easy" steps... We suppose that $g_{\mu\nu} = \eta_{\mu\nu} + h_{\mu\nu}$ with h << η = diag(1,-1,-1,-1) (This means also that the coordinates x^ν are nearly Cartesian).

- First, from the geodesic equation using the proper time τ as (affine) parameter we get the geodesic equation using the coordinate time t=t(τ) as (non-affine) parameter; the inverse is τ=τ(t). (See also Exercise 2.1.1, p.63, and subsection 2.1.)

```
Print["1) Geodesic equation with the coordinate time as parameter"]
AbsoluteD[vu[μ], τ] == 0
AbsoluteD[vu[μ], t] == h[τ[t]] vu[μ]
% // ExpandAbsoluteD[labs, {ν, σ}]
(eqn[2, 75] = % /. vu[i_] → TotalD[xu[i], t]) // FrameBox // DisplayForm
Print["(see eqn[2,75] and eqn[2,76], p.87)\nwhere"]
h[τ[t]] == TotalD[τ, {t, t}] / TotalD[τ, t] // TraditionalForm
Print["considering that"]
HoldForm[(-f''[τ] f'[τ]^-2 /. τ → InverseFunction[f][t]) ==
  InverseFunction[f]''[t] / InverseFunction[f]'[t]]
ReleaseHold[
 %]
```

1) Geodesic equation with the coordinate time as parameter

$$\frac{Dv^\mu}{d\tau} == 0$$

$$\frac{Dv^\mu}{dt} == h[\tau[t]]\, v^\mu$$

$$\frac{dv^\mu}{dt} + v^\sigma \Gamma^\mu{}_{\nu\sigma} \frac{dx^\nu}{dt} == h[\tau[t]]\, v^\mu$$

$$\boxed{\frac{d^2x^\mu}{dt\,dt} + \Gamma^\mu{}_{\nu\sigma} \frac{dx^\nu}{dt} \frac{dx^\sigma}{dt} == h[\tau[t]] \frac{dx^\mu}{dt}}$$

(see eqn[2,75] and eqn[2,76], p.87)
where

$$h(\tau(t)) = \frac{\frac{d^2\tau}{dt^2}}{\frac{d\tau}{dt}}$$

considering that

$$\left(-\frac{f''[\tau]}{f'[\tau]^2} \,/.\, \tau \to f^{(-1)}[t]\right) == \frac{f^{(-1)''}[t]}{f^{(-1)'}[t]}$$

True

(see eqn[2,75] and eqn[2,76], p.87)
where

$$h(\tau(t)) = \frac{\frac{d^2\tau}{dt^2}}{\frac{d\tau}{dt}}$$

considering that

$$\left(-\frac{f''[\tau]}{f'[\tau]^2} \,/.\, \tau \to f^{(-1)}[t]\right) == \frac{f^{(-1)''}[t]}{f^{(-1)'}[t]}$$

True

- Now we construct all the approximations needed.

```
Print["2,1) Geodesic spatial part approximation"]
eqn[2, 75]
Print["Do a partial sum on time component and simplify"]
MapAt[PartialSum[0, {j, k}], %%, {{1, 2}}]
% // SymmetrizeSlots[]
% // MapLevelParts[SimplifyTensorSum, {1, {2, 3}}]
Print["Put ", xu[0] → c t]
SetAttributes[c, Constant]; $Assumptions = {c > 0};
%%% /. xu[0] → c t
Expand[# / c^2] & /@ %
Print["The second order velocity term on left can be neglected"]
%% /. TotalD[xu[a_], t] TotalD[xu[b_], t] → 0
Print["Do a partial array expansion on the time component and take the spatial part"]
%% // PartialArray[0, {i}]
(eqn[2, 77] = Last /@ %) // FrameBox // DisplayForm
Print["Compare with eqn[2,77], p.87."]
```

2,1) Geodesic spatial part approximation

$$\frac{dd^2 x^\mu}{dt\,dt} + \Gamma^\mu{}_{\nu\sigma}\frac{dx^\nu}{dt}\frac{dx^\sigma}{dt} == h[\tau[t]]\frac{dx^\mu}{dt}$$

Do a partial sum on time component and simplify

$$\Gamma^\mu{}_{00}\left(\frac{dx^0}{dt}\right)^2 + \Gamma^\mu{}_{j0}\frac{dx^0}{dt}\frac{dx^j}{dt} + \Gamma^\mu{}_{0k}\frac{dx^0}{dt}\frac{dx^k}{dt} + \Gamma^\mu{}_{jk}\frac{dx^j}{dt}\frac{dx^k}{dt} + \frac{d^2x^\mu}{dt\,dt} == h[\tau[t]]\frac{dx^\mu}{dt}$$

$$\Gamma^\mu{}_{00}\left(\frac{dx^0}{dt}\right)^2 + \Gamma^\mu{}_{0j}\frac{dx^0}{dt}\frac{dx^j}{dt} + \Gamma^\mu{}_{0k}\frac{dx^0}{dt}\frac{dx^k}{dt} + \Gamma^\mu{}_{jk}\frac{dx^j}{dt}\frac{dx^k}{dt} + \frac{d^2x^\mu}{dt\,dt} == h[\tau[t]]\frac{dx^\mu}{dt}$$

$$\Gamma^\mu{}_{00}\left(\frac{dx^0}{dt}\right)^2 + 2\,\Gamma^\mu{}_{0j}\frac{dx^0}{dt}\frac{dx^j}{dt} + \Gamma^\mu{}_{jk}\frac{dx^j}{dt}\frac{dx^k}{dt} + \frac{d^2x^\mu}{dt\,dt} == h[\tau[t]]\frac{dx^\mu}{dt}$$

Put $x^0 \to c\,t$

$$c^2\,\Gamma^{\mu}{}_{00} + 2\,c\,\Gamma^{\mu}{}_{0j}\,\frac{dx^j}{dt} + \Gamma^{\mu}{}_{jk}\,\frac{dx^j}{dt}\,\frac{dx^k}{dt} + \frac{d^2x^{\mu}}{dt\,dt} == h[\tau[t]]\,\frac{dx^{\mu}}{dt}$$

$$\Gamma^{\mu}{}_{00} + \frac{2\,\Gamma^{\mu}{}_{0j}\,\frac{dx^j}{dt}}{c} + \frac{\Gamma^{\mu}{}_{jk}\,\frac{dx^j}{dt}\,\frac{dx^k}{dt}}{c^2} + \frac{\frac{d^2x^{\mu}}{dt\,dt}}{c^2} == \frac{h[\tau[t]]\,\frac{dx^{\mu}}{dt}}{c^2}$$

The second order velocity term on left can be neglected

$$\Gamma^{\mu}{}_{00} + \frac{2\,\Gamma^{\mu}{}_{0j}\,\frac{dx^j}{dt}}{c} + \frac{\frac{d^2x^{\mu}}{dt\,dt}}{c^2} == \frac{h[\tau[t]]\,\frac{dx^{\mu}}{dt}}{c^2}$$

Do a partial array expansion on the time component and take the spatial part

$$\left\{\Gamma^{0}{}_{00} + \frac{\frac{d^2x^0}{dt\,dt}}{c^2} + \frac{2\,\Gamma^{0}{}_{0j}\,\frac{dx^j}{dt}}{c},\ \Gamma^{i}{}_{00} + \frac{\frac{d^2x^i}{dt\,dt}}{c^2} + \frac{2\,\Gamma^{i}{}_{0j}\,\frac{dx^j}{dt}}{c}\right\} == \left\{\frac{h[\tau[t]]\,\frac{dx^0}{dt}}{c^2},\ \frac{h[\tau[t]]\,\frac{dx^i}{dt}}{c^2}\right\}$$

$$\boxed{\Gamma^{i}{}_{00} + \frac{\frac{d^2x^i}{dt\,dt}}{c^2} + \frac{2\,\Gamma^{i}{}_{0j}\,\frac{dx^j}{dt}}{c} == \frac{h[\tau[t]]\,\frac{dx^i}{dt}}{c^2}}$$

Compare with eqn[2,77], p.87.

```
Print["2,2) Up metric approximation"]
Print["this is the Kronecker"]
gdd[μ, ν] guu[σ, μ]
Print["ansatz (with h ≪ η)"]
%% /. gdd[μ, ν] → ηdd[μ, ν] + hdd[μ, ν] /. guu[σ, μ] → β ηuu[σ, μ] + α huu[σ, μ]
% // Expand
Print["simplify with metric g ≈ η"]
%% // MetricSimplify[η]
Print["with α=-1 and β=+1 we get the Kronecker to first order"]
%% /. α → -1 /. β → 1
```

2,2) Up metric approximation

this is the Kronecker

$$g_{\mu\nu}\,g^{\sigma\mu}$$

ansatz (with h ≪ η)

$$(h_{\mu\nu} + \eta_{\mu\nu})\,(\alpha\,h^{\sigma\mu} + \beta\,\eta^{\sigma\mu})$$

$$\alpha\,h_{\mu\nu}\,h^{\sigma\mu} + \alpha\,h^{\sigma\mu}\,\eta_{\mu\nu} + \beta\,h_{\mu\nu}\,\eta^{\sigma\mu} + \beta\,\eta_{\mu\nu}\,\eta^{\sigma\mu}$$

simplify with metric g ≈ η

$$\alpha\,h^{\sigma}{}_{\nu} + \beta\,h^{\sigma}{}_{\nu} + \alpha\,h_{\mu\nu}\,h^{\sigma\mu} + \beta\,\eta^{\sigma}{}_{\nu}$$

with α=-1 and β=+1 we get the Kronecker to first order

$$-h_{\mu\nu}\,h^{\sigma\mu} + \eta^{\sigma}{}_{\nu}$$

```
Print["2,3) Connection coefficients approximation"]
Γudd[μ, ν, σ] == 1 / 2 guu[μ, ρ] (PartialD[labs][gdd[σ, ρ], xu[ν]] +
    PartialD[labs][gdd[ν, ρ], xu[σ]] - PartialD[labs][gdd[ν, σ], xu[ρ]])
Print["Substituting the weak gravity approximation"]
%% /. gdd[a_, b_] → ηdd[a, b] + hdd[a, b]
```

```
Print["η does not depend on x"]
%% // NondependentPartialD[{η, x}]
Print["Up metric approximation"]
%% /. guu[a_, b_] → ηuu[a, b] - huu[a, b]
Print["Neglect h compared to η"]
%% /. huu[_, _] → 0
% /. PartialD[labs][hdd[a_, b_], xu[c_]] → hddd[a, b, Dif[c]];
ΓApproxRule = Rule @@ % // LHSSymbolsToPatterns[{μ, ν, σ}];
```

2,3) Connection coefficients approximation

$$\Gamma^{\mu}{}_{\nu\sigma} == \frac{1}{2} g^{\mu\rho} \left(\partial_{x^{\sigma}} g_{\nu\rho} - \partial_{x^{\rho}} g_{\nu\sigma} + \partial_{x^{\nu}} g_{\sigma\rho}\right)$$

Substituting the weak gravity approximation

$$\Gamma^{\mu}{}_{\nu\sigma} == \frac{1}{2} g^{\mu\rho} \left(\partial_{x^{\sigma}} h_{\nu\rho} - \partial_{x^{\rho}} h_{\nu\sigma} + \partial_{x^{\nu}} h_{\sigma\rho} + \partial_{x^{\sigma}} \eta_{\nu\rho} - \partial_{x^{\rho}} \eta_{\nu\sigma} + \partial_{x^{\nu}} \eta_{\sigma\rho}\right)$$

η does not depend on x

$$\Gamma^{\mu}{}_{\nu\sigma} == \frac{1}{2} g^{\mu\rho} \left(\partial_{x^{\sigma}} h_{\nu\rho} - \partial_{x^{\rho}} h_{\nu\sigma} + \partial_{x^{\nu}} h_{\sigma\rho}\right)$$

Up metric approximation

$$\Gamma^{\mu}{}_{\nu\sigma} == \frac{1}{2} \left(-h^{\mu\rho} + \eta^{\mu\rho}\right) \left(\partial_{x^{\sigma}} h_{\nu\rho} - \partial_{x^{\rho}} h_{\nu\sigma} + \partial_{x^{\nu}} h_{\sigma\rho}\right)$$

Neglect h compared to η

$$\Gamma^{\mu}{}_{\nu\sigma} == \frac{1}{2} \eta^{\mu\rho} \left(\partial_{x^{\sigma}} h_{\nu\rho} - \partial_{x^{\rho}} h_{\nu\sigma} + \partial_{x^{\nu}} h_{\sigma\rho}\right)$$

```
Print["2,4) Approximation for h(τ(t))"]
Print["Start with the derivative of the metric relation"]
(TotalD[τ, t])^2 == gdd[μ, ν] TotalD[xu[μ], t] TotalD[xu[ν], t] / c^2
Print["Substitute weak gravity metric"]
%% /. gdd[a_, b_] → ηdd[a, b] + hdd[a, b]
Print["Expand on time coordinate"]
%% // ExpandAll // PartialSum[0, {i, j}]
Print["Set ", xu[0] → c t];
%% /. xu[0] → c t
Print["Spatial velocities are small compared with time flow"]
%% /. TotalD[xu[i_], t] → 0
Print["Set ", ηdd[0, 0] → 1]
%% /. ηdd[0, 0] → 1
√# & /@ % // PowerExpand
Print["Take the first order Taylor series approximation."]
MapAt[Normal[# + O[hdd[0, 0]]^2] &, %%, 2] // TraditionalForm
Dτ = %[[2]];
Print["Taking the total derivative and expanding"]
TotalD[%%%, t]
MapAt[ExpandTotalD[labs, a], %, 2] /.
 PartialD[labs][hdd[a_, b_], xu[c_]] → hddd[a, b, Dif[c]]
Print["Partial sum on time component and putting ", xu[0] → c t]
%% // PartialSum[0, {i}]
% /. TotalD[xu[0], t] → TotalD[c t, t] // TraditionalForm
DDτ = %[[2]]
Print["Substitute expressions above"]
h[τ[t]] == TotalD[τ, {t, t}] / TotalD[τ, t]
h[τ[t]] == DDτ / Dτ
```

```
Print["Neglect h00 ≪ η00 = 1"]
%% /. hdd[0, 0] → 0
hApproxRule = Rule @@ %;
```

2,4) Approximation for h(τ(t))

Start with the derivative of the metric relation

$$Dt[\tau, t]^2 == \frac{g_{\mu\nu} \frac{dx^\mu}{dt} \frac{dx^\nu}{dt}}{c^2}$$

Substitute weak gravity metric

$$Dt[\tau, t]^2 == \frac{(h_{\mu\nu} + \eta_{\mu\nu}) \frac{dx^\mu}{dt} \frac{dx^\nu}{dt}}{c^2}$$

Expand on time coordinate

$$Dt[\tau, t]^2 == \frac{h_{00} \left(\frac{dx^0}{dt}\right)^2}{c^2} + \frac{\eta_{00} \left(\frac{dx^0}{dt}\right)^2}{c^2} + \frac{h_{i0} \frac{dx^0}{dt} \frac{dx^i}{dt}}{c^2} + \frac{\eta_{i0} \frac{dx^0}{dt} \frac{dx^i}{dt}}{c^2} + \frac{h_{0j} \frac{dx^0}{dt} \frac{dx^j}{dt}}{c^2} + \frac{\eta_{0j} \frac{dx^0}{dt} \frac{dx^j}{dt}}{c^2} + \frac{h_{ij} \frac{dx^i}{dt} \frac{dx^j}{dt}}{c^2} + \frac{\eta_{ij} \frac{dx^i}{dt} \frac{dx^j}{dt}}{c^2}$$

Set $x^0 \to c\, t$

$$Dt[\tau, t]^2 == h_{00} + \eta_{00} + \frac{h_{i0} \frac{dx^i}{dt}}{c} + \frac{\eta_{i0} \frac{dx^i}{dt}}{c} + \frac{h_{0j} \frac{dx^j}{dt}}{c} + \frac{\eta_{0j} \frac{dx^j}{dt}}{c} + \frac{h_{ij} \frac{dx^i}{dt} \frac{dx^j}{dt}}{c^2} + \frac{\eta_{ij} \frac{dx^i}{dt} \frac{dx^j}{dt}}{c^2}$$

Spatial velocities are small compared with time flow

$$Dt[\tau, t]^2 == h_{00} + \eta_{00}$$

Set $\eta_{00} \to 1$

$$Dt[\tau, t]^2 == 1 + h_{00}$$

$$Dt[\tau, t] == \sqrt{1 + h_{00}}$$

Take the first order Taylor series approximation.

$$\frac{d\tau}{dt} = \frac{1}{2} h_{00} + 1$$

Taking the total derivative and expanding

$$Dt[\tau, \{t, 2\}] == \frac{1}{2} \frac{dh_{00}}{dt}$$

$$Dt[\tau, \{t, 2\}] == \frac{1}{2} h_{00,a} \frac{dx^a}{dt}$$

Partial sum on time component and putting $x^0 \to c\, t$

$$Dt[\tau, \{t, 2\}] == \frac{1}{2} h_{00,0} \frac{dx^0}{dt} + \frac{1}{2} h_{00,i} \frac{dx^i}{dt}$$

$$\frac{d^2\tau}{dt^2} = \frac{1}{2} c\, h_{00,0} + \frac{1}{2} h_{00,i} \frac{dx^i}{dt}$$

$$\frac{1}{2} c\, h_{00,0} + \frac{1}{2} h_{00,i} \frac{dx^i}{dt}$$

Substitute expressions above

$$h[\tau[t]] == \frac{Dt[\tau, \{t, 2\}]}{Dt[\tau, t]}$$

$$h[\tau[t]] == \frac{\frac{1}{2} c\, h_{00,0} + \frac{1}{2} h_{00,i} \frac{dx^i}{dt}}{1 + \frac{1}{2} h_{00}}$$

Neglect $h_{00} \ll \eta_{00} = 1$

$$h[\tau[t]] == \frac{1}{2} c\, h_{00,0} + \frac{1}{2} h_{00,i} \frac{dx^i}{dt}$$

(David Park: "I have kept both terms here, departing slightly from FN. The first term is small because of the quasi-static approximation. The second term is small because of low spatial velocities.")

- We put here all together!

```
Print["3) Finally..."]
eqn[2, 77]
Print["Replace ", Γudd[i, j, k]]
%% /. ΓApproxRule
Print["Quasi-static field approximation and η^i0=0"]
%% /. hddd[_, _, Dif[0]] → 0 /. ρ → k
Print["Neglect approximation for h(τ(t))"]
%% /. hApproxRule
%[[1]] == 0
Print["Test particle mass m and rearranging"]
Map[# m c^2 &, %%, {2}]
eqn[2, 80] = (%[[1, 2]] == -%[[1, 1]] - %[[1, 3]]);
% // FrameBox // DisplayForm
Print["Compare with eqn[2,80], p.88."]
```

3) Finally...

$$\Gamma^i{}_{00} + \frac{\frac{d^2x^i}{dt\,dt}}{c^2} + \frac{2\,\Gamma^i{}_{0j} \frac{dx^j}{dt}}{c} == \frac{h[\tau[t]] \frac{dx^i}{dt}}{c^2}$$

Replace $\Gamma^i{}_{jk}$

$$\frac{1}{2} \left(-h_{00,\rho} + 2 h_{0\rho,0}\right) \eta^{i\rho} + \frac{\frac{d^2x^i}{dt\,dt}}{c^2} + \frac{\left(-h_{0j,\rho} + h_{0\rho,j} + h_{j\rho,0}\right) \eta^{i\rho} \frac{dx^j}{dt}}{c} == \frac{h[\tau[t]] \frac{dx^i}{dt}}{c^2}$$

Quasi-static field approximation and η^{i0}=0

$$-\frac{1}{2} h_{00,k}\, \eta^{ik} + \frac{\frac{d^2x^i}{dt\,dt}}{c^2} + \frac{\left(-h_{0j,k} + h_{0k,j}\right) \eta^{ik} \frac{dx^j}{dt}}{c} == \frac{h[\tau[t]] \frac{dx^i}{dt}}{c^2}$$

Neglect approximation for h(τ(t))

$$-\frac{1}{2} h_{00,k}\, \eta^{ik} + \frac{\frac{d^2x^i}{dt\,dt}}{c^2} + \frac{\left(-h_{0j,k} + h_{0k,j}\right) \eta^{ik} \frac{dx^j}{dt}}{c} == \frac{\frac{dx^i}{dt} \left(\frac{1}{2} c\, h_{00,0} + \frac{1}{2} h_{00,i} \frac{dx^i}{dt}\right)}{c^2}$$

$$-\frac{1}{2} h_{00,k} \eta^{ik} + \frac{\frac{d^2 x^i}{dt\,dt}}{c^2} + \frac{\left(-h_{0j,k} + h_{0k,j}\right) \eta^{ik} \frac{dx^j}{dt}}{c} == 0$$

Test particle mass m and rearranging

$$-\frac{1}{2} c^2 m\, h_{00,k} \eta^{ik} + m \frac{d^2 x^i}{dt\,dt} + c\, m \left(-h_{0j,k} + h_{0k,j}\right) \eta^{ik} \frac{dx^j}{dt} == 0$$

$$\boxed{m \frac{d^2 x^i}{dt\,dt} == \frac{1}{2} c^2 m\, h_{00,k} \eta^{ik} - c\, m \left(-h_{0j,k} + h_{0k,j}\right) \eta^{ik} \frac{dx^j}{dt}}$$

Compare with eqn[2,80], p.88.

The second rhs term "clearly smacks of rotation" (FN) and is zero in a *nonrotating* reference system:

```
eqnnr = Delete[eqn[2, 80], {2, 2}]
```

$$m \frac{d^2 x^i}{dt\,dt} == \frac{1}{2} c^2 m\, h_{00,k} \eta^{ik}$$

We can derive the rhs term from a gradient of a potential V:

```
(eqn[2, 82] = Tensor[V] == 1 / 2 c^2 hdd[0, 0] + const)
PartialD[#, k] & /@ eqn[2, 82]
Reverse[%] /. Equal → Rule;
Print["With ", %, " and intoducing the Kronecker delta we get finally"]
eqnnr /. %%;
(% /. %[[2, 3]] → -δuu[i, k]) // FrameBox // DisplayForm
Print["Compare with eqn[2,81], p.89."]
```

$$V == \text{const} + \frac{1}{2} c^2 h_{00}$$

$$V_{,k} == \frac{1}{2} c^2 h_{00,k}$$

With $\frac{1}{2} c^2 h_{00,k} \to V_{,k}$ and intoducing the Kronecker delta we get finally

$$\boxed{m \frac{d^2 x^i}{dt\,dt} == -m\, V_{,k}\, \delta^{ik}}$$

Compare with eqn[2,81], p.89.

This is the Newtonian equation of motion for a particle moving in a gravitational field of potential V, *if* we can make the identification $g_{00} = 1 + V / c^2$.

2.8 Newton's law of universal gravitation p. 89 - 90

```
Needs["TGeneralRelativity1`GeneralRelativity`"]
$PrePrint =.
labs = {x, δ, g, Γ};
DeclareBaseIndices[{0, 1, 2, 3}]
DefineTensorShortcuts[
 {{x}, 1},
 {{δ, g, η, h}, 2},
 {{Γ, h}, 3}]
SetTensorValueRules[ηuu[i, j],
 DiagonalMatrix[{1, Table[-1, {NDim - 1}] /. List → Sequence}]]
SchwarzschildCoordinates = {c t, r, θ, ϕ};
SetAttributes[{c, G, M}, Constant]
(* A little adjustment... *)
∂c t f[t]
(
 Unprotect[D];
 D[fun_, c t] := c^-1 D[fun, t];
 Protect[D];
)
%%
```

$\partial_{c\,t}\,f[t]$

$$\frac{f'[t]}{c}$$

FN: "Newton’s law of universal gravitation does not survive intact in general relativity, which is after all a new theory replacing the Newtonian theory. However, we should be able to recover it as an approximation. The Schwarzschild solution is an exact solution of the field equations of general relativity, and it may be identified as representing the field produced by a massive body."

```
Print["Jacobian matrix Δ for Cartesian ↔ spherical coordinates transformation"]
Λ = Outer[D, {c t, r Sin[θ] Cos[ϕ], r Sin[θ] Sin[ϕ], r Cos[θ]}, {c t, r, θ, ϕ}] // Simplify;
% // MatrixForm
InvΛ = Inverse[Λ] // Simplify;
% // MatrixForm
Print["A little test..."]
Transpose[Λ].DiagonalMatrix[{1, -1, -1, -1}].Λ // Simplify // MatrixForm
Transpose[InvΛ].%.InvΛ // Simplify // MatrixForm
```

Jacobian matrix Δ for Cartesian ↔ spherical coordinates transformation

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \text{Cos}[\phi]\,\text{Sin}[\theta] & r\,\text{Cos}[\theta]\,\text{Cos}[\phi] & -r\,\text{Sin}[\theta]\,\text{Sin}[\phi] \\ 0 & \text{Sin}[\theta]\,\text{Sin}[\phi] & r\,\text{Cos}[\theta]\,\text{Sin}[\phi] & r\,\text{Cos}[\phi]\,\text{Sin}[\theta] \\ 0 & \text{Cos}[\theta] & -r\,\text{Sin}[\theta] & 0 \end{pmatrix}$$

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \text{Cos}[\phi]\,\text{Sin}[\theta] & \text{Sin}[\theta]\,\text{Sin}[\phi] & \text{Cos}[\theta] \\ 0 & \frac{\text{Cos}[\theta]\,\text{Cos}[\phi]}{r} & \frac{\text{Cos}[\theta]\,\text{Sin}[\phi]}{r} & -\frac{\text{Sin}[\theta]}{r} \\ 0 & -\frac{\text{Csc}[\theta]\,\text{Sin}[\phi]}{r} & \frac{\text{Cos}[\phi]\,\text{Csc}[\theta]}{r} & 0 \end{pmatrix}$$

A little test...

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -r^2 & 0 \\ 0 & 0 & 0 & -r^2 \mathrm{Sin}[\theta]^2 \end{pmatrix}$$

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}$$

```
Print["Schwarzschild metric tensor gμν"]
SM = SchwarzschildMetric /. m → G c^-2 M;
gμν == (% // MatrixForm)
Print["Schwarzschild line element"]
(c Dt[τ])^2 == Dt[SchwarzschildCoordinates].SM.Dt[SchwarzschildCoordinates] //
 TraditionalForm
Print["Flat space metric η and line element in spherical coordinates"]
FSM = Transpose[Λ].DiagonalMatrix[{1, -1, -1, -1}].Λ // Simplify;
HoldForm[ημν spherical] == (% // MatrixForm)
(c Dt[τ])^2 == Dt[SchwarzschildCoordinates].FSM.Dt[SchwarzschildCoordinates] //
 TraditionalForm
Print["hμν=gμν-ημν (spherical base)"]
H = SM - FSM;
% // MatrixForm
Print["hμν (Cartesian base)"]
Transpose[InvΛ].H.InvΛ // Simplify;
SetTensorValues[hdd[μ, ν], %]
%% // MatrixForm
Print["hμν → 0 for large r"]
Limit[%%, r → Infinity] // MatrixForm
Print["This is eqn[2.80] from subsection 2.7:"]
Equal[Times[m, TotalD[Tensor[x, List[i], List[Void]], List[t, t]]],
 Plus[Times[Rational[1, 2], Power[c, 2], m, Tensor[h, List[Void, Void, Void],
    List[0, 0, Dif[k]]], Tensor[\[Eta], List[i,k], List[Void,Void]],
  Times[-1, c, m, Plus[Times[-1, Tensor[h, List[Void, Void, Void], List[0, j, Dif[k]]]],
    Tensor[h, List[Void, Void, Void], List[0, k, Dif[j]]]]], Tensor[\[Eta],
    List[i,k], List[Void,Void]], TotalD[Tensor[x, List[j], List[Void]], t]]]]
rint["Expand the derivatives"]
% // ExpandPartialD[labs]
rint["Expand to (spatial) components"]
% // ToArrayValues[{1, 2, 3}]
rint["r=√(x² + y² + z²) and use Cartesian coordinates symbols"]
% /. r → √(x² + y² + z²) // UseCoordinates[{c t, x, y, z}] // ColumnForm // TraditionalForm
```

Schwarzschild metric tensor $g_{\mu\nu}$

$$g_{\mu\nu} == \begin{pmatrix} 1-\frac{2GM}{c^2 r} & 0 & 0 & 0 \\ 0 & -\frac{1}{1-\frac{2GM}{c^2 r}} & 0 & 0 \\ 0 & 0 & -r^2 & 0 \\ 0 & 0 & 0 & -r^2 \mathrm{Sin}[\theta]^2 \end{pmatrix}$$

Schwarzschild line element

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

Flat space metric η and line element in spherical coordinates

$$\eta_{\mu\nu\,\text{spherical}} == \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -r^2 & 0 \\ 0 & 0 & 0 & -r^2 \operatorname{Sin}[\theta]^2 \end{pmatrix}$$

$$c^2 (d\tau)^2 = -(dr)^2 + c^2 (dt)^2 - r^2 (d\theta)^2 - r^2 (d\phi)^2 \sin^2(\theta)$$

$h_{\mu\nu}=g_{\mu\nu}-\eta_{\mu\nu}$ (spherical base)

$$\begin{pmatrix} -\frac{2GM}{c^2 r} & 0 & 0 & 0 \\ 0 & 1-\frac{1}{1-\frac{2GM}{c^2 r}} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$h_{\mu\nu}$ (Cartesian base)

$$\begin{pmatrix} -\frac{2GM}{c^2 r} & 0 & 0 & 0 \\ 0 & \frac{2GM\operatorname{Cos}[\phi]^2 \operatorname{Sin}[\theta]^2}{2GM-c^2 r} & \frac{GM\operatorname{Sin}[\theta]^2 \operatorname{Sin}[2\phi]}{2GM-c^2 r} & \frac{GM\operatorname{Cos}[\phi]\operatorname{Sin}[2\theta]}{2GM-c^2 r} \\ 0 & \frac{GM\operatorname{Sin}[\theta]^2 \operatorname{Sin}[2\phi]}{2GM-c^2 r} & \frac{2GM\operatorname{Sin}[\theta]^2 \operatorname{Sin}[\phi]^2}{2GM-c^2 r} & \frac{GM\operatorname{Sin}[2\theta]\operatorname{Sin}[\phi]}{2GM-c^2 r} \\ 0 & \frac{GM\operatorname{Cos}[\phi]\operatorname{Sin}[2\theta]}{2GM-c^2 r} & \frac{GM\operatorname{Sin}[2\theta]\operatorname{Sin}[\phi]}{2GM-c^2 r} & \frac{2GM\operatorname{Cos}[\theta]^2}{2GM-c^2 r} \end{pmatrix}$$

$h_{\mu\nu} \to 0$ for large r

$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

This is eqn[2.80] from subsection 2.7:

$$m \frac{d^2 x^i}{dt\,dt} == \frac{1}{2} c^2 m\, h_{00,k}\, \eta^{ik} - c\, m \left(-h_{0j,k} + h_{0k,j}\right) \eta^{ik} \frac{dx^j}{dt}$$

Expand the derivatives

$$m \frac{d^2 x^i}{dt\,dt} == \frac{G\, m\, M\, \eta^{ik}\, \partial_{x^k} r}{r^2} - c\, m\, \eta^{ik} \frac{dx^j}{dt} \left(-\partial_{x^k} h_{0j} + \partial_{x^j} h_{0k}\right)$$

Expand to (spatial) components

$$\left\{m \frac{d^2 x^1}{dt\,dt} == -\frac{G\, m\, M\, \partial_{x^1} r}{r^2},\ m \frac{d^2 x^2}{dt\,dt} == -\frac{G\, m\, M\, \partial_{x^2} r}{r^2},\ m \frac{d^2 x^3}{dt\,dt} == -\frac{G\, m\, M\, \partial_{x^3} r}{r^2}\right\}$$

$r=\sqrt{x^2+y^2+z^2}$ and use Cartesian coordinates symbols

$$m \frac{d^2 x}{dt^2} = -\frac{G\, m\, M\, x}{\left(x^2+y^2+z^2\right)^{3/2}}$$

$$m \frac{d^2 y}{dt^2} = -\frac{G\, m\, M\, y}{\left(x^2+y^2+z^2\right)^{3/2}}$$

$$m \frac{d^2 z}{dt^2} = -\frac{G\, m\, M\, z}{\left(x^2+y^2+z^2\right)^{3/2}}$$

This is eqn[2.80] from subsection 2.7:

$$m \frac{d^2 x^i}{dt\,dt} == \frac{1}{2} c^2 m\, h_{00,k}\, \eta^{ik} - c\, m \left(-h_{0j,k} + h_{0k,j}\right) \eta^{ik} \frac{dx^j}{dt}$$

Expand the derivatives

$$m \frac{d^2 x^i}{dt\,dt} == \frac{G\,m\,M\,\eta^{ik}\,\partial_{x^k} r}{r^2} - c\,m\,\eta^{ik}\,\frac{dx^j}{dt}\left(-\partial_{x^k} h_{0j} + \partial_{x^j} h_{0k}\right)$$

Expand to (spatial) components

$$\left\{m \frac{d^2 x^1}{dt\,dt} == -\frac{G\,m\,M\,\partial_{x^1} r}{r^2},\ m \frac{d^2 x^2}{dt\,dt} == -\frac{G\,m\,M\,\partial_{x^2} r}{r^2},\ m \frac{d^2 x^3}{dt\,dt} == -\frac{G\,m\,M\,\partial_{x^3} r}{r^2}\right\}$$

$r = \sqrt{x^2 + y^2 + z^2}$ and use Cartesian coordinates symbols

$$m \frac{d^2 x}{dt^2} = -\frac{G\,m\,M\,x}{\left(x^2+y^2+z^2\right)^{3/2}}$$

$$m \frac{d^2 y}{dt^2} = -\frac{G\,m\,M\,y}{\left(x^2+y^2+z^2\right)^{3/2}}$$

$$m \frac{d^2 z}{dt^2} = -\frac{G\,m\,M\,z}{\left(x^2+y^2+z^2\right)^{3/2}}$$

This seems to be typically Newtonian... Those we recovered the classical Newton's law from the general relativistic Schwarzschild solution.

2.9 A rotating reference system p. 90 - 93

```
Needs["TensorCalculus3`Tensorial`"]
$PrePrint = .
labs = {x, δ, g, Γ};
DeclareBaseIndices[{0, 1, 2, 3}]
DefineTensorShortcuts[
 {{x, dx, zero}, 1},
 {{δ, g, Λ}, 2},
 {{Γ}, 3}]
DeclareZeroTensor[zero]
MyRed = StyleForm[Superscript[#, "′"], FontColor → RGBColor[1, 0, 0]] &;
DeclareIndexFlavor[{red, MyRed}]
SetAttributes[c, Constant]
(
 Unprotect[D];
 D[fun_, c t] := c^-1 D[fun, t];
 Protect[D];
)
```

FN: "The principle of equivalence (see the Introduction) implies that the “fictitious” forces of accelerating coordinate systems are essentially in the same category as the “real” forces of gravity. Put another way, if the geodesic equation contains gravity in the $\Gamma^{\mu}_{\nu\sigma}$ it must also contain any accelerations which may have been built in by choice of coordinate system. In a curved space-time it is not always easy, and often impossible, to sort these forces out, but in flat spacetime we have only the fictitious forces of acceleration and these should be included in the $\Gamma^{\mu}_{\nu\sigma}$. As an example of this, let us consider a rotating reference system in flat spacetime."

```
Print["Rotating' to nonrotating coordinate transformation"]
xu[red@a] → xu[a]
SetTensorValues[xu[red@a], {c t, x, y, z}]
cTXYZ = {c t, x Cos[ω t] - y Sin[ω t], x Sin[ω t] + y Cos[ω t], z};
SetTensorValues[xu[a], cTXYZ]
(eqn[2, 85] = ({c T, X, Y, Z} == cTXYZ // Thread)) // TableForm // FrameBox // DisplayForm
```

```
Print["Jacobian matrix"]
Λud[a, red@b] == PartialD[xu[a], red@b]
Λmat = %[[2]] // ExpandPartialD[labs] // ToArrayValues[];
Λmat // MatrixForm
SetTensorValues[Λud[a, red@b], Λmat]
SetTensorValues[Λud[red@a, b], Inverse[Λmat] // Simplify]
Print["Nonrotating metric (Minkowski metric) to rotating' metric"]
SetMetricValues[g, DiagonalMatrix[{1, -1, -1, -1}]]
gdd[red@a, red@b] == Λud[i, red@a] Λud[j, red@b] gdd[i, j]
cmetric = %[[2]] // ToArrayValues[] // Simplify;
% // MatrixForm
ClearTensorValues[{xu[red@a], xu[a]}]
(* Achtung mit c t → x^0! *)
metric = cmetric // CoordinatesToTensors[{c t, x, y, z}, x, red];
% // MatrixForm
SetMetricValues[g, metric, red]
Print["Calculate the Christoffel symbols in the rotating' system"]
christoffel = CalculateChristoffelu[xu[red@i], metric, Simplify];
SetTensorValueRules[Γudd[a, b, c] // ToFlavor[red], %];
SelectedTensorRules[Γ, Γudd[_, a_, b_] /; OrderedQ[{a, b}]] // TableForm
SetTensorValues[Γudd[a, b, c] // ToFlavor[red], christoffel]
```

Rotating' to nonrotating coordinate transformation

$x^{a'} \to x^{a}$

$$\begin{array}{|l|}\hline c\,T == c\,t \\ X == x\,\mathrm{Cos}[t\,\omega] - y\,\mathrm{Sin}[t\,\omega] \\ Y == y\,\mathrm{Cos}[t\,\omega] + x\,\mathrm{Sin}[t\,\omega] \\ Z == z \\ \hline\end{array}$$

Jacobian matrix

$\Lambda^{a}{}_{b'} == x^{a}{}_{,b'}$

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ \frac{-y\,\omega\,\mathrm{Cos}[t\,\omega] - x\,\omega\,\mathrm{Sin}[t\,\omega]}{c} & \mathrm{Cos}[t\,\omega] & -\mathrm{Sin}[t\,\omega] & 0 \\ \frac{x\,\omega\,\mathrm{Cos}[t\,\omega] - y\,\omega\,\mathrm{Sin}[t\,\omega]}{c} & \mathrm{Sin}[t\,\omega] & \mathrm{Cos}[t\,\omega] & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Nonrotating metric (Minkowski metric) to rotating' metric

$g_{a'b'} == g_{ij}\,\Lambda^{i}{}_{a'}\,\Lambda^{j}{}_{b'}$

$$\begin{pmatrix} \frac{c^2 - (x^2+y^2)\,\omega^2}{c^2} & \frac{y\,\omega}{c} & -\frac{x\,\omega}{c} & 0 \\ \frac{y\,\omega}{c} & -1 & 0 & 0 \\ -\frac{x\,\omega}{c} & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}$$

$$\begin{pmatrix} \frac{c^2 - \omega^2\left((x^{1'})^2 + (x^{2'})^2\right)}{c^2} & \frac{\omega\,x^{2'}}{c} & -\frac{\omega\,x^{1'}}{c} & 0 \\ \frac{\omega\,x^{2'}}{c} & -1 & 0 & 0 \\ -\frac{\omega\,x^{1'}}{c} & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}$$

Calculate the Christoffel symbols in the rotating' system

$\Gamma^{1'}{}_{0'0'} \to -\frac{\omega^2 x^{1'}}{c^2}$

$\Gamma^{1'}{}_{0'2'} \to -\frac{\omega}{c}$

$\Gamma^{2'}{}_{0'0'} \to -\frac{\omega^2 x^{2'}}{c^2}$

$\Gamma^{2'}{}_{0'1'} \to \frac{\omega}{c}$

```
Print["Geodesic equation in the rotating system"]
TotalD[xu[i], {τ, τ}] + Γudd[i, j, k] TotalD[xu[j], τ] TotalD[xu[k], τ] == zerou[i] //
  ToFlavor[red]
Print["Expanding and using coordinates"]
%% // ToArrayValues[]
% // UseCoordinates[{c t, x, y, z}, x, red];
MapAt[# / c &, %, {{1, 1}, {1, 2}}];
(eqn[2, 87] = %) // TableForm // FrameBox // DisplayForm // TraditionalForm
Print["dt/dτ = constant and mass m"]
(eqn[2, 88] = Distribute[m #] & /@ # & /@ Drop[eqn[2, 87], 1] /. τ → t) // TableForm //
    FrameBox // DisplayForm // TraditionalForm
Print["Compare with eqn[2,88], p.91."]
```

Geodesic equation in the rotating system

$$\frac{d^2x^{i'}}{d\tau d\tau} + \Gamma^{i'}{}_{j'k'} \frac{dx^{j'}}{d\tau} \frac{dx^{k'}}{d\tau} == \text{zero}^{i'}$$

Expanding and using coordinates

$$\left\{\frac{d^2x^{0'}}{d\tau d\tau} == 0,\ -\frac{\omega^2 x^{1'}\left(\frac{dx^{0'}}{d\tau}\right)^2}{c^2} + \frac{d^2x^{1'}}{d\tau d\tau} - \frac{2\,\omega \frac{dx^{0'}}{d\tau} \frac{dx^{2'}}{d\tau}}{c} == 0,\right.$$
$$\left. -\frac{\omega^2 x^{2'}\left(\frac{dx^{0'}}{d\tau}\right)^2}{c^2} + \frac{2\,\omega \frac{dx^{0'}}{d\tau} \frac{dx^{1'}}{d\tau}}{c} + \frac{d^2x^{2'}}{d\tau d\tau} == 0,\ \frac{d^2x^{3'}}{d\tau d\tau} == 0\right\}$$

$$\begin{array}{|l|}\hline \frac{d^2t}{d\tau^2} = 0 \\ -x\,\omega^2\left(\frac{dt}{d\tau}\right)^2 - 2\,\omega\,\frac{dy}{d\tau}\,\frac{dt}{d\tau} + \frac{d^2x}{d\tau^2} = 0 \\ -y\,\omega^2\left(\frac{dt}{d\tau}\right)^2 + 2\,\omega\,\frac{dx}{d\tau}\,\frac{dt}{d\tau} + \frac{d^2y}{d\tau^2} = 0 \\ \frac{d^2z}{d\tau^2} = 0 \\ \hline\end{array}$$

dt/dτ = constant and mass m

$$\begin{array}{|l|}\hline -m\,x\,\omega^2 - 2\,m\,\frac{dy}{dt}\,\omega + m\,\frac{d^2x}{dt^2} = 0 \\ -m\,y\,\omega^2 + 2\,m\,\frac{dx}{dt}\,\omega + m\,\frac{d^2y}{dt^2} = 0 \\ m\,\frac{d^2z}{dt^2} = 0 \\ \hline\end{array}$$

Compare with eqn[2,88], p.91.

We compare now this result with the classical formula giving the transformation of the force m·b measured in an inertial system Σ to the force m·b' seen by an non-inertial system Σ'.

```
b = {0, 0, 0};
B = {0, 0, 0};
Ω = {0, 0, ω}; SetAttributes[ω, Constant]
```

```
DΩ = TotalD[#, t] & /@ Ω;
rp = {x, y, z};
vp = TotalD[#, t] & /@ rp;
bp = TotalD[#, {t, t}] & /@ rp;
Print["Classical vector equation for the forces in a non-inertial system"]
HoldForm[m bp - (m b - m B - m (DΩ × rp) - m Ω × (Ω × rp) - 2 m Ω × vp) == 0]
Print["With our conditions this gives in coordinate form..."]
Thread[ReleaseHold[%%]];
% // TableForm // TraditionalForm
Print["Equal to eqn[2,88]?"];
%%% === eqn[2, 88]
Clear[a, b]
```

Classical vector equation for the forces in a non-inertial system

m bp - (m b - m B - m DΩ × rp - m Ω × (Ω × rp) - 2 m Ω × vp) == 0

With our conditions this gives in coordinate form...

$$-m\,x\,\omega^2 - 2\,m\,\frac{dy}{dt}\,\omega + m\,\frac{d^2x}{dt^2} = 0$$

$$-m\,y\,\omega^2 + 2\,m\,\frac{dx}{dt}\,\omega + m\,\frac{d^2y}{dt^2} = 0$$

$$m\,\frac{d^2z}{dt^2} = 0$$

Equal to eqn[2,88]?

True

Exercise 2.9.1 p.93.

```
c^2 dτ^2 == gdd[μ, ν] dxu[μ] dxu[ν] // ToFlavor[red]
(eqn[2, 86] = % // EinsteinSum[] // UseCoordinates[{c dt, dx, dy, dz}, dx, red] //
    UseCoordinates[{c t, x, y, z}, x, red]) // FrameBox // DisplayForm
guu[red@a, red@b] // EinsteinArray[] // UseCoordinates[{c t, x, y, z}, x, red];
guu[red@a, red@b] == HoldForm[c^-2] MatrixForm[% c^2 // Simplify] // FrameBox // DisplayForm
```

$$c^2\, d\tau^2 == dx^{\mu'}\, dx^{\nu'}\, g_{\mu'\nu'}$$

$$c^2\, d\tau^2 == -dx^2 - dy^2 - dz^2 - 2\, dt\, dy\, x\, \omega + 2\, dt\, dx\, y\, \omega + dt^2\left(c^2 - \left(x^2 + y^2\right)\omega^2\right)$$

$$g^{a'b'} == \frac{1}{c^2}\begin{pmatrix} c^2 & c\,y\,\omega & -c\,x\,\omega & 0 \\ c\,y\,\omega & -c^2 + y^2\,\omega^2 & -x\,y\,\omega^2 & 0 \\ -c\,x\,\omega & -x\,y\,\omega^2 & -c^2 + x^2\,\omega^2 & 0 \\ 0 & 0 & 0 & -c^2 \end{pmatrix}$$

- **Problem 2.8 p.94.** One can conceive of an observer in a swivel chair located above the Sun, looking down on the plane of the Earth's orbit. If the chair rotates at the rate of one revolution a year, then to the observer the Earth appears stationary. If for some reason all heavenly bodies other that the Earth and the Sun are invisible, how does the observer explain why the Earth does not collapse in towards the Sun, there being no detectable orbit?

Chapter 3: Field equations and curvature

3.0 Introduction p. 97

FN: "The main purpose of this chapter is to establish the field equations of general relativity, which couple the gravitational field (contained in the curvature of spacetime) with its sources. We start by discussing a tensor which effectively and concisely describes the sources, and follow that with a discussion of curvature, then bring these together in the field equations. [...] The chapter finishes with an exact solution of the field equations representing the gravitational field of spherically symmetric massive body."

3.1 The stress tensor and fluid motion p. 97 - 102

```
Needs["TGeneralRelativity1`GeneralRelativity`"]
$PrePrint =.
labs = {x, δ, g, Γ};
DeclareBaseIndices[{0, 1, 2, 3}]
DefineTensorShortcuts[
 {{x, λ, u, v, p, ρ, zero}, 1},
 {{δ, g, T, η, δ}, 2},
 {{Γ}, 3}]
DeclareZeroTensor[zero]
SetAttributes[c, Constant]
FlatToCurvedSpacetime = {η → g, TotalD → AbsoluteD, Dif → Cov};
Print["Two needed 4-velocity relations as rules:"]
uu[ν] ud[ν] == c^2
usquare = Rule @@ % // LHSSymbolsToPatterns[{ν}]
PartialD[#, μ] & /@ %%
% /. PartialD[c^2, μ] → 0
Print["Using UpDownSwap on first term and turning it into a rule"]
MapAt[UpDownSwap[ν], %%, {1, 1}]
# / 2 & /@ %;
uidentity = Rule @@ % // LHSSymbolsToPatterns[{μ, ν}]
Print["Another little adjustment..."]
PartialD[Tensor[f], 0] // ExpandPartialD[labs] //
  UseCoordinates[{c t, x, y, z}] // TraditionalForm
(
 Unprotect[PartialD];
 PartialD[l_][tensor_, c t] := c^-1 PartialD[l][tensor, t];
 Protect[PartialD];
)
%%
```

Two needed 4-velocity relations as rules:

$u_\nu u^\nu == c^2$

$u_{\nu\_} u^{\nu\_} \to c^2$

$u^\nu u_{\nu,\mu} + u_\nu u^\nu{}_{,\mu} ==$ PartialD$[c^2, \mu]$

$u^\nu u_{\nu,\mu} + u_\nu u^\nu{}_{,\mu} == 0$

Using UpDownSwap on first term and turning it into a rule

$2 u_\nu u^\nu{}_{,\mu} == 0$

$u_{\nu\_} u^{\nu\_}{}_{,\mu\_} \to 0$

Another little adjustment...

$$\frac{\partial f}{\partial c t}$$

$$\frac{\partial_t f}{c}$$

■ Expanding the temporal and spatial parts of a tensor expressions

We can expand a 4-vector as follows...

```
λu[μ]
λu[μ] // EinsteinArray[]
λu[μ] // EinsteinArray[{0}]
λu[μ] // EinsteinArray[{1, 2, 3}]
λu[μ] // PartialArray[0, {i}]
MapAt[EinsteinArray[{1, 2, 3}], %, 2]
({λ0, λs} = %%) // FullForm
λs // EinsteinArray[{1, 2, 3}]
```

λ^μ

$\{\lambda^0, \lambda^1, \lambda^2, \lambda^3\}$

$\{\lambda^0\}$

$\{\lambda^1, \lambda^2, \lambda^3\}$

$\{\lambda^0, \lambda^i\}$

$\{\lambda^0, \{\lambda^1, \lambda^2, \lambda^3\}\}$

List[Tensor[\[Lambda], List[0], List[Void]], Tensor[\[Lambda], List[i], List[Void]]]

$\{\lambda^1, \lambda^2, \lambda^3\}$

Mixed expansions. The number of separate parts goes as 2^n where n is the number of free indices plus the number of dummy pairs.

```
Tuu[μ, ν] λd[ν]
% // PartialArray[0, {i}]
% // PartialSum[0, {j}]
%% // EinsteinSum[]
```

$T^{\mu\nu}\,\lambda_\nu$

$\{T^{0\nu}\,\lambda_\nu,\ T^{i\nu}\,\lambda_\nu\}$

$\{T^{00}\,\lambda_0 + T^{0j}\,\lambda_j,\ T^{i0}\,\lambda_0 + T^{ij}\,\lambda_j\}$

$\{T^{00}\,\lambda_0 + T^{01}\,\lambda_1 + T^{02}\,\lambda_2 + T^{03}\,\lambda_3,\ T^{i0}\,\lambda_0 + T^{i1}\,\lambda_1 + T^{i2}\,\lambda_2 + T^{i3}\,\lambda_3\}$

■ Quantities used in the relativistic description of particle and fluid dynamics

m = proper mass of a particle. (A scalar)
t = coordinate time. (Not a scalar but a component of the coordinates)
τ = proper time. (A scalar. "Proper" actually comes from the French and means "own". The time that the particle sees.)
$\gamma = dt / d\tau = 1 \big/ \sqrt{1 - v^2 / c^2}$ where v is the particle's speed. (Not a scalar)
$\mathcal{E} = \gamma\, m\, c^2$ = energy of particle (Not a scalar but the first component of the energy-momentum tensor. We can't use E in Mathematica.)
$u^\mu = dx^\mu \big/ d\tau$ = world velocity. (A vector)
$v^\mu = dx^\mu \big/ dt = u^\mu / \gamma$ = coordinate velocity. (Not a vector)
$p^\mu = m\, u^\mu$ = 4-momentum of the particle. (A vector. Also called "momergy", by me.)
ρ = proper density of a continuous distribution of matter. (A scalar)
P = pressure

```
Print["Coordinate velocity for a particle ", vu[μ]]
vu[μ] // PartialArray[0, {i}]
% /. vu[μ_] → TotalD[xu[μ], t]
MapAt[EinsteinArray[{1, 2, 3}], %, 2]
% // UseCoordinates[{c t, x, y, z}] // TraditionalForm
```

Coordinate velocity for a particle v^μ

$\{v^0, v^i\}$

$\left\{\frac{dx^0}{dt}, \frac{dx^i}{dt}\right\}$

$\left\{\frac{dx^0}{dt}, \left\{\frac{dx^1}{dt}, \frac{dx^2}{dt}, \frac{dx^3}{dt}\right\}\right\}$

$\left\{c, \left\{\frac{dx}{dt}, \frac{dy}{dt}, \frac{dz}{dt}\right\}\right\}$

```
Print["4-velocity for a particle ", uu[μ]]
uu[μ] // PartialArray[0, {i}]
% /. uu[μ_] → TotalD[xu[μ], τ]
% /. TotalD[xu[μ_], τ] → TotalD[t, τ] TotalD[xu[μ], t]
% /. TotalD[t, τ] → γ
MapAt[EinsteinArray[{1, 2, 3}], %, 2]
% // UseCoordinates[{c t, x, y, z}] // TraditionalForm
% /. γ → 1/Sqrt[1 - v^2/c^2] // TraditionalForm
```

4-velocity for a particle u^μ

$\{u^0, u^i\}$

$$\left\{\frac{dx^0}{d\tau}, \frac{dx^i}{d\tau}\right\}$$

$$\left\{Dt[t, \tau]\,\frac{dx^0}{dt}, Dt[t, \tau]\,\frac{dx^i}{dt}\right\}$$

$$\left\{\gamma\,\frac{dx^0}{dt}, \gamma\,\frac{dx^i}{dt}\right\}$$

$$\left\{\gamma\,\frac{dx^0}{dt}, \left\{\gamma\,\frac{dx^1}{dt}, \gamma\,\frac{dx^2}{dt}, \gamma\,\frac{dx^3}{dt}\right\}\right\}$$

$$\left\{c\,\gamma, \left\{\gamma\,\frac{dx}{dt}, \gamma\,\frac{dy}{dt}, \gamma\,\frac{dz}{dt}\right\}\right\}$$

$$\left\{\frac{c}{\sqrt{1-\frac{v^2}{c^2}}}, \left\{\frac{\frac{dx}{dt}}{\sqrt{1-\frac{v^2}{c^2}}}, \frac{\frac{dy}{dt}}{\sqrt{1-\frac{v^2}{c^2}}}, \frac{\frac{dz}{dt}}{\sqrt{1-\frac{v^2}{c^2}}}\right\}\right\}$$

```
Print["4-momentum for a particle"]
pu[μ] == m uu[μ]
Print["Definition of u and use of coordinate time"]
%% /. uu[μ_] → TotalD[xu[μ], τ]
% /. TotalD[xu[μ_], τ] → TotalD[t, τ] TotalD[xu[μ], t]
Print["Definition of γ and v^μ"]
%% /. TotalD[t, τ] → γ /. TotalD[xu[μ_], t] → vu[μ]
Print["Broken into time and space componentp"]
MapAt[PartialArray[0, {i}], %%, 2]
Print["Definition of ", vu[0], " and substituting coordinates"]
%% /. {vu[0] → TotalD[xu[0], t]}
% // UseCoordinates[{c t, x, y, z}]
Print["Substituting definitions for energy ε and spatial momentum p"]
%% /. {m γ vu[i_] → p[i], m γ → ε / c²}
```

4-momentum for a particle

$$p^\mu == m\,u^\mu$$

Definition of u and use of coordinate time

$$p^\mu == m\,\frac{dx^\mu}{d\tau}$$

$$p^\mu == m\,Dt[t, \tau]\,\frac{dx^\mu}{dt}$$

Definition of γ and v^μ

$$p^\mu == m\,\gamma\,v^\mu$$

Broken into time and space componentp

$$p^\mu == \left\{m\,\gamma\,v^0, m\,\gamma\,v^i\right\}$$

Definition of v^0 and substituting coordinates

$$p^\mu == \left\{m\,\gamma\,\frac{dx^0}{dt}, m\,\gamma\,v^i\right\}$$

$p^\mu == \{c\, m\, \gamma,\ m\, \gamma\, v^i\}$

Substituting definitions for energy $\mathcal{E}$ and spatial momentum p

$$p^\mu == \left\{\frac{\mathcal{E}}{c},\ p^i\right\}$$

■ The (special) relativistic energy-momentum-stress tensor for a perfect fluid

```
Print["Energy-momentum-stress tensor T^μν for a perfect fluid (definition (3.2), p.99)"]
(eqn[3, 2] = Tuu[μ, ν] == (Tensor[ρ] + Tensor[P] / c^2) uu[μ] uu[ν] - Tensor[P] ηuu[μ, ν]) //
  FrameBox // DisplayForm
Trule = Rule @@ eqn[3, 2] // LHSSymbolsToPatterns[{μ, ν}]
```

Energy-momentum-stress tensor $T^{\mu\nu}$ for a perfect fluid (definition (3.2), p.99)

$$\boxed{T^{\mu\nu} == \left(\frac{P}{c^2} + \rho\right) u^\mu\, u^\nu - P\, \eta^{\mu\nu}}$$

$$T^{\mu\_\nu\_} \to \left(\frac{P}{c^2} + \rho\right) u^\mu\, u^\nu - P\, \eta^{\mu\nu}$$

```
Print["Contracting T^μν with ", ud[ν]]
Tuu[μ, ν] ud[ν] == (Tuu[μ, ν] ud[ν] /. Trule)
% // ExpandAll
Print["Using ", usquare]
%% /. usquare
Print["Metric simplify"]
%% // MetricSimplify[η]
Print["Compare with the 4-momentum density:"]
Tensor[ρ] uu[μ]
```

Contracting $T^{\mu\nu}$ with u_ν

$$T^{\mu\nu}\, u_\nu == u_\nu \left(\left(\frac{P}{c^2} + \rho\right) u^\mu\, u^\nu - P\, \eta^{\mu\nu}\right)$$

$$T^{\mu\nu}\, u_\nu == \frac{P\, u_\nu\, u^\mu\, u^\nu}{c^2} + \rho\, u_\nu\, u^\mu\, u^\nu - P\, u_\nu\, \eta^{\mu\nu}$$

Using $u_{\nu\_}\, u^{\nu\_} \to c^2$

$$T^{\mu\nu}\, u_\nu == P\, u^\mu + c^2\, \rho\, u^\mu - P\, u_\nu\, \eta^{\mu\nu}$$

Metric simplify

$$T^{\mu\nu}\, u_\nu == c^2\, \rho\, u^\mu$$

Compare with the 4-momentum density:

$\rho\, u^\mu$

■ The divergence of the energy-momentum-stress tensor

The divergence of $T^{\mu\nu}$ leads to the **continuity equation** and the **equation of motion of the perfect fluid**. (Because of the symmetry of $T^{\mu\nu}$ there is only one divergence.)

```
Print["Setting T^μν,μ to 0"]
eqn[3, 2, bis] =
 PartialD[Tuu[μ, ν], μ] == (PartialD[NestedTensor[Tuu[μ, ν] /. Trule], μ]) == 0
```

```
Print["Expanding and using constancy of η"]
%% // UnnestTensor
% /. PartialD[ηuu[_, _], _] → 0
Print["Dropping first equation and expanding"]
eqn[3, 3] = (Drop[%%, 1] // ExpandAll)
```

Setting $T^{\mu\nu}{}_{,\mu}$ to 0

$$T^{\mu\nu}{}_{,\mu} == \left(\left(\frac{P}{c^2}+\rho\right) u^\mu u^\nu - P\,\eta^{\mu\nu}\right)_{,\mu} == 0$$

Expanding and using constancy of η

$$T^{\mu\nu}{}_{,\mu} == \left(\frac{P}{c^2}+\rho\right)\left(u^\nu u^\mu{}_{,\mu} + u^\mu u^\nu{}_{,\mu}\right) - P_{,\mu}\,\eta^{\mu\nu} - P\,\eta^{\mu\nu}{}_{,\mu} + u^\mu u^\nu \left(\frac{P_{,\mu}}{c^2}+\rho_{,\mu}\right) == 0$$

$$T^{\mu\nu}{}_{,\mu} == \left(\frac{P}{c^2}+\rho\right)\left(u^\nu u^\mu{}_{,\mu} + u^\mu u^\nu{}_{,\mu}\right) - P_{,\mu}\,\eta^{\mu\nu} + u^\mu u^\nu \left(\frac{P_{,\mu}}{c^2}+\rho_{,\mu}\right) == 0$$

Dropping first equation and expanding

$$\frac{P_{,\mu}\,u^\mu u^\nu}{c^2} + \frac{P\,u^\nu u^\mu{}_{,\mu}}{c^2} + \rho\,u^\nu u^\mu{}_{,\mu} + \frac{P\,u^\mu u^\nu{}_{,\mu}}{c^2} + \rho\,u^\mu u^\nu{}_{,\mu} - P_{,\mu}\,\eta^{\mu\nu} + u^\mu u^\nu \rho_{,\mu} == 0$$

```
Print["I) Equation of continuity"]
Print["Contracting T^μν_,μ with ", ud[ν]]
Distribute[# ud[ν]] & /@ eqn[3, 3]
Print["Using ", usquare, " and ", uidentity]
%% /. {usquare, uidentity}
MapAt[MapLevelParts[MetricSimplify[η], {{3, 4}}], %, {1, 4}]
eqn[3, 5, 1] = %;
# / c^2 & /@ %% // Simplify
Print["Check proposed simplification"]
(eqn[3, 5] = First[%%[[1]]] +
    PartialD[NestedTensor[Tensor[ρ] uu[μ]], μ] == 0)
%%% == (% // UnnestTensor)
```

I) *Equation of continuity*

Contracting $T^{\mu\nu}{}_{,\mu}$ with u_ν

$$\frac{P_{,\mu}\,u_\nu u^\mu u^\nu}{c^2} + \frac{P\,u_\nu u^\nu u^\mu{}_{,\mu}}{c^2} + \rho\,u_\nu u^\nu u^\mu{}_{,\mu} + \frac{P\,u_\nu u^\mu u^\nu{}_{,\mu}}{c^2} + \rho\,u_\nu u^\mu u^\nu{}_{,\mu} - P_{,\mu}\,u_\nu\,\eta^{\mu\nu} + u_\nu u^\mu u^\nu \rho_{,\mu} == 0$$

Using $u_{\nu\_}\,u^{\nu\_} \to c^2$ and $u_{\nu\_}\,u^{\nu\_}{}_{,\mu\_} \to 0$

$$P_{,\mu}\,u^\mu + P\,u^\mu{}_{,\mu} + c^2\,\rho\,u^\mu{}_{,\mu} - P_{,\mu}\,u_\nu\,\eta^{\mu\nu} + c^2\,u^\mu\,\rho_{,\mu} == 0$$

$$P\,u^\mu{}_{,\mu} + c^2\,\rho\,u^\mu{}_{,\mu} + c^2\,u^\mu\,\rho_{,\mu} == 0$$

$$\frac{P\,u^\mu{}_{,\mu}}{c^2} + \rho\,u^\mu{}_{,\mu} + u^\mu\,\rho_{,\mu} == 0$$

Check proposed simplification

$$\frac{P\,u^\mu{}_{,\mu}}{c^2} + \left(\rho\,u^\mu\right)_{,\mu} == 0$$

True

Compare with equation (3.5), p.99.

```
Print["II) Equation of motion"]
Print["Factoring the set of terms of T^μν,μ
   that contains the continuity relation and setting it to zero"]
eqn[3, 3]
% // MapLevelParts[Factor, {1, {2, 3, 7}}]
% /. Rule @@ eqn[3, 5, 1]
Print["Factoring terms and rearranging"]
# c² & /@ %% // Simplify
eqn[3, 6] = (%[[1, 1]] == -%[[1, 2]])
```

II) Equation of motion

Factoring the set of terms of $T^{\mu\nu}{}_{,\mu}$ that contains the continuity relation and setting it to zero

$$\frac{P_{,\mu}\, u^\mu\, u^\nu}{c^2} + \frac{P\, u^\nu\, u^\mu{}_{,\mu}}{c^2} + \rho\, u^\nu\, u^\mu{}_{,\mu} + \frac{P\, u^\mu\, u^\nu{}_{,\mu}}{c^2} + \rho\, u^\mu\, u^\nu{}_{,\mu} - P_{,\mu}\, \eta^{\mu\nu} + u^\mu\, u^\nu\, \rho_{,\mu} == 0$$

$$\frac{P_{,\mu}\, u^\mu\, u^\nu}{c^2} + \frac{P\, u^\mu\, u^\nu{}_{,\mu}}{c^2} + \rho\, u^\mu\, u^\nu{}_{,\mu} - P_{,\mu}\, \eta^{\mu\nu} + \frac{u^\nu \left(P\, u^\mu{}_{,\mu} + c^2\, \rho\, u^\mu{}_{,\mu} + c^2\, u^\mu\, \rho_{,\mu}\right)}{c^2} == 0$$

$$\frac{P_{,\mu}\, u^\mu\, u^\nu}{c^2} + \frac{P\, u^\mu\, u^\nu{}_{,\mu}}{c^2} + \rho\, u^\mu\, u^\nu{}_{,\mu} - P_{,\mu}\, \eta^{\mu\nu} == 0$$

Factoring terms and rearranging

$$\left(P + c^2\, \rho\right)\, u^\mu\, u^\nu{}_{,\mu} + P_{,\mu}\, \left(u^\mu\, u^\nu - c^2\, \eta^{\mu\nu}\right) == 0$$

$$\left(P + c^2\, \rho\right)\, u^\mu\, u^\nu{}_{,\mu} == -P_{,\mu}\, \left(u^\mu\, u^\nu - c^2\, \eta^{\mu\nu}\right)$$

Compare with equation (3.6), p.99.

Note: We can use the following relation to transform the relativistic motion equation (see p.101).

```
%[[1, {2, 3}]]
% // ExpandPartialD[labs]
% /. uu[α_] → TotalD[xu[α], τ] // TraditionalForm
% /. ReversePartialTotalChainRule
```

$$u^\mu\, u^\nu{}_{,\mu}$$

$$u^\mu\, \partial_{x^\mu}\, u^\nu$$

$$\frac{dx^\mu}{d\tau}\, \frac{\partial \frac{dx^\nu}{d\tau}}{\partial x^\mu}$$

$$\frac{d^2 x^\nu}{d\tau\, d\tau}$$

■ Classical limit of the continuity equation and the equation of motion

```
Print["Note: Series expansion of the γ(v(ξ)) factor and its derivative:"]
γ[v_] := (1 - (v / c)²)^(-1/2)
Print["γ = ", Series[γ[v[ξ]], {v[ξ], 0, 4}]]
Print["γ,ξ = ", HoldForm[0] + Series[D[γ[v[ξ]], ξ], {v[ξ], 0, 4}]]
Print["Hence γ ≈ 1 and γ,ξ ≈ 0 approximation
   means low velocity and slowly varing velocity wrt ξ."]
```

Note: Series expansion of the $\gamma(v(\xi))$ factor and its derivative:

$$\gamma = 1 + \frac{v[\xi]^2}{2 c^2} + \frac{3 v[\xi]^4}{8 c^4} + O[v[\xi]]^5$$

$$\gamma_{,\xi} = 0 + \frac{v'[\xi] v[\xi]}{c^2} + \frac{3 v'[\xi] v[\xi]^3}{2 c^4} + O[v[\xi]]^5$$

Hence $\gamma \approx 1$ and $\gamma_{,\xi} \approx 0$ approximation means low velocity and slowly varing velocity wrt ξ.

```
Print["I) Limit of the relativistic continuity equation"]
eqn[3, 5]
Print["Low pressure approximation"]
%% /. Tensor[P] → 0
Print["Substituing world velocity components"]
%% /. uu[μ_] → Tensor[γ] vu[μ]
% // UnnestTensor
Print["γ ≈ 1 and γ,μ ≈ 0 approximation"]
%% /. Tensor[γ] → 1 /. Tensor[γ, List[Void], List[Dif[μ]]] → 0
Print["Check proposed simplification"]
PartialD[NestedTensor[Tensor[ρ] vu[μ]], μ] == 0
UnnestTensor[%] == %%%
Print["Breaking into time and space components"]
(# /. μ → 0) + (# /. μ → i) & /@ %%%
Print["v⁰ is equal c"]
%% /. vu[0] → c
% // ExpandPartialD[labs] // UseCoordinates[{c t, x, y, z}];
MapAt[UnnestTensor, %, {1, 1}]
% // TraditionalForm
Print["...which is the same as the classical continuity equation."]
```

I) Limit of the relativistic continuity equation

$$\frac{P u^{\mu}{}_{,\mu}}{c^2} + (\rho u^{\mu})_{,\mu} == 0$$

Low pressure approximation

$$(\rho u^{\mu})_{,\mu} == 0$$

Substituing world velocity components

$$(\gamma \rho v^{\mu})_{,\mu} == 0$$

$$\rho v^{\mu} \gamma_{,\mu} + \gamma (\rho v^{\mu}{}_{,\mu} + v^{\mu} \rho_{,\mu}) == 0$$

$\gamma \approx 1$ and $\gamma_{,\mu} \approx 0$ approximation

$$\rho v^{\mu}{}_{,\mu} + v^{\mu} \rho_{,\mu} == 0$$

Check proposed simplification

$$(\rho v^{\mu})_{,\mu} == 0$$

True

Breaking into time and space components

$$(\rho v^0)_{,0} + (\rho v^i)_{,i} == 0$$

v^0 is equal c

$$(c \rho)_{,0} + (\rho v^i)_{,i} == 0$$

$\partial_t \rho + \partial_{x^i} (\rho\, v^i) == 0$

$$\frac{\partial \rho}{\partial t} + \frac{\partial (\rho\, v^i)}{\partial x^i} = 0$$

...which is the same as the *classical continuity equation*.

Compare with equation (3.7), p.100.

```
Print["II) Limit of the relativistic equation of motion for a perfect fluid"]
eqn[3, 6]
Print["Low pressure approximation"]
%% /. Tensor[P] → 0
Print["Substituting world velocity components"]
%% /. uu[μ_] → Tensor[γ] vu[μ] /. PartialD[uu[ν], μ] → PartialD[Tensor[γ] vu[ν], μ]
Print["γ ≈ 1 and γ,μ ≈ 0 approximation"]
%% /. Tensor[γ] → 1 /. Tensor[γ, List[Void], List[Dif[μ]]] → 0
Print["Expand, separate into time and space components"]
%% // ExpandAll
% // PartialSum[0, {j}];
% // PartialArray[0, {i}] // Thread
Print["Simplify η, ",
 vu[0] → c, ", ",
 vu[i_] vu[j_] → 0, ", ", PartialD[vu[0], j_] → 0]
%% /. ηuu[0, 0] → 1 /. ηuu[i_, j_] /; i == 0 ⋁ j == 0 → 0;
% /. vu[0] → c /. vu[i_] vu[j_] → 0 /. PartialD[vu[0], j_] → 0
Print["We can neglect the P,j term provided the rate of change of pressure in space
   is small,getting 0=0. So we take only the spatial part.\nWe can neglect
   the P,0 term provided the rate of change of pressure with time is small."]
%%[[2]] /. Tensor[P, List[Void], List[Dif[0]]] → 0
# c^-2 & /@ % // Simplify
% /. ηuu[j, i] → -δud[j, i]
% // KroneckerAbsorb[δ]
% // ExpandPartialD[labs] // UseCoordinates[{c t, x, y, z}];
% // MapLevelParts[Factor, {1, {1, 2}}]
% // TraditionalForm
Print["...which is Euler's classical equation of motion for a perfect fluid (1755)."]
```

II) Limit of the relativistic equation of motion for a perfect fluid

$\left(P + c^2 \rho\right) u^\mu\, u^\nu{}_{,\mu} == -P_{,\mu} \left(u^\mu\, u^\nu - c^2\, \eta^{\mu\nu}\right)$

Low pressure approximation

$c^2\, \rho\, u^\mu\, u^\nu{}_{,\mu} == -P_{,\mu} \left(u^\mu\, u^\nu - c^2\, \eta^{\mu\nu}\right)$

Substituting world velocity components

$c^2\, \gamma\, \rho\, v^\mu \left(\gamma\, v^\nu{}_{,\mu} + v^\nu\, \gamma_{,\mu}\right) == -P_{,\mu} \left((\gamma)^2\, v^\mu\, v^\nu - c^2\, \eta^{\mu\nu}\right)$

$\gamma \approx 1$ and $\gamma_{,\mu} \approx 0$ approximation

$c^2\, \rho\, v^\mu\, v^\nu{}_{,\mu} == -P_{,\mu} \left(v^\mu\, v^\nu - c^2\, \eta^{\mu\nu}\right)$

Expand, separate into time and space components

$c^2\, \rho\, v^\mu\, v^\nu{}_{,\mu} == -P_{,\mu}\, v^\mu\, v^\nu + c^2\, P_{,\mu}\, \eta^{\mu\nu}$

$\{c^2\, \rho\, v^0\, v^0{}_{,0} + c^2\, \rho\, v^j\, v^0{}_{,j} == -P_{,0} \left(v^0\right)^2 - P_{,j}\, v^0\, v^j + c^2\, P_{,0}\, \eta^{00} + c^2\, P_{,j}\, \eta^{j0},$
$c^2\, \rho\, v^0\, v^i{}_{,0} + c^2\, \rho\, v^j\, v^i{}_{,j} == -P_{,0}\, v^0\, v^i - P_{,j}\, v^i\, v^j + c^2\, P_{,0}\, \eta^{0i} + c^2\, P_{,j}\, \eta^{ji}\}$

Simplify η, $v^0 \to c$, $v^{i\_}\, v^{j\_} \to 0$, $v^0{}_{,j\_} \to 0$

$$\left\{0 == -c\, P_{,j}\, v^j,\; c^3\, \rho\, v^i{}_{,0} + c^2\, \rho\, v^j\, v^i{}_{,j} == -c\, P_{,0}\, v^i + c^2\, P_{,j}\, \eta^{ji}\right\}$$

We can neglect the $P_{,j}$ term provided the rate of change of
pressure in space is small,getting 0=0. So we take only the spatial part.
We can neglect the $P_{,0}$ term provided the rate of change of pressure with time is small.

$$c^3\, \rho\, v^i{}_{,0} + c^2\, \rho\, v^j\, v^i{}_{,j} == c^2\, P_{,j}\, \eta^{ji}$$

$$\rho\left(c\, v^i{}_{,0} + v^j\, v^i{}_{,j}\right) == P_{,j}\, \eta^{ji}$$

$$\rho\left(c\, v^i{}_{,0} + v^j\, v^i{}_{,j}\right) == -P_{,j}\, \delta^j{}_i$$

$$\rho\left(c\, v^i{}_{,0} + v^j\, v^i{}_{,j}\right) == -P_{,i}$$

$$\rho\left(\partial_t\, v^i + v^j\, \partial_{x^j}\, v^i\right) == -\partial_{x^i}\, P$$

$$\rho\left(\frac{\partial v^i}{\partial t} + v^j\, \frac{\partial v^i}{\partial x^j}\right) = -\frac{\partial P}{\partial x^i}$$

...which is *Euler's classical equation of motion for a perfect fluid* (1755).

Compare with equation (3.9), p.100.

■ General relativity case

```
eqn[3, 2] /. FlatToCurvedSpacetime // FrameBox // DisplayForm
eqn[3, 2, bis] /. FlatToCurvedSpacetime // FrameBox // DisplayForm
```

$$\boxed{T^{\mu\nu} == -P\, g^{\mu\nu} + \left(\frac{P}{c^2} + \rho\right) u^\mu\, u^\nu}$$

$$\boxed{T^{\mu\nu}{}_{;\mu} == \left(-P\, g^{\mu\nu} + \left(\frac{P}{c^2} + \rho\right) u^\mu\, u^\nu\right)_{;\mu} == 0}$$

Compare with equations (3.10) and (3.11), p.101.

3.2 The curvature tensor and related tensors p. 102 - 105

```
Needs["TGeneralRelativity1`GeneralRelativity`"]
$PrePrint =.
labs = {x, δ, g, Γ};
DeclareBaseIndices[{0, 1, 2, 3}]
MyRed = StyleForm[Superscript[#, "′"], FontColor → RGBColor[1, 0, 0]] &;
DeclareIndexFlavor[{red, MyRed}]
DefineTensorShortcuts[
 {{x, λ, e}, 1},
 {{δ, g, R, G, τ}, 2},
 {{Γ, R, τ}, 3},
 {{R}, 4},
 {{R}, 5}]
(* - - - *)
RiemannToChristoffelRule = LHSSymbolsToPatterns[{a, b, c, d}]@RiemannRule
ChristoffelToRiemannRule = LHSSymbolsToPatterns[{a, b, c, d, e}]@Reverse@RiemannRule
ChDoRule = LHSSymbolsToPatterns[{a, b, c}][ChristoffelDownRule]
PaMeRule = LHSSymbolsToPatterns[{a, b, c}][PartialMetricRule]
```

$R^{d\_}{}_{a\_b\_c\_} \to -\Gamma^{d}{}_{ce}\,\Gamma^{e}{}_{ab} + \Gamma^{d}{}_{be}\,\Gamma^{e}{}_{ac} - \partial_{x^c}\Gamma^{d}{}_{ab} + \partial_{x^b}\Gamma^{d}{}_{ac}$

$-\Gamma^{d\_}{}_{c\_e\_}\,\Gamma^{e\_}{}_{a\_b\_} + \Gamma^{d\_}{}_{b\_e\_}\,\Gamma^{e\_}{}_{a\_c\_} - \partial_{x^{c\_}}\Gamma^{d\_}{}_{a\_b\_} + \partial_{x^{b\_}}\Gamma^{d\_}{}_{a\_c\_} \to R^{d}{}_{abc}$

$\Gamma_{a\_b\_c\_} \to \frac{1}{2}\left(g_{ac,b} + g_{ba,c} - g_{bc,a}\right)$

$g_{a\_b\_,c\_} \to \Gamma_{abc} + \Gamma_{bac}$

1) *Curvature tensor* $R^{d}{}_{abc}$ from commutator of covariant differentiation

```
Print["Commutator of the covariant differentiation of a covariant vector field ", λd[a]]
commutator = CovariantD[λd[a], {b, c}] - CovariantD[λd[a], {c, b}]
Print["Expanding and collecting ", λd[d], " terms"]
%% // ExpandCovariantD[labs, {d, e}]
Collect[%, λd[d]]
Print["Using symmetry of Γ"]
%% // SymmetrizeSlots[Γ, 3, {1, {2, 3}}]
Print["Last terms cancel out"]
%% // MapLevelParts[SimplifyTensorSum, {{2, 4}}];
% // MapLevelParts[SimplifyTensorSum, {{2, 3}}]
Print["Substituting the definition of the Curvature tensor"]
commutator == (eqn[3, 12] = %% /. ChristoffelToRiemannRule) // FrameBox // DisplayForm
Print["Dies ist die Ricci-Identität. Ricci identity."]
```

Commutator of the covariant differentiation of a covariant vector field λ_a

$\lambda_{a;b;c} - \lambda_{a;c;b}$

Expanding and collecting λ_d terms

$-\lambda_d\,\partial_{x^c}\Gamma^{d}{}_{ba} + \lambda_d\,\partial_{x^b}\Gamma^{d}{}_{ca} + \Gamma^{e}{}_{bc}\left(-\Gamma^{d}{}_{ea}\,\lambda_d + \partial_{x^e}\lambda_a\right) - \Gamma^{e}{}_{cb}\left(-\Gamma^{d}{}_{ea}\,\lambda_d + \partial_{x^e}\lambda_a\right) +$
$\Gamma^{d}{}_{ca}\,\partial_{x^b}\lambda_d - \Gamma^{d}{}_{ba}\,\partial_{x^c}\lambda_d - \Gamma^{e}{}_{ca}\left(-\Gamma^{d}{}_{be}\,\lambda_d + \partial_{x^b}\lambda_e\right) + \Gamma^{e}{}_{ba}\left(-\Gamma^{d}{}_{ce}\,\lambda_d + \partial_{x^c}\lambda_e\right)$

$$\lambda_d \left(-\Gamma^d{}_{ce}\,\Gamma^e{}_{ba} - \Gamma^d{}_{ea}\,\Gamma^e{}_{bc} + \Gamma^d{}_{be}\,\Gamma^e{}_{ca} + \Gamma^d{}_{ea}\,\Gamma^e{}_{cb} - \partial_{x^c}\Gamma^d{}_{ba} + \partial_{x^b}\Gamma^d{}_{ca}\right) + \Gamma^e{}_{bc}\,\partial_{x^e}\lambda_a - \Gamma^e{}_{cb}\,\partial_{x^e}\lambda_a + \Gamma^d{}_{ca}\,\partial_{x^b}\lambda_d - \Gamma^d{}_{ba}\,\partial_{x^c}\lambda_d - \Gamma^e{}_{ca}\,\partial_{x^b}\lambda_e + \Gamma^e{}_{ba}\,\partial_{x^c}\lambda_e$$

Using symmetry of Γ

$$\lambda_d \left(-\Gamma^d{}_{ce}\,\Gamma^e{}_{ab} + \Gamma^d{}_{be}\,\Gamma^e{}_{ac} - \partial_{x^c}\Gamma^d{}_{ab} + \partial_{x^b}\Gamma^d{}_{ac}\right) + \Gamma^d{}_{ac}\,\partial_{x^b}\lambda_d - \Gamma^d{}_{ab}\,\partial_{x^c}\lambda_d - \Gamma^e{}_{ac}\,\partial_{x^b}\lambda_e + \Gamma^e{}_{ab}\,\partial_{x^c}\lambda_e$$

Last terms cancel out

$$\lambda_d \left(-\Gamma^d{}_{ce}\,\Gamma^e{}_{ab} + \Gamma^d{}_{be}\,\Gamma^e{}_{ac} - \partial_{x^c}\Gamma^d{}_{ab} + \partial_{x^b}\Gamma^d{}_{ac}\right)$$

Substituting the definition of the Curvature tensor

$$\boxed{\lambda_{a;b;c} - \lambda_{a;c;b} == R^d{}_{abc}\,\lambda_d}$$

Dies ist die Ricci-Identität. Ricci identity.

If the curvature tensor is identically zero, then covariant differentiation is (generally) commutative. In flat spacetime we can always find a coordinate system in which all the connection coefficients are zero, namely the Cartesian form of the Minkowski metric. Therefore the curvature tensor will be zero, not only in the Cartesian form of the metric, but in all coordinate systems. The connection coefficients are not necessarily zero in other coordinate systems. They are not tensors.

■ 2) Der *vollständig kovariante Riemannsche Krümmungstensor* (down curvature tensor) R_{abcd}

Wir wollen hier verschiedene Formen des *vollständig kovarianten Riemannschen Krümmungstensors* berechnen.

```
Print["Curvature tensor definition"]
Ruddd[f, b, c, d] == (Ruddd[f, b, c, d] /. RiemannToChristoffelRule)
Print["Lowering the index"]
Distribute[gdd[a, f] #] & /@ %%
step1 = CovariantRiemannRule = % // MetricSimplify[g]
```

Curvature tensor definition

$$R^f{}_{bcd} == \Gamma^e{}_{bd}\,\Gamma^f{}_{ce} - \Gamma^e{}_{bc}\,\Gamma^f{}_{de} - \partial_{x^d}\Gamma^f{}_{bc} + \partial_{x^c}\Gamma^f{}_{bd}$$

Lowering the index

$$g_{af}\,R^f{}_{bcd} == g_{af}\,\Gamma^e{}_{bd}\,\Gamma^f{}_{ce} - g_{af}\,\Gamma^e{}_{bc}\,\Gamma^f{}_{de} - g_{af}\,\partial_{x^d}\Gamma^f{}_{bc} + g_{af}\,\partial_{x^c}\Gamma^f{}_{bd}$$

$$R_{abcd} == \Gamma^e{}_{bd}\,\Gamma_{ace} - \Gamma^e{}_{bc}\,\Gamma_{ade} - g_{af}\,\partial_{x^d}\Gamma^f{}_{bc} + g_{af}\,\partial_{x^c}\Gamma^f{}_{bd}$$

We cannot lower an index through a partial derivative. Instead we should use the following steps to arrive at the rule for lowering the index...

```
Print["Partial derivative of metric times Christoffel and then expanded"]
stepa = PartialD[NestedTensor[gdd[a, f] Γudd[f, b, c]], d]
stepb = % // UnnestTensor
Print["Equating the two quantities and rearranging"]
stepb == stepa
# - Part[%, 1, 1] & /@ %
Print["Using the metric on lhs"]
MapAt[MetricSimplify[g], %%, {2, 2, 1}] // UnnestTensor
Print["Using the derivative of the metric in terms of Christoffel symbols"]
%% /. PaMeRule
Print["Expanding the partial derivatives and converting to a rule"]
%% // ExpandPartialD[labs]
metricpartialrule = LHSSymbolsToPatterns[{a, b, c, d, f}][Rule @@ %]
```

Partial derivative of metric times Christoffel and then expanded

$$\left(g_{af}\,\Gamma^{f}{}_{bc}\right)_{,d}$$

$$g_{af,d}\,\Gamma^{f}{}_{bc} + g_{af}\,\Gamma^{f}{}_{bc,d}$$

Equating the two quantities and rearranging

$$g_{af,d}\,\Gamma^{f}{}_{bc} + g_{af}\,\Gamma^{f}{}_{bc,d} == \left(g_{af}\,\Gamma^{f}{}_{bc}\right)_{,d}$$

$$g_{af}\,\Gamma^{f}{}_{bc,d} == -g_{af,d}\,\Gamma^{f}{}_{bc} + \left(g_{af}\,\Gamma^{f}{}_{bc}\right)_{,d}$$

Using the metric on lhs

$$g_{af}\,\Gamma^{f}{}_{bc,d} == -g_{af,d}\,\Gamma^{f}{}_{bc} + \Gamma_{abc,d}$$

Using the derivative of the metric in terms of Christoffel symbols

$$g_{af}\,\Gamma^{f}{}_{bc,d} == -\Gamma^{f}{}_{bc}\left(\Gamma_{afd} + \Gamma_{fad}\right) + \Gamma_{abc,d}$$

Expanding the partial derivatives and converting to a rule

$$g_{af}\,\partial_{x^{d}}\Gamma^{f}{}_{bc} == -\Gamma^{f}{}_{bc}\left(\Gamma_{afd} + \Gamma_{fad}\right) + \partial_{x^{d}}\Gamma_{abc}$$

$$g_{a\_f\_}\,\partial_{x^{d\_}}\Gamma^{f\_}{}_{b\_c\_} \to -\Gamma^{f}{}_{bc}\left(\Gamma_{afd} + \Gamma_{fad}\right) + \partial_{x^{d}}\Gamma_{abc}$$

We can now use this to calculate the covariant version of the curvature tensor.

```
step1
Print["Using rule above for lowering an index on a partial derivative"]
%% /. metricpartialrule
Print["Expanding, using symmetry of Γ and simplifying pairs of terms"]
%% // ExpandAll
% // SymmetrizeSlots[Γ, 3, {1, {2, 3}}];
% // MapLevelParts[SimplifyTensorSum, {2, {1, 2}}];
% // MapLevelParts[SimplifyTensorSum, {2, {1, 2}}] // IndexChange[{f, e}]
Print[
 "David Park: \"The above is the definition that I will use in calculating the Riemann
   tensor. But that form is not convenient for proving the symmetries.\""]
Print["Γudd → Γddd"]
%%% /. Γudd[e_, b_, d_] → guu[e, f] Γddd[f, b, d]
% // MapLevelParts[Factor, {2, {1, 2}}]
Print["Γ, → g"]
%% // MapLevelParts[# /. ChDoRule &, {2, {2, 3}}] // ExpandPartialD[labs]
(eqn[3, 15] = % // MapLevelParts[Factor[Expand[#]] &, {2, {2, 3}}]) // FrameBox //
 DisplayForm
Print["Vollständig kovariante Riemannsche
   Krümmungstensor. Completely covariant curvature tensor."]
RDownRule = LHSSymbolsToPatterns[{a, b, c, d}][Rule @@ eqn[3, 15]];
Print["In Komma-Notation:"]
eqn[3, 15] /. PartialD[_][Tensor[g, List[Void, Void], List[a_, b_]],
   List[Tensor[x, List[c_], List[Void]], Tensor[x, List[d_], List[Void]]]] →
  Tensor[g, List[Void, Void, Void, Void], List[a, b, Dif[c], Dif[d]]]
Print["Nur mit Metrik g:"]
%% /. ChDoRule // FullSimplify
```

$$R_{abcd} == \Gamma^{e}{}_{bd}\,\Gamma_{ace} - \Gamma^{e}{}_{bc}\,\Gamma_{ade} - g_{af}\,\partial_{x^{d}}\Gamma^{f}{}_{bc} + g_{af}\,\partial_{x^{c}}\Gamma^{f}{}_{bd}$$

Using rule above for lowering an index on a partial derivative

$$R_{abcd} == \Gamma^{e}{}_{bd}\,\Gamma_{ace} - \Gamma^{e}{}_{bc}\,\Gamma_{ade} - \Gamma^{f}{}_{bd}\,(\Gamma_{afc} + \Gamma_{fac}) + \Gamma^{f}{}_{bc}\,(\Gamma_{afd} + \Gamma_{fad}) - \partial_{x^d}\Gamma_{abc} + \partial_{x^c}\Gamma_{abd}$$

Expanding, using symmetry of Γ and simplifying pairs of terms

$$R_{abcd} == \Gamma^{e}{}_{bd}\,\Gamma_{ace} - \Gamma^{e}{}_{bc}\,\Gamma_{ade} - \Gamma^{f}{}_{bd}\,\Gamma_{afc} + \Gamma^{f}{}_{bc}\,\Gamma_{afd} - \Gamma^{f}{}_{bd}\,\Gamma_{fac} + \Gamma^{f}{}_{bc}\,\Gamma_{fad} - \partial_{x^d}\Gamma_{abc} + \partial_{x^c}\Gamma_{abd}$$

$$R_{abcd} == -\Gamma^{e}{}_{bd}\,\Gamma_{eac} + \Gamma^{e}{}_{bc}\,\Gamma_{ead} - \partial_{x^d}\Gamma_{abc} + \partial_{x^c}\Gamma_{abd}$$

David Park: "The above is the definition that I will use in calculating
the Riemann tensor. But that form is not convenient for proving the symmetries."

Γudd → Γddd

$$R_{abcd} == g^{ef}\,\Gamma_{ead}\,\Gamma_{fbc} - g^{ef}\,\Gamma_{eac}\,\Gamma_{fbd} - \partial_{x^d}\Gamma_{abc} + \partial_{x^c}\Gamma_{abd}$$

$$R_{abcd} == g^{ef}\,(\Gamma_{ead}\,\Gamma_{fbc} - \Gamma_{eac}\,\Gamma_{fbd}) - \partial_{x^d}\Gamma_{abc} + \partial_{x^c}\Gamma_{abd}$$

Γ, → g

$$R_{abcd} ==$$
$$g^{ef}\,(\Gamma_{ead}\,\Gamma_{fbc} - \Gamma_{eac}\,\Gamma_{fbd}) + \frac{1}{2}\left(-\partial_{x^b,x^d}g_{ac} - \partial_{x^c,x^d}g_{ba} + \partial_{x^a,x^d}g_{bc}\right) + \frac{1}{2}\left(\partial_{x^b,x^c}g_{ad} + \partial_{x^c,x^d}g_{ba} - \partial_{x^a,x^c}g_{bd}\right)$$

$$\boxed{R_{abcd} == g^{ef}\,(\Gamma_{ead}\,\Gamma_{fbc} - \Gamma_{eac}\,\Gamma_{fbd}) + \frac{1}{2}\left(-\partial_{x^b,x^d}g_{ac} + \partial_{x^b,x^c}g_{ad} + \partial_{x^a,x^d}g_{bc} - \partial_{x^a,x^c}g_{bd}\right)}$$

Vollständig kovariante Riemannsche Krümmungstensor. Completely covariant curvature tensor.

In Komma-Notation:

$$R_{abcd} == \frac{1}{2}\left(-g_{ac,b,d} + g_{ad,b,c} + g_{bc,a,d} - g_{bd,a,c}\right) + g^{ef}\,(\Gamma_{ead}\,\Gamma_{fbc} - \Gamma_{eac}\,\Gamma_{fbd})$$

Nur mit Metrik g:

$$R_{abcd} ==$$
$$\frac{1}{4}\Big(g^{ef}\left((-g_{ad,e} + g_{ae,d} + g_{ed,a})\,(-g_{bc,f} + g_{bf,c} + g_{fc,b}) - (-g_{ac,e} + g_{ae,c} + g_{ec,a})\,(-g_{bd,f} + g_{bf,d} + g_{fd,b})\right) +$$
$$2\,(-g_{ac,b,d} + g_{ad,b,c} + g_{bc,a,d} - g_{bd,a,c})\Big)$$

■ 3) Symmetries and proprieties of curvature tensor $R^{d}{}_{abc}$

```
Print["a) First symmetry"]
(eqn[3, 16] = Rdddd[a, b, c, d] == -Rdddd[b, a, c, d]) // FrameBox // DisplayForm
Print["RDownRule, simplify and expand"]
eqn[3, 16] /. RDownRule // Simplify
% // ExpandAll
Print["Metric Simplify (according to eqn 2.34)"]
%% // MetricSimplify[g]
Print["UpDownSwap on 2nd and 4th terms"]
%% // MapLevelParts[UpDownSwap[f], {1, {2, 4}}]
```

a) *First symmetry*

$$\boxed{R_{abcd} == -R_{bacd}}$$

RDownRule, simplify and expand

$$g^{ef}\,(\Gamma_{ebd}\,\Gamma_{fac} - \Gamma_{ebc}\,\Gamma_{fad} + \Gamma_{ead}\,\Gamma_{fbc} - \Gamma_{eac}\,\Gamma_{fbd}) == 0$$

$$g^{ef}\,\Gamma_{ebd}\,\Gamma_{fac} - g^{ef}\,\Gamma_{ebc}\,\Gamma_{fad} + g^{ef}\,\Gamma_{ead}\,\Gamma_{fbc} - g^{ef}\,\Gamma_{eac}\,\Gamma_{fbd} == 0$$

Metric Simplify (according to eqn 2.34)

$$\Gamma^{f}{}_{bd}\,\Gamma_{fac} - \Gamma^{f}{}_{bc}\,\Gamma_{fad} + \Gamma^{f}{}_{ad}\,\Gamma_{fbc} - \Gamma^{f}{}_{ac}\,\Gamma_{fbd} == 0$$

UpDownSwap on 2nd and 4th terms

True

```
Print["b) Second symmetry"]
(eqn[3, 17] = Rdddd[a, b, c, d] == -Rdddd[a, b, d, c]) // FrameBox // DisplayForm
eqn[3, 17] /. RDownRule // Simplify
```

b) *Second symmetry*

$$R_{abcd} == -R_{abdc}$$

True

```
Print["c) Third symmetry"]
(eqn[3, 18] = Rdddd[a, b, c, d] == Rdddd[c, d, a, b]) // FrameBox // DisplayForm
Print["RDownRule, use g and Γ symmetries, simplify and expand"]
eqn[3, 18] /. RDownRule // SymmetrizeSlots[g, 2, {1, {1, 2}}] //
  SymmetrizeSlots[Γ, 3, {1, {2, 3}}] // Simplify
% // ExpandAll // MetricSimplify[g]
Print["UpDownSwap on second term"]
MapAt[UpDownSwap[f], %%, {{1, 2}}]
```

c) *Third symmetry*

$$R_{abcd} == R_{cdab}$$

RDownRule, use g and Γ symmetries, simplify and expand

$$g^{ef}\left(\Gamma_{ebc}\,\Gamma_{fad} - \Gamma_{ead}\,\Gamma_{fbc}\right) == 0$$

$$\Gamma^{f}{}_{bc}\,\Gamma_{fad} - \Gamma^{f}{}_{ad}\,\Gamma_{fbc} == 0$$

UpDownSwap on second term

True

```
Print["d) Cyclic identity (Exercise 3.2.2)"]
(CyclicIdentity = Plus @@ MapThread[Ruddd[a, Sequence @@ #] &,
      {Table[RotateLeft[{b, c, d}, i], {i, 0, 2}]}] == 0) // FrameBox // DisplayForm
Print["Expanding the terms to their definitions"]
CyclicIdentity /. RiemannToChristoffelRule
Print["Using symmetry of Γ"]
%% // SymmetrizeSlots[Γ, 3, {1, {2, 3}}]
```

d) *Cyclic identity* (Exercise 3.2.2)

$$R^{a}{}_{bcd} + R^{a}{}_{cdb} + R^{a}{}_{dbc} == 0$$

Expanding the terms to their definitions

$$-\Gamma^{a}{}_{de}\,\Gamma^{e}{}_{bc} + \Gamma^{a}{}_{ce}\,\Gamma^{e}{}_{bd} + \Gamma^{a}{}_{de}\,\Gamma^{e}{}_{cb} - \Gamma^{a}{}_{be}\,\Gamma^{e}{}_{cd} - \Gamma^{a}{}_{ce}\,\Gamma^{e}{}_{db} + \Gamma^{a}{}_{be}\,\Gamma^{e}{}_{dc} - \partial_{x^d}\Gamma^{a}{}_{bc} + \partial_{x^c}\Gamma^{a}{}_{bd} + \partial_{x^d}\Gamma^{a}{}_{cb} - \partial_{x^b}\Gamma^{a}{}_{cd} - \partial_{x^c}\Gamma^{a}{}_{db} + \partial_{x^b}\Gamma^{a}{}_{dc} == 0$$

Using symmetry of Γ

True

```
Print["e) Bianchi identity",
 "\nAt any point P we can construct a geodesic coordinate system where"]
redΓrule = Γudd[a, b, c] → 0 // ToFlavor[red] // LHSSymbolsToPatterns[{a, b, c}]
Print[
 "Definition of the curvature tensor at point P in the red coordinates and reindex"]
RiemannRule
% // IndexChange[{{d, a}, {a, b}, {b, c}, {c, d}}] // ToFlavor[red]
Print["Unevaluating the partial derivatives"]
%% /. HoldPattern[PartialD[labs_][t_Tensor, Tensor[x, {a_}, {Void}]]] :→ PartialD[t, a]
Print["Taking the partial derivative of each side"]
PartialD[#, red@e] & /@ %%
Print["Setting the Christoffel symbols to zero, but not their derivatives"]
step1 = %% /. redΓrule
Print["But in the geodesic coordinates system, the
   covariant derivative is the same as the partial derivative..."]
First[step1] /. Dif → Cov
% // ExpandCovariantD[{x, δ, g, Γ}, red@f]
% /. redΓrule
% /. HoldPattern[PartialD[labs_][t_Tensor, Tensor[x, {a_}, {Void}]]] :→ PartialD[t, a]
Print["Therefore, at point P we can write a general rule:"]
MapAt[# /. Dif → Cov &, step1, 1];
RBrule = LHSSymbolsToPatterns[{a, b, c, d, e}][%]
Print["We now do a cyclic permutation of {c,d,e} on R; and add"]
Table[RotateLeft[red /@ {c, d, e}, i], {i, 0, 2}]
CovariantD[Ruddd[red@a, red@b, #1, #2], #3] & @@ # & /@ %
step2 = Plus @@ % == 0
step2 /. RBrule
% // ExpandPartialD[labs]
Print["Since the point P was arbitrary, we can use the pointwise principle and write"]
(BianchiIdentity = step2 // ToFlavor[Identity, red]) // FrameBox // DisplayForm
Print["This is the Bianchi identity."]
```

e) *Bianchi identity*
At any point P we can construct a geodesic coordinate system where

$$\Gamma^{a\_'}{}_{b\_' c\_'} \to 0$$

Definition of the curvature tensor at point P in the red coordinates and reindex

$$R^{d}{}_{abc} \to -\Gamma^{d}{}_{ce}\,\Gamma^{e}{}_{ab} + \Gamma^{d}{}_{be}\,\Gamma^{e}{}_{ac} - \partial_{x^c}\Gamma^{d}{}_{ab} + \partial_{x^b}\Gamma^{d}{}_{ac}$$

$$R^{a'}{}_{b'c'd'} \to -\Gamma^{a'}{}_{d'e'}\,\Gamma^{e'}{}_{b'c'} + \Gamma^{a'}{}_{c'e'}\,\Gamma^{e'}{}_{b'd'} - \partial_{x^{d'}}\Gamma^{a'}{}_{b'c'} + \partial_{x^{c'}}\Gamma^{a'}{}_{b'd'}$$

Unevaluating the partial derivatives

$$R^{a'}{}_{b'c'd'} \to -\Gamma^{a'}{}_{d'e'}\,\Gamma^{e'}{}_{b'c'} + \Gamma^{a'}{}_{c'e'}\,\Gamma^{e'}{}_{b'd'} - \Gamma^{a'}{}_{b'c',d'} + \Gamma^{a'}{}_{b'd',c'}$$

Taking the partial derivative of each side

$$R^{a'}{}_{b'c'd',e'} \to \Gamma^{e'}{}_{b'd'}\,\Gamma^{a'}{}_{c'e',e'} - \Gamma^{e'}{}_{b'c'}\,\Gamma^{a'}{}_{d'e',e'} - \Gamma^{a'}{}_{d'e'}\,\Gamma^{e'}{}_{b'c',e'} + \Gamma^{a'}{}_{c'e'}\,\Gamma^{e'}{}_{b'd',e'} - \Gamma^{a'}{}_{b'c',d',e'} + \Gamma^{a'}{}_{b'd',c',e'}$$

Setting the Christoffel symbols to zero, but not their derivatives

$$R^{a'}{}_{b'c'd',e'} \to -\Gamma^{a'}{}_{b'c',d',e'} + \Gamma^{a'}{}_{b'd',c',e'}$$

But in the geodesic coordinates system, the
  covariant derivative is the same as the partial derivative...

$$R^{a'}{}_{b'c'd';e'}$$

$R^{f'}{}_{b'c'd'}\,\Gamma^{a'}{}_{e'f'} - R^{a'}{}_{f'c'd'}\,\Gamma^{f'}{}_{e'b'} - R^{a'}{}_{b'f'd'}\,\Gamma^{f'}{}_{e'c'} - R^{a'}{}_{b'c'f'}\,\Gamma^{f'}{}_{e'd'} + \partial_{x^{e'}} R^{a'}{}_{b'c'd'}$

$\partial_{x^{e'}} R^{a'}{}_{b'c'd'}$

$R^{a'}{}_{b'c'd',e'}$

Therefore, at point P we can write a general rule:

$R^{a\_'}{}_{b\_'c\_'d\_';e\_'} \to -\Gamma^{a'}{}_{b'c',d',e'} + \Gamma^{a'}{}_{b'd',c',e'}$

We now do a cyclic permutation of {c,d,e} on $R_{;}$ and add

{{c′, d′, e′}, {d′, e′, c′}, {e′, c′, d′}}

$\{R^{a'}{}_{b'c'd';e'},\ R^{a'}{}_{b'd'e';c'},\ R^{a'}{}_{b'e'c';d'}\}$

$R^{a'}{}_{b'c'd';e'} + R^{a'}{}_{b'd'e';c'} + R^{a'}{}_{b'e'c';d'} == 0$

$-\Gamma^{a'}{}_{b'c',d',e'} + \Gamma^{a'}{}_{b'c',e',d'} + \Gamma^{a'}{}_{b'd',c',e'} - \Gamma^{a'}{}_{b'd',e',c'} - \Gamma^{a'}{}_{b'e',c',d'} + \Gamma^{a'}{}_{b'e',d',c'} == 0$

True

Since the point P was arbitrary, we can use the pointwise principle and write

$$\boxed{R^{a}{}_{bcd;e} + R^{a}{}_{bde;c} + R^{a}{}_{bec;d} == 0}$$

This is the Bianchi identity.

```
Print["f) Total and indipendent components of the curvature tensor Rabcd"]
Prepend[{ToString[#1] <> "D", #1^4, #1^2 (#1^2 - 1) / 12} & /@ Range[5],
  {"Manifold dimension", "tot. comp.", "ind. comp."}] // TableForm
```

f) *Total and indipendent components of the curvature tensor* R_{abcd}

Manifold dimension	tot. comp.	ind. comp.
1D	1	0
2D	16	1
3D	81	6
4D	256	20
5D	625	50

■ 4) *Ricci tensor* R_{ab}

The *Ricci tensor* is obtained by contracting the curvature tensor.

```
(eqn[3, 21] = Rdd[a, b] == Ruddd[c, a, b, c]) // FrameBox // DisplayForm
```

$$\boxed{R_{ab} == R^{c}{}_{abc}}$$

```
Print[
 "By contracting the cyclic identity we can prove that the Ricci tensor is symmetric."]
CyclicIdentity
Print["Contract a with d"]
%% /. d → a
Print["Use antisymmetry on 3rd term"]
MapAt[# /. Ruddd[a_, b_, c_, d_] → -Ruddd[a, b, d, c] &, %%, {1, 3}]
Print["Contraction of first two slotp is zero"]
%% /. Ruddd[a_, a_, c_, d_] → 0
Print["Use definition of Ricci tensor and rearrange"]
%% /. Ruddd[a_, b_, c_, a_] → Rdd[b, c]
# - Part[%, 1, 2] & /@ %
```

By contracting the cyclic identity we can prove that the Ricci tensor is symmetric.

$R^{a}{}_{bcd} + R^{a}{}_{cdb} + R^{a}{}_{dbc} == 0$

Contract a with d

$R^{a}{}_{abc} + R^{a}{}_{bca} + R^{a}{}_{cab} == 0$

Use antisymmetry on 3rd term

$R^{a}{}_{abc} + R^{a}{}_{bca} - R^{a}{}_{cba} == 0$

Contraction of first two slotp is zero

$R^{a}{}_{bca} - R^{a}{}_{cba} == 0$

Use definition of Ricci tensor and rearrange

$R_{bc} - R_{cb} == 0$

$R_{bc} == R_{cb}$

■ 5) *Curvature scalar* R

Contracting the Ricci tensor gives the *curvature scalar*.

```
(eqn[3, 22, 1] = Tensor[R] == guu[a, b] Rdd[a, b]) // FrameBox // DisplayForm
(eqn[3, 22, 2] = eqn[3, 22, 1] // MetricSimplify[g])
```

$R == g^{ab} R_{ab}$

$R == R^{b}{}_{b}$

■ 6) *Einstein tensor* G_{ab}

The *Einstein tensor* is defined by...

```
(eqn[3, 23] = Gdd[a, b] == Rdd[a, b] - 1 / 2 Tensor[R] gdd[a, b]) // FrameBox // DisplayForm
```

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

Since G is symmetric it possesses only one divergence and this divergence is zero.

```
Print["Proof that the divergence of the Einstein tensor G is zero."]
Print["Divergence of G"]
divG = CovariantD[NestedTensor[Guu[a, b]], b]
```

```
Print["Definition of G"]
%% /. Guu[a, b] → Ruu[a, b] - 1 / 2 Tensor[R] guu[a, b]
Print["Evaluate"]
%% // UnnestTensor
Print["Covariant derivative of metric is zero"]
%% /. CovariantD[guu[_, _], _] → 0
Print["Use symmetry of Ricci tensor and lower index"]
%% /. Ruud[a_, b_, c_] → Ruud[b, a, c]
Distribute[gdd[a, c] #] & /@ (% == divG)
step1 = % // MetricSimplify[g]
```

Proof that the divergence of the Einstein tensor G is zero.

Divergence of G

$G^{ab}{}_{;b}$

Definition of G

$\left(-\frac{1}{2} R\, g^{ab} + R^{ab}\right)_{;b}$

Evaluate

$\frac{1}{2}\left(-R\, g^{ab}{}_{;b} - g^{ab} R_{,b}\right) + R^{ab}{}_{;b}$

Covariant derivative of metric is zero

$-\frac{1}{2} g^{ab} R_{,b} + R^{ab}{}_{;b}$

Use symmetry of Ricci tensor and lower index

$-\frac{1}{2} g^{ab} R_{,b} + R^{ba}{}_{;b}$

$-\frac{1}{2} g^{ab} g_{ac} R_{,b} + g_{ac} R^{ba}{}_{;b} == g_{ac} G^{ab}{}_{;b}$

$-\frac{1}{2} R_{,c} + R^{b}{}_{c;b} == G_{c}{}^{b}{}_{;b}$

Using the Bianchi identity we can show that lhs is zero.

```
Print["Bianchi identity"]
BianchiIdentity
Print["Contracting a with d and introducing the Ricci tensor"]
%% /. d → a
% /. Rudddd[a_, b_, c_, a_, d_] → Rddd[b, c, d]
Print["Using antisymmetry on second term and Ricci tensor again"]
MapAt[# /. Rudddd[a_, b_, c_, d_, e_] → -Rudddd[a, b, d, c, e] &, %%, {1, 2}]
% /. Rudddd[a_, b_, c_, a_, d_] → Rddd[b, c, d]
Print["Raise the b index and contract b with e"]
Distribute[guu[b, f] #] & /@ %%
(% // MetricSimplify[g]) /. f → b
% /. e → b
Print["Introduce the curvature scalar"]
%% /. Rudd[a_, a_, Cov[b_]] → CovariantD[Tensor[R], b]
Print["Use symmetry ", Ruudd[a, b, c, d] → Ruudd[b, a, d, c],
 " and ", Ruudd[a, b, c, a] → Rud[b, c], " to simplify the last term."]
MapAt[(# /. Ruuddd[a_, b_, c_, d_, e_] → Ruuddd[b, a, d, c, e] /.
    Ruuddd[a_, b_, c_, a_, d_] → Rudd[b, c, d] &), %%, {{1, 3}}]
```

```
Print["Simplify"]
%% // IndexChange[{a, b}]
Distribute[# / 2] & /@ %
Print["But lhs is ", step1[[2]]]
%%[[1]] == HoldForm[Evaluate[step1[[1]]]]
% // ReleaseHold
Print["Hence the divergence of G is zero:"]
divG == 0 // FrameBox // DisplayForm
```

Bianchi identity

$R^{a}{}_{bcd;e} + R^{a}{}_{bde;c} + R^{a}{}_{bec;d} == 0$

Contracting a with d and introducing the Ricci tensor

$R^{a}{}_{bae;c} + R^{a}{}_{bca;e} + R^{a}{}_{bec;a} == 0$

$R_{bc;e} + R^{a}{}_{bae;c} + R^{a}{}_{bec;a} == 0$

Using antisymmetry on second term and Ricci tensor again

$R_{bc;e} - R^{a}{}_{bea;c} + R^{a}{}_{bec;a} == 0$

$R_{bc;e} - R_{be;c} + R^{a}{}_{bec;a} == 0$

Raise the b index and contract b with e

$g^{bf} R_{bc;e} - g^{bf} R_{be;c} + g^{bf} R^{a}{}_{bec;a} == 0$

$R^{b}{}_{c;e} - R^{b}{}_{e;c} + R^{ab}{}_{ec;a} == 0$

$-R^{b}{}_{b;c} + R^{b}{}_{c;b} + R^{ab}{}_{bc;a} == 0$

Introduce the curvature scalar

$-R_{,c} + R^{b}{}_{c;b} + R^{ab}{}_{bc;a} == 0$

Use symmetry $R^{ab}{}_{cd} \to R^{ba}{}_{dc}$ and $R^{ab}{}_{ca} \to R^{b}{}_{c}$ to simplify the last term.

$-R_{,c} + R^{a}{}_{c;a} + R^{b}{}_{c;b} == 0$

Simplify

$-R_{,c} + 2 R^{b}{}_{c;b} == 0$

$$-\frac{1}{2} R_{,c} + R^{b}{}_{c;b} == 0$$

But lhs is $G_{c}{}^{b}{}_{;b}$

$$-\frac{1}{2} R_{,c} + R^{b}{}_{c;b} == -\frac{1}{2} R_{,c} + R^{b}{}_{c;b}$$

True

Hence the divergence of G is zero:

$$\boxed{G^{ab}{}_{;b} == 0}$$

7) Extras

a) For covariant differentiation we can reach inside the derivative and lower the index.

```
step1 = CovariantD[NestedTensor[gdd[a, d] τud[d, b]], c]
step2 = % // UnnestTensor
step2 == step1
Print["Covariant derivative of metric tensor is zero"]
%% /. CovariantD[gdd[_, _], _] → 0
Print["We can lower indices inside a covariant derivative from outside"]
MapAt[MetricSimplify[g], %%, {2, 1}] // FrameBox // DisplayForm
```

$(g_{ad}\,\tau^{d}{}_{b})_{;c}$

$g_{ad;c}\,\tau^{d}{}_{b} + g_{ad}\,\tau^{d}{}_{b;c}$

$g_{ad;c}\,\tau^{d}{}_{b} + g_{ad}\,\tau^{d}{}_{b;c} == (g_{ad}\,\tau^{d}{}_{b})_{;c}$

Covariant derivative of metric tensor is zero

$g_{ad}\,\tau^{d}{}_{b;c} == (g_{ad}\,\tau^{d}{}_{b})_{;c}$

We can lower indices inside a covariant derivative from outside

$\boxed{g_{ad}\,\tau^{d}{}_{b;c} == \tau_{ab;c}}$

b) Covariant differentiation commutator.

```
? CovariantCommutator
```

CovariantCommutator[{c1, c2}, R, d][term] will calculate the covariant commutator, CovariantD[term,{c1,c2}] - CovariantD[term,{c2,c1}], and express the result in terms of the Riemann tensor R. d is the dummy index introduced.

```
{λd[a], λu[a], τuu[a, b], τuud[a, b, c], λu[a] λd[b]}
CovariantCommutator[{d, e}, R, f] /@ %
```

$\{\lambda_a,\ \lambda^a,\ \tau^{ab},\ \tau^{ab}{}_c,\ \lambda^a\,\lambda_b\}$

$\{R^f{}_{ade}\,\lambda_f,\ -R^a{}_{fde}\,\lambda^f,\ -R^b{}_{fde}\,\tau^{af} - R^a{}_{fde}\,\tau^{fb},\ R^f{}_{cde}\,\tau^{ab}{}_f - R^b{}_{fde}\,\tau^{af}{}_c - R^a{}_{fde}\,\tau^{fb}{}_c,\ -R^a{}_{fde}\,\lambda^f\,\lambda_b + R^f{}_{bde}\,\lambda^a\,\lambda_f\}$

c) The following routines, from the GeneralRelativity package, can be used to calculate the curvature tensor, the Ricci tensor, the curvature scalar and the Einstein tensor.

```
? CalculateRiemannd
```

CalculateRiemannd[labels, flavor:Identity, simplifyroutine:Identity] will calculate the down version of the Riemann tensor and return it as an array. labels is the list {x, δ, g, Γ}. g and Γ are the symbols for the metric tensor and Christoffel connections. They must have defined shortcuts and have been given tensor values or rules. flavor and simplifyroutine are optional arguments. flavor is the index flavor used in the g and Γ values. simplifyroutine is applied to each of the independent elements as they are calculated. Only the independent elements are separately calculated and the complete array is generated from them.

```
? CalculateRRRG
```

```
CalculateRRRG[g, riemanndown, flavor:Identity, simplifyroutine:Identity] will calculate
  Ruddd, Rdd, R, and Gdd, the up version of the Riemann tensor, the Ricci tensor,
  curvature scalar, and the Einstein tensor. They are returned as arrays in the list
  {riemann, ricci, curvaturescalar, einstein}. g is the symbol for the metric matrix.
  riemanndown is the down version of the Riemann tensor in array form. It can be
  precalculated with CalculateRiemannTensord. Tensor shortcuts must be defined for g
  and values or rules stored for the up and down metric matrix. flavor is an optional
  argument that gives the flavor of the g definitions. simplifyroutine is an optional
  argument that gives a routine to be applied to each of the elements of the arrays.
```

- d) Exercise: Is the manifold flat or curved?

```
msg = "A cylinder ξ(ϕ,z) whose cross section is a circle of radius 1,
    using {ϕ,z} borrowed from cylindrical coordinates as parameters.";
varnames = {ϕ, z};
ξ[ϕ_, z_] := {Cos[ϕ], Sin[ϕ], z}

msg = "A cone.";
varnames = {z, ϕ};
ξ[z_, ϕ_] := {z Cos[ϕ], z Sin[ϕ], z}

msg = "This parametrization, where -∞<u|v<∞, gives a hyperbolic paraboloid.";
varnames = {u, v};
ξ[u_, v_] := {u + v, u - v, 2 u v}

msg = "A sphere ξ(θ,ϕ) of radius ρ, using angles
    {θ,ϕ} borrowed from spherical coordinates as parameters.";
varnames = {θ, ϕ};
ξ[θ_, ϕ_] := ρ {Sin[θ] Cos[ϕ], Sin[θ] Sin[ϕ], Cos[θ]}

msg = "Wavy surface.";
varnames = {x, y};
ξ[x_, y_] := {x, y, Sin[x]}
```

```
DeclareBaseIndices[{1, 2}]
SetTensorValues[δud[μ, ν], IdentityMatrix[NDim]]
Print["Example: ", msg]
ξ[Sequence @@ varnames];
Print["ξ", varnames, "=", %]
SetTensorValueRules[ed[i], {∂varnames[[1]] %%, ∂varnames[[2]] %%}, True]
ed[i];
Print["natural basis ", %, ":"]
%% // ToArrayValues[]
gdd[i, j] == ed[i].ed[j];
Print["metric ", %, ":"]
%%[[2]] // ToArrayValues[] // Simplify;
% // MatrixForm
metric = %% // CoordinatesToTensors[varnames];
SetChristoffelValueRules[xu[i], metric, Γ]
Γudd[i, j, k];
Print["Christoffel symbols ", %, ":"]
%% // ToArrayValues[] // Simplify // UseCoordinates[varnames] // MatrixForm
riemannd = CalculateRiemannd[labs];
SetTensorValueRules[Rdddd[a, b, c, d], riemannd]
Rdddd[a, b, c, d];
Print["Down curvature tensor ", %, ":"]
%% // ToArrayValues[] // Simplify // UseCoordinates[varnames] // MatrixForm
If[Union[Flatten[%]] == {0}, "The manifold is flat!", "The manifold seems to be curved..."]
```

Example: Wavy surface.

ξ{x, y}={x, y, Sin[x]}

natural basis e_i:

{{1, 0, Cos[x]}, {0, 1, 0}}

metric $g_{ij} == e_i . e_j$:

$$\begin{pmatrix} 1 + \text{Cos}[x]^2 & 0 \\ 0 & 1 \end{pmatrix}$$

Christoffel symbols $\Gamma^i{}_{jk}$:

$$\begin{pmatrix} \begin{pmatrix} -\frac{\text{Sin}[2x]}{3+\text{Cos}[2x]} \\ 0 \end{pmatrix} & \begin{pmatrix} 0 \\ 0 \end{pmatrix} \\ \begin{pmatrix} 0 \\ 0 \end{pmatrix} & \begin{pmatrix} 0 \\ 0 \end{pmatrix} \end{pmatrix}$$

Down curvature tensor R_{abcd}:

$$\begin{pmatrix} \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} & \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \\ \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} & \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \end{pmatrix}$$

The manifold is flat!

Up Curvature, Ricci, Curvature Scalar and Einstein tensors given by the **CalculateRRRG[]** function:

```
SetMetricValueRules[g, metric]
MatrixForm /@ (CalculateRRRG[metric, riemannd] // Simplify // UseCoordinates[varnames])
```

$$\left\{ \begin{pmatrix} \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} & \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \\ \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} & \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \end{pmatrix}, \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}, 0, \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \right\}$$

FN: "A manifold is *flat* if at each point of it $R^a{}_{bcd}=0$, otherwise it is *curved*. (We may also speak of flat regions of a manifold.) It may be shown that in any region where $R^a{}_{bcd}=0$ it is possible to introduce [...] a Cartesian coordinate system [...]."

3.3 Curvature and parallel transport p. 105 - 110

```
Needs["TGeneralRelativity1`GeneralRelativity`"]
$PrePrint =.
labs = {x, δ, g, Γ};
DeclareBaseIndices[{0, 1, 2, 3}]
DefineTensorShortcuts[
 {{λ, λo, x, xP, ξ, Δλ, i, j, u}, 1},
 {{δ, g, f}, 2},
 {{Γ, ΓP}, 3},
 {{R}, 4}]
DeclareTensorSymmetries[Γ, 3, {1, {2, 3}}]
(* - - - *)
RiemannToChristoffelRule = LHSSymbolsToPatterns[{a, b, c, d}]@RiemannRule;
ChristoffelToRiemannRule = LHSSymbolsToPatterns[{a, b, c, d, e}]@Reverse@RiemannRule;
Format[Integral[expr_]] := DisplayForm[RowBox[{"∫", RowBox[{"(", expr, ")"}]}]]
Format[ContourIntegral[expr_]] := DisplayForm[RowBox[{"∮", RowBox[{"(", expr, ")"}]}]]
{Integral["any expression"], ContourIntegral["any expression"]}
victor[a_, b_, linecolor_: Hue[.7]] := {
   Graphics3D[{Hue[.4], PointSize[0.02], Point[a]}],
   Graphics3D[{linecolor, Line[{a, a + b}]}],
   Graphics3D[{Hue[.0], PointSize[0.0075], Point[a + b]}]};
```

$\{\int(\text{any expression}), \oint(\text{any expression})\}$

The purpose of this section is to make the connection between the curvature tensor and parallel transport clear. We shall show explicitly how the change $\Delta\lambda^a$ that results from parallelly transporting a vector λ^a around a small loop γ near a point P depends on the curvature tensor $R^a{}_{bcd}$ at P. By suitable choices for γ, this relationship can be exploited to *measure* the components of the curvature tensor at P.

1) Derivation of the equation of parallel transport deviation

We start by constructing an integral version of the parallel transport equation.

```
Print["Parallel transport equation for vector λ (eq.(2.23),p.65)"]
AbsoluteD[λu[a], t] == 0
Print["Expanding and rearranging the equation"]
MapAt[ExpandAbsoluteD[labs, {c, b}], %%, 1];
% // SymmetrizeSlots[]
(eqn[3, 24] = # - %[[1, 1]] & /@ %)
Print["Integration (path-dependent!) of the equation for λ"]
%% /. HoldPattern[TotalD[a_, b_]] :→ TotalD[a]
Integral /@ %
% /. Integral[TotalD[λu[a]]] → λu[a] - λou[a]
(eqn[3, 25] = # - %[[1, 2]] & /@ %)
```

Parallel transport equation for vector λ (eq.(2.23),p.65)

$$\frac{D\lambda^a}{dt} = 0$$

Expanding and rearranging the equation

$$\Gamma^a{}_{bc}\,\lambda^b\,\frac{dx^c}{dt} + \frac{d\lambda^a}{dt} == 0$$

$$\frac{d\lambda^a}{dt} == -\Gamma^a{}_{bc}\,\lambda^b\,\frac{dx^c}{dt}$$

Integration (path-dependent!) of the equation for λ

$$d\lambda^a == -\Gamma^a{}_{bc}\,\lambda^b\,dx^c$$

$$\int (d\lambda^a) == \int \left(-\Gamma^a{}_{bc}\,\lambda^b\,dx^c\right)$$

$$\lambda^a - \lambda o^a == \int \left(-\Gamma^a{}_{bc}\,\lambda^b\,dx^c\right)$$

$$\lambda^a == \int \left(-\Gamma^a{}_{bc}\,\lambda^b\,dx^c\right) + \lambda o^a$$

We will parallel transport λ around a small loop γ centered at xP starting from O. Let ξ be the displacement from the center xP and write...

```
xu[c] == xPu[c] + ξu[c]
TotalD[%]
% /. TotalD[xPu[_]] → 0
Print["Substituting ξ for x gives eq.(3.26),p.106:"]
(eqn[3, 26] = eqn[3, 25] /. xu[c] → ξu[c])
```

$$x^c == xP^c + \xi^c$$

$$dx^c == dxP^c + d\xi^c$$

$$dx^c == d\xi^c$$

Substituting ξ for x gives eq.(3.26),p.106:

$$\lambda^a == \int \left(-\Gamma^a{}_{bc}\,\lambda^b\,d\xi^c\right) + \lambda o^a$$

This equation is not solvable in a straightforward manner because λ appears on both sides of the equation, once in the integral. The FN method for solving the equation is to consider ξ to be small and expand the solution up to the second order in ξ. To do this, the equation is substituted in terms of itself for the λ inside the integral.

```
eqn[3, 26]
Print["First order integral equation for λ^a"]
λu[b] → λou[b]
eqn[3, 26] /. % /. Equal → TildeTilde
Print["Second order integral equation for ∮ on γ"]
λu[b] → (%%[[2]] /. {a → b, b → d, c → e})
eqn[3, 26] /. Integral → ContourIntegral /. % /. Equal → TildeTilde
Print["Expand integral and separate integral terms"]
%% // ExpandAll
% /. ContourIntegral[a_ + b_] → ContourIntegral[a] + ContourIntegral[b]
Print["Take constant factors outside the integrals"]
(eqn[3, 27] = %% //. {Integral[a_ b_] /; MatchQ[a, (-1 | λou[_])] → a Integral[b],
    ContourIntegral[a_ b_] /; MatchQ[a, (-1 | λou[_])] → a ContourIntegral[b]})
```

$$\lambda^a == \int \left(-\Gamma^a{}_{bc}\,\lambda^b\,d\xi^c\right) + \lambda o^a$$

First order integral equation for λ^a

$$\lambda^b \to \lambda o^b$$

$$\lambda^a \approx \int \left(-\Gamma^a{}_{bc}\, \lambda o^b\, d\xi^c\right) + \lambda o^a$$

Second order integral equation for $\oint$ on γ

$$\lambda^b \to \int \left(-\Gamma^b{}_{de}\, \lambda o^d\, d\xi^e\right) + \lambda o^b$$

$$\lambda^a \approx \oint \left(-\Gamma^a{}_{bc} \left(\int \left(-\Gamma^b{}_{de}\, \lambda o^d\, d\xi^e\right) + \lambda o^b\right) d\xi^c\right) + \lambda o^a$$

Expand integral and separate integral terms

$$\lambda^a \approx \oint \left(-\int \left(-\Gamma^b{}_{de}\, \lambda o^d\, d\xi^e\right) \Gamma^a{}_{bc}\, d\xi^c - \Gamma^a{}_{bc}\, \lambda o^b\, d\xi^c\right) + \lambda o^a$$

$$\lambda^a \approx \oint \left(-\int \left(-\Gamma^b{}_{de}\, \lambda o^d\, d\xi^e\right) \Gamma^a{}_{bc}\, d\xi^c\right) + \oint \left(-\Gamma^a{}_{bc}\, \lambda o^b\, d\xi^c\right) + \lambda o^a$$

Take constant factors outside the integrals

$$\lambda^a \approx \lambda o^a - \left(\oint \left(\Gamma^a{}_{bc}\, d\xi^c\right)\right) \lambda o^b + \left(\oint \left(\left(\int \left(\Gamma^b{}_{de}\, d\xi^e\right)\right) \Gamma^a{}_{bc}\, d\xi^c\right)\right) \lambda o^d$$

```
Print["a) Evaluation of the first integral with second order accuracy in ξ"]
fi = Part[eqn[3, 27], 2, 2, 2]
Print["Approximation for Γ"]
Γudd[a, b, c] == ΓPudd[a, b, c] + PartialD[ΓPudd[a, b, c], d] ξu[d];
Γsubstitution[d_] = LHSSymbolsToPatterns[{a, b, c}][Rule @@ %]
Print["Substitute approximation for Γ"]
fi /. Γsubstitution[d] // ExpandAll
Print["Split integral and remove constant terms from integral"]
%% /. ContourIntegral[a_ + b_] → ContourIntegral[a] + ContourIntegral[b]
% //. ContourIntegral[a_ b_] /; MatchQ[a, Tensor[ΓP, __]] → a ContourIntegral[b]
Print["The first integral around a loop is
   zero because ξ must return to its original value. Finally..."]
fi ≈ (integral1 = Drop[%%, 1])
Print["b) Evaluation of the second integral with second order accuracy in ξ"]
si = Part[eqn[3, 27], 2, 3, 1]
Print["Substitute Γ at point P"]
%% /. Γ → ΓP
Print["Remove constant terms from integral"]
%% //. {Integral[a_ b_] /; MatchQ[a, Tensor[ΓP, __]] → a Integral[b],
  ContourIntegral[a_ b_] /; MatchQ[a, Tensor[ΓP, __]] → a ContourIntegral[b]}
Print["Perform the inner integration, result: ", ξ^e + const, ". Finally..."]
si ≈ (integral2 = %% /. Integral[TotalD[ξu[a_]]] → ξu[a])
```

a) Evaluation of the first integral with second order accuracy in ξ

$$\oint \left(\Gamma^a{}_{bc}\, d\xi^c\right)$$

Approximation for Γ

$$\Gamma^{a\_}{}_{b\_c\_} \to \Gamma P^a{}_{bc} + \Gamma P^a{}_{bc,d}\, \xi^d$$

Substitute approximation for Γ

$$\oint \left(\Gamma P^a{}_{bc}\, d\xi^c + \Gamma P^a{}_{bc,d}\, \xi^d\, d\xi^c\right)$$

Split integral and remove constant terms from integral

$$\oint \left(\Gamma P^a{}_{bc}\, d\xi^c\right) + \oint \left(\Gamma P^a{}_{bc,d}\, \xi^d\, d\xi^c\right)$$

$$\left(\oint (d\xi^c)\right) \Gamma P^a{}_{bc} + \left(\oint (\xi^d\, d\xi^c)\right) \Gamma P^a{}_{bc,d}$$

The first integral around a loop is zero because ξ must return to its original value. Finally...

$$\oint (\Gamma^a{}_{bc}\, d\xi^c) \approx \left(\oint (\xi^d\, d\xi^c)\right) \Gamma P^a{}_{bc,d}$$

b) Evaluation of the second integral with second order accuracy in ξ

$$\oint \left(\left(\int (\Gamma^b{}_{de}\, d\xi^e)\right) \Gamma^a{}_{bc}\, d\xi^c\right)$$

Substitute Γ at point P

$$\oint \left(\left(\int (\Gamma P^b{}_{de}\, d\xi^e)\right) \Gamma P^a{}_{bc}\, d\xi^c\right)$$

Remove constant terms from integral

$$\left(\oint \left(\left(\int (d\xi^e)\right) d\xi^c\right)\right) \Gamma P^a{}_{bc}\, \Gamma P^b{}_{de}$$

Perform the inner integration, result: const + ξ^e. Finally...

$$\oint \left(\left(\int (\Gamma^b{}_{de}\, d\xi^e)\right) \Gamma^a{}_{bc}\, d\xi^c\right) \approx \left(\oint (\xi^e\, d\xi^c)\right) \Gamma P^a{}_{bc}\, \Gamma P^b{}_{de}$$

```
Print["Substituting the two integral approximations in eq.(3.27)"]
eqn[3, 27]
ReplacePart[%, {integral1, integral2}, {{2, 2, 2}, {2, 3, 1}}, {{1}, {2}}]
Print["Removing constant factors from integral and rearranging"]
# - %%[[2, 1]] & /@ %%
% /. λu[a] - λou[a] → Δλu[a]
Print["Reindexing"]
MapAt[IndexChange[{{d, c}, {c, d}}], %%, {2, 1}];
MapAt[IndexChange[{{d, b}, {b, e}, {e, c}, {c, d}}], %, {2, 2}]
Print["Factoring, we get eq.(3.28),p.107:"]
(eqn[3, 28] = MapAt[Factor, %%, 2])
```

Substituting the two integral approximations in eq.(3.27)

$$\lambda^a \approx \lambda o^a - \left(\oint (\Gamma^a{}_{bc}\, d\xi^c)\right) \lambda o^b + \left(\oint \left(\left(\int (\Gamma^b{}_{de}\, d\xi^e)\right) \Gamma^a{}_{bc}\, d\xi^c\right)\right) \lambda o^d$$

$$\lambda^a \approx \lambda o^a - \left(\oint (\xi^d\, d\xi^c)\right) \Gamma P^a{}_{bc,d}\, \lambda o^b + \left(\oint (\xi^e\, d\xi^c)\right) \Gamma P^a{}_{bc}\, \Gamma P^b{}_{de}\, \lambda o^d$$

Removing constant factors from integral and rearranging

$$\lambda^a - \lambda o^a \approx -\oint (\xi^d\, d\xi^c)\, \Gamma P^a{}_{bc,d}\, \lambda o^b + \left(\oint (\xi^e\, d\xi^c)\right) \Gamma P^a{}_{bc}\, \Gamma P^b{}_{de}\, \lambda o^d$$

$$\Delta\lambda^a \approx -\oint (\xi^d\, d\xi^c)\, \Gamma P^a{}_{bc,d}\, \lambda o^b + \left(\oint (\xi^e\, d\xi^c)\right) \Gamma P^a{}_{bc}\, \Gamma P^b{}_{de}\, \lambda o^d$$

Reindexing

$$\Delta\lambda^a \approx \left(\oint (\xi^c\, d\xi^d)\right) \Gamma P^a{}_{ed}\, \Gamma P^e{}_{bc}\, \lambda o^b - \left(\oint (\xi^c\, d\xi^d)\right) \Gamma P^a{}_{bd,c}\, \lambda o^b$$

Factoring, we get eq.(3.28),p.107:

$$\Delta\lambda^a \approx \left(\oint (\xi^c\, d\xi^d)\right) \left(\Gamma P^a{}_{ed}\, \Gamma P^e{}_{bc} - \Gamma P^a{}_{bd,c}\right) \lambda o^b$$

We can manipulate the contour integral $\oint$ to a form that shows it is an antisymmetric expression.

```
ξu[c] ξu[d];
TotalD[NestedTensor[%]] == TotalD[%]
Print["The following contour integral is zero"]
ContourIntegral /@ %% == 0 // Flatten
Print["Manipulate the equation and convert to a rule"]
Drop[%%, 1];
Distribute /@ %;
# - Part[%, 1, 1] & /@ %
cirule = LHSSymbolsToPatterns[{c, d}][Rule @@ %]
Print["Split integral and use antisymmetry"]
temp = ContourIntegral[ξu[c] TotalD[ξu[d]]];
temp == 1 / 2 HoldForm[a + a] /. a → temp
MapAt[(# /. cirule) &, %, {2, 2, 1, 2}] // ReleaseHold
(eqn[3, 29] = % /. ContourIntegral[a_] - ContourIntegral[b_] → ContourIntegral[a - b]) ==
 fuu[c, d]
dfrule = LHSSymbolsToPatterns[{c, d}][2 %[[1, 2]] → 2 %[[2]]]
```

$$đ(\xi^c \xi^d) == \xi^d đ\xi^c + \xi^c đ\xi^d$$

The following contour integral is zero

$$\oint(đ(\xi^c \xi^d)) == \oint(\xi^d đ\xi^c + \xi^c đ\xi^d) == 0$$

Manipulate the equation and convert to a rule

$$\oint(\xi^c đ\xi^d) == -\oint(\xi^d đ\xi^c)$$

$$\oint(\xi^{c\_} đ\xi^{d\_}) \to -\oint(\xi^d đ\xi^c)$$

Split integral and use antisymmetry

$$\oint(\xi^c đ\xi^d) == \frac{1}{2}\left(\oint(\xi^c đ\xi^d) + \oint(\xi^c đ\xi^d)\right)$$

$$\oint(\xi^c đ\xi^d) == \frac{1}{2}\left(-\oint(\xi^d đ\xi^c) + \oint(\xi^c đ\xi^d)\right)$$

$$\left(\oint(\xi^c đ\xi^d) == \frac{1}{2}\oint(-\xi^d đ\xi^c + \xi^c đ\xi^d)\right) == f^{cd}$$

$$\oint(-\xi^{d\_} đ\xi^{c\_} + \xi^{c\_} đ\xi^{d\_}) \to 2 f^{cd}$$

The integral is an antisymmetric tensor. Substituting this into eq.(3.28). In Exercise 1.8.3 we showed that contracting a symmetric tensor with an antisymmetric tensor gives zero. Therefore it is only the antisymmetric portion of the expression in brackets (involving ΓP) that contributes to the result.

```
(eqn[3, 29, 1] = eqn[3, 28] /. Rule @@ eqn[3, 29])
Print["Replacing second factor with its
   antisymmetric part and changing ΓP to Γ for convenience"]
MapAt[Factor@ Expand@ (AntiSymmetric[{c, d}] /@#) &, %%, {2, 3}] /. ΓP → Γ
Print["Expanding the partial derivatives"]
(eqn[3, 29, 2] = %% // ExpandPartialD[labs])
Print["SymmetrizeSlots on Γ"]
%% // SymmetrizeSlots[Γ, 3, {1, {2, 3}}]
Print["Substitute the curvature tensor in P"]
%% /. ChristoffelToRiemannRule /. R → RP
Print["SymmetrizeSlots on last two indices of RP and substituiting
```

```
  f^cd for the contour integral ∮. We get finally eq.(3.30),p.107:"]
(eqn[3, 30] = %% // SymmetrizeSlots[RP, 4, {-1, {3, 4}}])
% /. dfrule // FrameBox // DisplayForm
```

$$\Delta\lambda^a \approx \frac{1}{2}\left(\oint\left(-\xi^d\, d\xi^c + \xi^c\, d\xi^d\right)\right)\left(\Gamma P^a{}_{ed}\,\Gamma P^e{}_{bc} - \Gamma P^a{}_{bd,c}\right)\lambda o^b$$

Replacing second factor with its antisymmetric part and changing ΓP to Γ for convenience

$$\Delta\lambda^a \approx \frac{1}{4}\left(\oint\left(-\xi^d\, d\xi^c + \xi^c\, d\xi^d\right)\right)\left(\Gamma^a{}_{ed}\,\Gamma^e{}_{bc} - \Gamma^a{}_{ec}\,\Gamma^e{}_{bd} + \Gamma^a{}_{bc,d} - \Gamma^a{}_{bd,c}\right)\lambda o^b$$

Expanding the partial derivatives

$$\Delta\lambda^a \approx \frac{1}{4}\left(\oint\left(-\xi^d\, d\xi^c + \xi^c\, d\xi^d\right)\right)\lambda o^b\left(\Gamma^a{}_{ed}\,\Gamma^e{}_{bc} - \Gamma^a{}_{ec}\,\Gamma^e{}_{bd} + \partial_{x^d}\Gamma^a{}_{bc} - \partial_{x^c}\Gamma^a{}_{bd}\right)$$

SymmetrizeSlots on Γ

$$\Delta\lambda^a \approx \frac{1}{4}\left(\oint\left(-\xi^d\, d\xi^c + \xi^c\, d\xi^d\right)\right)\lambda o^b\left(\Gamma^a{}_{de}\,\Gamma^e{}_{bc} - \Gamma^a{}_{ce}\,\Gamma^e{}_{bd} + \partial_{x^d}\Gamma^a{}_{bc} - \partial_{x^c}\Gamma^a{}_{bd}\right)$$

Substitute the curvature tensor in P

$$\Delta\lambda^a \approx \frac{1}{4}\left(\oint\left(-\xi^d\, d\xi^c + \xi^c\, d\xi^d\right)\right) RP^a{}_{bdc}\,\lambda o^b$$

SymmetrizeSlots on last two indices of RP and
  substituiting f^{cd} for the contour integral ∮. We get finally eq.(3.30),p.107:

$$\Delta\lambda^a \approx -\frac{1}{4}\left(\oint\left(-\xi^d\, d\xi^c + \xi^c\, d\xi^d\right)\right) RP^a{}_{bcd}\,\lambda o^b$$

$$\boxed{\Delta\lambda^a \approx -\frac{1}{2} f^{cd}\, RP^a{}_{bcd}\,\lambda o^b}$$

This is the *equation of parallel transport deviation*.

▪ 2) Investigate the components of the curvature tensor

We take γ to be a small loop lying in a surface Σ embedded in the manifold and surrounding P: $(x^a)_G = (x^a)_P + x\, i^a + y\, j^a$, where $\{i^a, j^a\}$ are an orthogonal pair of unit vectors at P and x, y defining the point G ∈ γ are small. The pair (x, y) act as locally Cartesian coordinates on Σ, with P as origin and "axes" $\{i^a, j^a\}$.

```
Print["f^cd in eq.(3.30) is defined as"]
eqn[3, 29][[2]]
Print["Substitute ξ in terms of x and y"]
ξu[a_] → x iu[a] + y ju[a]
%%% /. % // TraditionalForm
Print["Set the total derivatives of the (constant) unit vectors to zero"]
%% /. {TotalD[iu[_]] → 0, TotalD[ju[_]] → 0}
Print["Simplify the integrand expression"]
MapAt[Minus /@ (# // Expand // Factor) &, %%, {2, 1}]
Print["Remove the constant factor from the contour integral"]
%% /. ContourIntegral[a_? (FreeQ[#, HoldPattern[TotalD[_]]] &) b_] → a ContourIntegral[b]
Print[
 "1/2 the contour integral is very nearly the area of the (small) loop γ, so we get"]
fexpr = ReplacePart[%%, 2 area_γ, 2]
```

f^{cd} in eq.(3.30) is defined as

$$\frac{1}{2}\oint\left(-\xi^{d}\,d\xi^{c}+\xi^{c}\,d\xi^{d}\right)$$

Substitute ξ in terms of x and y

$$\xi^{a\_}\to x\,i^{a}+y\,j^{a}$$

$$\frac{1}{2}\oint\left((x\,i^{c}+y\,j^{c})\left(dx\,i^{d}+dy\,j^{d}+x\,d\,i^{d}+y\,d\,j^{d}\right)-\left(x\,i^{d}+y\,j^{d}\right)(dx\,i^{c}+dy\,j^{c}+x\,d\,i^{c}+y\,d\,j^{c})\right)$$

Set the total derivatives of the (constant) unit vectors to zero

$$\frac{1}{2}\oint\left(-(\mathrm{Dt}[x]\,i^{c}+\mathrm{Dt}[y]\,j^{c})\left(x\,i^{d}+y\,j^{d}\right)+(x\,i^{c}+y\,j^{c})\left(\mathrm{Dt}[x]\,i^{d}+\mathrm{Dt}[y]\,j^{d}\right)\right)$$

Simplify the integrand expression

$$\frac{1}{2}\oint\left((-y\,\mathrm{Dt}[x]+x\,\mathrm{Dt}[y])\left(-i^{d}\,j^{c}+i^{c}\,j^{d}\right)\right)$$

Remove the constant factor from the contour integral

$$\frac{1}{2}\left(\oint(-y\,\mathrm{Dt}[x]+x\,\mathrm{Dt}[y])\right)\left(-i^{d}\,j^{c}+i^{c}\,j^{d}\right)$$

1/2 the contour integral is very nearly the area of the (small) loop γ, so we get

$$\mathrm{area}_{\gamma}\left(-i^{d}\,j^{c}+i^{c}\,j^{d}\right)$$

We can now substitute this approximation for f^{cd} into our equation of parallel transport deviation given by eq. (3.30).

```
eqn[3, 30] /. RP → R
Print["Replacing contour integral and rearranging"]
ReplacePart[%%, 2 fexpr, {2, 2}]
# / area_γ & /@ %
Print["Expanding, reindexing and using an antisymmetry
   of R leads finally to the approximation eq.(3.32),p.108"]
%% // ExpandAll
(eqn[3, 32] = MapAt[IndexChange[{{c, d}, {d, c}}], %, {2, 1}] //
     SymmetrizeSlots[R, 4, {-1, {3, 4}}]) // FrameBox // DisplayForm
Print["This is equal to"]
eqn[3, 32] // IndexChange[{{c, d}, {d, c}}] // SymmetrizeSlots[R, 4, {-1, {3, 4}}]
```

$$\Delta\lambda^{a}\approx-\frac{1}{4}\left(\oint\left(-\xi^{d}\,d\xi^{c}+\xi^{c}\,d\xi^{d}\right)\right)R^{a}{}_{bcd}\,\lambda o^{b}$$

Replacing contour integral and rearranging

$$\Delta\lambda^{a}\approx-\frac{1}{2}\,\mathrm{area}_{\gamma}\left(-i^{d}\,j^{c}+i^{c}\,j^{d}\right)R^{a}{}_{bcd}\,\lambda o^{b}$$

$$\frac{\Delta\lambda^{a}}{\mathrm{area}_{\gamma}}\approx-\frac{1}{2}\left(-i^{d}\,j^{c}+i^{c}\,j^{d}\right)R^{a}{}_{bcd}\,\lambda o^{b}$$

Expanding, reindexing and using an
  antisymmetry of R leads finally to the approximation eq.(3.32),p.108

$$\frac{\Delta\lambda^{a}}{\mathrm{area}_{\gamma}}\approx\frac{1}{2}\,i^{d}\,j^{c}\,R^{a}{}_{bcd}\,\lambda o^{b}-\frac{1}{2}\,i^{c}\,j^{d}\,R^{a}{}_{bcd}\,\lambda o^{b}$$

$$\boxed{\frac{\Delta\lambda^{a}}{\mathrm{area}_{\gamma}}\approx-i^{c}\,j^{d}\,R^{a}{}_{bcd}\,\lambda o^{b}}$$

```
This is equal to
```

$$\frac{\triangle\lambda^{a}}{\text{area}_\gamma} \approx i^{d}\, j^{c}\, R^{a}{}_{bcd}\, \lambda o^{b}$$

Spezialfall 2D-Mannigfaltigkeit (special case 2D-manifold):

```
oldindices = BaseIndices;
DeclareBaseIndices[{1, 2}]
eqn[3, 32]
% /. TildeTilde → Equal // EinsteinSum[] // EinsteinArray[];
(% // SymmetrizeSlots[R, 4, {-1, {3, 4}}] // Simplify);
% /. Ruddd[a_, a_, _, _] → 0
(Solve[%, {Ruddd[1, 2, 1, 2], Ruddd[2, 1, 1, 2]}] // Flatten) /. Rule → TildeTilde //
   TableForm // FrameBox // DisplayForm
DeclareBaseIndices[oldindices]
```

$$\frac{\triangle\lambda^{a}}{\text{area}_\gamma} \approx -i^{c}\, j^{d}\, R^{a}{}_{bcd}\, \lambda o^{b}$$

$$\left\{\frac{\triangle\lambda^{1}}{\text{area}_\gamma} == \left(i^2 j^1 - i^1 j^2\right) R^{1}{}_{212}\, \lambda o^{2},\ \frac{\triangle\lambda^{2}}{\text{area}_\gamma} == \left(i^2 j^1 - i^1 j^2\right) R^{2}{}_{112}\, \lambda o^{1}\right\}$$

$$R^{1}{}_{212} \approx \frac{\triangle\lambda^{1}}{\text{area}_\gamma \left(i^2 j^1 - i^1 j^2\right) \lambda o^{2}}$$

$$R^{2}{}_{112} \approx \frac{\triangle\lambda^{2}}{\text{area}_\gamma \left(i^2 j^1 - i^1 j^2\right) \lambda o^{1}}$$

3) Ants on a surface measuring the curvature tensor

Ants living on a certain surface decide to explore the geometrical proprieties of their world and use the equation of parallel transport deviation to measure the non-null components of the curvature tensor $R^{a}{}_{bcd}$ at some point P. They measure (maybe using Lanchester's transporter, see FN p.233) the change $\triangle\lambda^{a}$ that results from parallelly transporting a vector λ^{a} around a small loop γ centered at P and the area area_γ enclosed by the loop and apply then eq.(3.32) for a 2D-manifold.

A) Definition of the experimental settings.

```
msg = "sphere";
pϕ[ϕ_] := 0; pθ[θ_] := 0;

msg = "shell potatoid";
pϕ[ϕ_] := (-Pi - ϕ)^2; pθ[θ_] := 1;

msg = "X potatoid";
pϕ[ϕ_] := Sin[ϕ]^2; pθ[θ_] := Sin[2 θ]^2;

Print["Example: ", msg]
Print["Surface {X[ϕ,θ],Y[ϕ,θ],Z[ϕ,θ]}:"]
Rr[ϕ_, θ_] = 1 + pϕ[ϕ] pθ[θ];
surface[ϕ_, θ_] =
  {Rr[ϕ, θ] Cos[ϕ] Sin[θ],
   Rr[ϕ, θ] Sin[ϕ] Sin[θ],
   Rr[ϕ, θ] Cos[θ]};
rϕ = {ϕ, -Pi, Pi}; rθ = {θ, 0, Pi};
surface[ϕ, θ] // Simplify // MatrixForm
naturalBasis = {nb1[ϕ_, θ_], nb2[ϕ_, θ_]} = {D[surface[ϕ, θ], ϕ], D[surface[ϕ, θ], θ]};
gd[ϕ_, θ_] = Outer[Dot, naturalBasis, naturalBasis, 1];
(* - - - *)
```

```
Print["Chosen point P:"]
ϕp = 1.5; θp = 1.0;
p = surface[ϕp, θp]
Print["Chosen orthonormal vector pair {i,j} at P (natural basis and 3D-basis):"]
Clear[iu1, iu2, ju1, ju2]
Solve[{iu1 == 0.5,
   {iu1, iu2}.gd[ϕp, θp].{iu1, iu2} == 1,
   {ju1, ju2}.gd[ϕp, θp].{ju1, ju2} == 1,
   {iu1, iu2}.gd[ϕp, θp].{ju1, ju2} == 0}];
%[[1]] /. Rule → Set;
ColumnForm /@ {{iu1, iu2}, {ju1, ju2}}
{i, j} = {iu1 nb1[ϕp, θp] +
     iu2 nb2[ϕp, θp], ju1 nb1[ϕp, θp] +
     ju2 nb2[ϕp, θp]};
ColumnForm /@ %
Print["Check orthonormality: i.i,j.j and i.j"]
{i.i, j.j, i.j} // Chop
Print["Loop γ:"]
ϵ = 0.1;
xx[t_] := ϵ Cos[t]; yy[t_] := ϵ Sin[t];
xG1[t_] := ϕp + xx[t] iu1 + yy[t] ju1
xG2[t_] := θp + xx[t] iu2 + yy[t] ju2
"{ϕ(t),θ(t)}" == {xG1[t], xG2[t]}
Print["Parallel transported vector λ (natural basis):"]
{λo1, λo2} = {1, 1}
```

Example: X potatoid

Surface {X[ϕ,θ],Y[ϕ,θ],Z[ϕ,θ]}:

$$\begin{pmatrix} \text{Cos}[\phi]\,\text{Sin}[\theta]\,\left(1+\text{Sin}[2\,\theta]^2\,\text{Sin}[\phi]^2\right) \\ \text{Sin}[\theta]\,\text{Sin}[\phi]\,\left(1+\text{Sin}[2\,\theta]^2\,\text{Sin}[\phi]^2\right) \\ \text{Cos}[\theta]\,\left(1+\text{Sin}[2\,\theta]^2\,\text{Sin}[\phi]^2\right) \end{pmatrix}$$

Chosen point P:

{0.108492, 1.52989, 0.984801}

Chosen orthonormal vector pair {i,j} at P (natural basis and 3D-basis):

$$\left\{\begin{matrix}0.5\\-0.255063\end{matrix},\ \begin{matrix}-0.416714\\-0.337998\end{matrix}\right\}$$

$$\left\{\begin{matrix}-0.756378\\0.175085\\0.630268\end{matrix},\ \begin{matrix}0.641387\\0.00921659\\0.767162\end{matrix}\right\}$$

Check orthonormality: i.i,j.j and i.j

{1., 1., 0}

Loop γ:

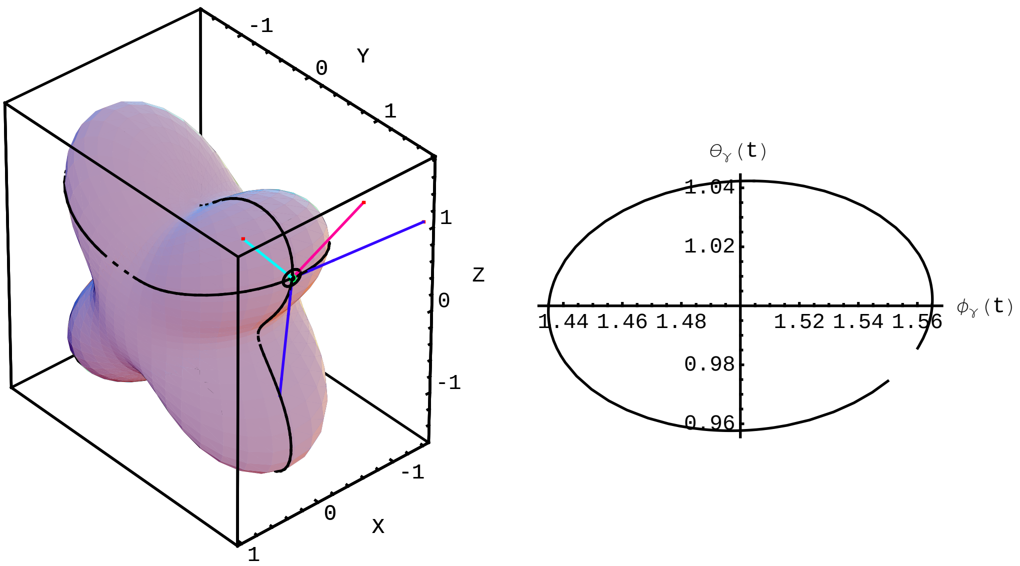
{ϕ(t),θ(t)} == {1.5 + 0.05 Cos[t] - 0.0416714 Sin[t], 1. - 0.0255063 Cos[t] - 0.0337998 Sin[t]}

Parallel transported vector λ (natural basis):

{1, 1}

```
pp = 40;
Potatoid = ParametricPlot3D[surface[ϕ, θ], Evaluate[rϕ],
   Evaluate[rθ], PlotPoints → {pp, pp}, DisplayFunction → Identity];
CLϕp = ParametricPlot3D[surface[ϕp, θ], Evaluate[rθ], DisplayFunction → Identity];
```

```
CLθp = ParametricPlot3D[surface[ϕ, θp], Evaluate[rϕ], DisplayFunction → Identity];
nb1p = victor[p, nb1[ϕp, θp]];
nb2p = victor[p, nb2[ϕp, θp]];
ip = victor[p, i, Hue[.9]];
jp = victor[p, j, Hue[.5]];
γp = ParametricPlot3D[surface[xG1[t], xG2[t]], {t, 0, 2 π}, DisplayFunction → Identity];
g1 = Show[Graphics3D[EdgeForm[], Axes → True,
     AxesLabel → {"X", "Y", "Z"}, ViewPoint → p + 10 {1, 1, 1}],
    Potatoid,
    Graphics3D[{PointSize[0.015], Point[p]}],
    CLϕp, CLθp, nb1p, nb2p, ip, jp, γp, DisplayFunction → Identity
   ];
g2 = ParametricPlot[{xG1[t], xG2[t]}, {t, 0, 2 π - .3}, AspectRatio → Automatic,
    AxesLabel → {"ϕγ(t)", "θγ(t)"}, DisplayFunction → Identity];
Show[GraphicsArray[{g1, g2}], ImageSize → 72 × 7];
```



★ Plot of ant's world with coordinate lines and natural basis at point P, orthonormal vector pair {i , j} and loop γ. ★

B) Theoretically expected results.

```
DeclareBaseIndices[{1, 2}]
metric = gd[ϕ, θ] // CoordinatesToTensors[{ϕ, θ}, u];
SetMetricValueRules[g, metric]; SetChristoffelValueRules[uu[k], metric, Γ];
Print["Vector parallel transport equation on loop γ"]
MapAt[ExpandAbsoluteD[labs, {c, b}], AbsoluteD[λu[a], t] == 0, 1] // SymmetrizeSlots[]
% // EinsteinSum[] // ArrayExpansion[a] // ToArrayValues[];
% // UseCoordinates[{xG1[t], xG2[t]}, u];
% // UseCoordinates[{λ1[t], λ2[t]}, λ];
% // UseCoordinates[{xG1[t], xG2[t]}, x];
VectorParallelTransportEquationOnγ = %;
Short[%, 5]
Ruddd[a, b, c, d]
% /. RiemannToChristoffelRule
{Ruddd[1, 2, 1, 2], Ruddd[2, 1, 1, 2]}
% /. RiemannToChristoffelRule
Rtheor = % // EinsteinSum[] // ToArrayValues[] // UseCoordinates[{ϕ, θ}, u] //
    UseCoordinates[{ϕ, θ}, x];
Rtheorp = {"Theory", Rtheor /. {ϕ → ϕp, θ → θp}}
```

```
Vector parallel transport equation on loop γ
```

$$\Gamma^{a}{}_{bc}\,\lambda^{b}\,\frac{dx^{c}}{dt}+\frac{d\lambda^{a}}{dt}==0$$

$$\Big\{\frac{1}{2}\,(-0.0337998\,\mathrm{Cos}[t]+0.0255063\,\mathrm{Sin}[t])\left(\frac{524\,288\,\ll5\gg\,(\ll1\gg)}{(\ll1\gg)^{2}\,(\ll1\gg)}+(64\,(-306+\ll16\gg+4\,\ll1\gg)\,\ll1\gg\,(\ll4\gg+\ll1\gg))\Big/\left((\ll1\gg)^{2}\,(\ll1\gg)\right)\right)\lambda1[t]+\ll3\gg+\lambda1'[t]==0,\ \ll1\gg==0\Big\}$$

$$R^{a}{}_{bcd}$$

$$-\Gamma^{a}{}_{de}\,\Gamma^{e}{}_{bc}+\Gamma^{a}{}_{ce}\,\Gamma^{e}{}_{bd}-\partial_{x^{d}}\Gamma^{a}{}_{bc}+\partial_{x^{c}}\Gamma^{a}{}_{bd}$$

$$\{R^{1}{}_{212},\ R^{2}{}_{112}\}$$

$$\{-\Gamma^{1}{}_{2e}\,\Gamma^{e}{}_{21}+\Gamma^{1}{}_{1e}\,\Gamma^{e}{}_{22}-\partial_{x^{2}}\Gamma^{1}{}_{21}+\partial_{x^{1}}\Gamma^{1}{}_{22},\ -\Gamma^{2}{}_{2e}\,\Gamma^{e}{}_{11}+\Gamma^{2}{}_{1e}\,\Gamma^{e}{}_{12}-\partial_{x^{2}}\Gamma^{2}{}_{11}+\partial_{x^{1}}\Gamma^{2}{}_{12}\}$$

{Theory, {8.65686, -3.6638}}

C) Ant's experimental results and comparison with theory.

```
Print["Measured areaγ"]
areaγ = 1 / 2 NIntegrate[xx[t] yy'[t] - yy[t] xx'[t], {t, 0, 2 π}]
Print["Measured deviation {Δλ1 ,Δλ2 }"]
NDSolve[Union[VectorParallelTransportEquationOnγ, {λ1[0] == λo1, λ2[0] == λo2}],
  {λ1, λ2}, {t, 0, 2 π}];
{λ1[2 π], λ2[2 π]} /. %;
{Δλ1 , Δλ2 } = (% // First) - {λo1, λo2}
{Ruddd[1, 2, 1, 2], Ruddd[2, 1, 1, 2]}
{"Experiment",
 {R1212 = Δλ1 / (areaγ λo2 ( iu2 ju1 - iu1 ju2)), R2112 = Δλ2 / (areaγ λo1 ( iu2 ju1 - iu1 ju2))}}
Rtheorp
{"Relativ error", {R1212 / %[[2, 1]] - 1, R2112 / %[[2, 2]] - 1} * 100 "%"}
```

Measured $area_{\gamma}$

0.0314159

Measured deviation {Δλ1 ,Δλ2 }

{0.0713319, -0.0330282}

$$\{R^{1}{}_{212},\ R^{2}{}_{112}\}$$

{Experiment, {8.24799, -3.81899}}

{Theory, {8.65686, -3.6638}}

{Relativ error, {-4.72305 %, 4.23566 %}}

"Ants are small and silent. And gigantically mysterious..."

3.4 Geodesic deviation p. 110 - 112

```
Needs["TGeneralRelativity1`GeneralRelativity`"]
$PrePrint =.
labs = {x, δ, g, Γ};
DeclareBaseIndices[{0, 1, 2, 3}]
DefineTensorShortcuts[
 {{x, xp, v, vp, ξ, zero}, 1},
 {{δ, g}, 2},
 {{Γ, Γp}, 3},
 {{R}, 4}]
DeclareZeroTensor[zero]
DeclareTensorSymmetries[Γ, 3, {1, {2, 3}}]
(* - - - *)
ChristoffelToRiemannRule = LHSSymbolsToPatterns[{a, b, c, d, e}]@Reverse@RiemannRule;
```

We derive here the *equation of geodesic deviation*.

```
Print["Let there be two affinely parametrized nearby geodesics."]
AbsoluteD[vpu[a], u] == 0
MapAt[ExpandAbsoluteD[{xp, δ, g, Γp}, {b, c}], %, 1]
(eqn[3, 33] = % /. vpu[a_] → TotalD[xpu[a], u]) // FrameBox // DisplayForm
AbsoluteD[vu[a], u] == 0
MapAt[ExpandAbsoluteD[{x, δ, g, Γ}, {b, c}], %, 1]
(eqn[3, 34] = % /. vu[a_] → TotalD[xu[a], u]) // FrameBox // DisplayForm
Print["(We are going to need the last equation in the form of a rule...)"]
geodesicrule[b_, c_] =
 LHSSymbolsToPatterns[{a}][Rule @@ (# - Part[eqn[3, 34], 1, 2] & /@ eqn[3, 34])]
```

```
Let there be two affinely parametrized nearby geodesics.
```

$$\frac{D vp^a}{du} == 0$$

$$\frac{d vp^a}{du} + vp^c\, \Gamma p^a{}_{bc} \frac{d xp^b}{du} == 0$$

$$\boxed{\frac{d^2 xp^a}{du\, du} + \Gamma p^a{}_{bc} \frac{d xp^b}{du} \frac{d xp^c}{du} == 0}$$

$$\frac{D v^a}{du} == 0$$

$$\frac{d v^a}{du} + v^c\, \Gamma^a{}_{bc} \frac{d x^b}{du} == 0$$

$$\boxed{\frac{d^2 x^a}{du\, du} + \Gamma^a{}_{bc} \frac{d x^b}{du} \frac{d x^c}{du} == 0}$$

```
(We are going to need the last equation in the form of a rule...)
```

$$\frac{d^2 x^{a\_}}{du\, du} \to -\Gamma^a{}_{bc} \frac{d x^b}{du} \frac{d x^c}{du}$$

```
Print["Subtracting the two equations"]
Inner[Subtract, eqn[3, 33], eqn[3, 34], Equal]
Print["We let ξ(u)=xp(u)-x(u) be a small vector connecting points on the two curves
   with the same parameter value u. Substitute xp in terms of x and ξ."]
%% /. xpu[a_] → xu[a] + ξu[a]
Print["We have to first order:"]
Γpudd[a_, b_, c_] → Γudd[a, b, c] + PartialD[Γudd[a, b, c], d] ξu[d]
%%% /. %
AA = % // ExpandAll
Print["Pick out only first order terms in ξ and expand partial derivatives"]
MapAt[Select[#, Count[{#}, ξu[_], ∞] < 2 &] &, AA, 1]
(step1 = MapAt[ExpandPartialD[labs], %, {1, 1, 1}]) // FrameBox // DisplayForm
Print["(The neglected second or higher order terms in ξ are:)"]
Select[AA[[1]], Count[{#}, ξu[_], ∞] >= 2 &]
```

Subtracting the two equations

$$-\frac{d^2x^a}{dudu} - \Gamma^a{}_{bc}\frac{dx^b}{du}\frac{dx^c}{du} + \frac{d^2xp^a}{dudu} + \Gamma p^a{}_{bc}\frac{dxp^b}{du}\frac{dxp^c}{du} == 0$$

We let $\xi(u)=xp(u)-x(u)$ be a *small* vector connecting points on the two curves with the same parameter value u. Substitute xp in terms of x and ξ.

$$-\Gamma^a{}_{bc}\frac{dx^b}{du}\frac{dx^c}{du} + \frac{d^2\xi^a}{dudu} + \Gamma p^a{}_{bc}\left(\frac{dx^b}{du}+\frac{d\xi^b}{du}\right)\left(\frac{dx^c}{du}+\frac{d\xi^c}{du}\right) == 0$$

We have to first order:

$$\Gamma p^{a\_}{}_{b\_c\_} \to \Gamma^a{}_{bc} + \Gamma^a{}_{bc,d}\,\xi^d$$

$$-\Gamma^a{}_{bc}\frac{dx^b}{du}\frac{dx^c}{du} + \frac{d^2\xi^a}{dudu} + \left(\Gamma^a{}_{bc}+\Gamma^a{}_{bc,d}\,\xi^d\right)\left(\frac{dx^b}{du}+\frac{d\xi^b}{du}\right)\left(\frac{dx^c}{du}+\frac{d\xi^c}{du}\right) == 0$$

$$\Gamma^a{}_{bc,d}\,\xi^d\frac{dx^b}{du}\frac{dx^c}{du} + \frac{d^2\xi^a}{dudu} + \Gamma^a{}_{bc}\frac{dx^c}{du}\frac{d\xi^b}{du} + \Gamma^a{}_{bc,d}\,\xi^d\frac{dx^c}{du}\frac{d\xi^b}{du} + \Gamma^a{}_{bc}\frac{dx^b}{du}\frac{d\xi^c}{du} + \Gamma^a{}_{bc,d}\,\xi^d\frac{dx^b}{du}\frac{d\xi^c}{du} + \Gamma^a{}_{bc}\frac{d\xi^b}{du}\frac{d\xi^c}{du} + \Gamma^a{}_{bc,d}\,\xi^d\frac{d\xi^b}{du}\frac{d\xi^c}{du} == 0$$

Pick out only first order terms in ξ and expand partial derivatives

$$\Gamma^a{}_{bc,d}\,\xi^d\frac{dx^b}{du}\frac{dx^c}{du} + \frac{d^2\xi^a}{dudu} + \Gamma^a{}_{bc}\frac{dx^c}{du}\frac{d\xi^b}{du} + \Gamma^a{}_{bc}\frac{dx^b}{du}\frac{d\xi^c}{du} == 0$$

$$\boxed{\frac{d^2\xi^a}{dudu} + \Gamma^a{}_{bc}\frac{dx^c}{du}\frac{d\xi^b}{du} + \Gamma^a{}_{bc}\frac{dx^b}{du}\frac{d\xi^c}{du} + \xi^d\frac{dx^b}{du}\frac{dx^c}{du}\partial_{x^d}\Gamma^a{}_{bc} == 0}$$

(The neglected second or higher order terms in ξ are:)

$$\Gamma^a{}_{bc,d}\,\xi^d\frac{dx^c}{du}\frac{d\xi^b}{du} + \Gamma^a{}_{bc,d}\,\xi^d\frac{dx^b}{du}\frac{d\xi^c}{du} + \Gamma^a{}_{bc}\frac{d\xi^b}{du}\frac{d\xi^c}{du} + \Gamma^a{}_{bc,d}\,\xi^d\frac{d\xi^b}{du}\frac{d\xi^c}{du}$$

Note: We must suppose that $\xi(u)$ and $d\xi(u)/du$ are *both* small! In the following counterexample **f[x]** is arbitrarily small, but **f'[x]** and **f[x]f'[x]** are arbitrarily big with suitable chosen ϵ and ω (Gegenbeispiel: ϵ beliebig klein, ω beliebig gross ⇒ **f[x]** beliebig klein, **f'[x]** und **f[x]f'[x]** beliebig gross.)

```
{f[x_] = ϵ Sin[ω/ϵ^2 x], f'[x], f[x] f'[x]} // Simplify
```

$$\left\{\epsilon\,\text{Sin}\left[\frac{x\,\omega}{\epsilon^2}\right],\ \frac{\omega\,\text{Cos}\left[\frac{x\,\omega}{\epsilon^2}\right]}{\epsilon},\ \frac{1}{2}\,\omega\,\text{Sin}\left[\frac{2\,x\,\omega}{\epsilon^2}\right]\right\}$$

We want to express the lhs as the absolute derivative of ξ plus other terms. So lets calculate the absolute derivative and subtract it from the lhs terms to see what we obtain.

```
Print["a) Absolute derivative of ξ"]
AbsoluteD[ξu[a], {u, u}]
% // ExpandAbsoluteD[labs, {{b, c}, {d, e}}]
% // ExpandAll
absolutedterm = MapAt[ExpandTotalD[labs, d], %, {3, 3}]
Print["b) Remaining terms on lhs"]
First[step1] - absolutedterm
Print["3rd and reindexed&symmetrized 4th terms cancel"]
MapAt[(# // IndexChange[{{d, c}, {e, b}}] // SymmetrizeSlots[Γ, 3, {1, {2, 3}}]) &, %%, 4]
Print["Use the second equation to reduce the 1st term"]
%% /. geodesicrule[d, e]
Print["Reindex 2nd and 3rd term and factor"]
MapAt[IndexChange[{{e, b}, {b, e}}], %%, 2]
MapAt[IndexChange[{{d, c}, {c, d}}], %, 3]
% // Factor
Print["Symmetrize Γ and use Riemann definition rule"]
%% // SymmetrizeSlots[Γ, 3, {1, {2, 3}}];
lastterms = % /. ChristoffelToRiemannRule
Print["c) Reassemble the terms"]
AbsoluteD[ξu[a], {u, u}] + lastterms == 0
Print["Reindex 2nd term and use antisymmetry of R. We
   get finally the equation of geodesic deviation, eq.(3.35),p.111:"]
%% // IndexChange[{{c, b}, {b, c}}]
(eqn[3, 35] = % // SymmetrizeSlots[R, 4, {-1, {3, 4}}]) // FrameBox // DisplayForm
```

a) Absolute derivative of ξ

$$\frac{D^2\xi^a}{dudu}$$

$$\xi^c\,\frac{dx^b}{du}\,\frac{d\Gamma^a{}_{bc}}{du}+\frac{d^2\xi^a}{dudu}+\Gamma^a{}_{bc}\left(\xi^c\,\frac{d^2x^b}{dudu}+\frac{dx^b}{du}\,\frac{d\xi^c}{du}\right)+\Gamma^a{}_{de}\,\frac{dx^d}{du}\left(\Gamma^e{}_{bc}\,\xi^c\,\frac{dx^b}{du}+\frac{d\xi^e}{du}\right)$$

$$\Gamma^a{}_{bc}\,\xi^c\,\frac{d^2x^b}{dudu}+\Gamma^a{}_{de}\,\Gamma^e{}_{bc}\,\xi^c\,\frac{dx^b}{du}\,\frac{dx^d}{du}+\xi^c\,\frac{dx^b}{du}\,\frac{d\Gamma^a{}_{bc}}{du}+\frac{d^2\xi^a}{dudu}+\Gamma^a{}_{bc}\,\frac{dx^b}{du}\,\frac{d\xi^c}{du}+\Gamma^a{}_{de}\,\frac{dx^d}{du}\,\frac{d\xi^e}{du}$$

$$\Gamma^a{}_{bc}\,\xi^c\,\frac{d^2x^b}{dudu}+\Gamma^a{}_{de}\,\Gamma^e{}_{bc}\,\xi^c\,\frac{dx^b}{du}\,\frac{dx^d}{du}+\frac{d^2\xi^a}{dudu}+\Gamma^a{}_{bc}\,\frac{dx^b}{du}\,\frac{d\xi^c}{du}+\Gamma^a{}_{de}\,\frac{dx^d}{du}\,\frac{d\xi^e}{du}+\xi^c\,\frac{dx^b}{du}\,\frac{dx^d}{du}\,\partial_{x^d}\Gamma^a{}_{bc}$$

b) Remaining terms on lhs

$$-\Gamma^a{}_{bc}\,\xi^c\,\frac{d^2x^b}{dudu}-\Gamma^a{}_{de}\,\Gamma^e{}_{bc}\,\xi^c\,\frac{dx^b}{du}\,\frac{dx^d}{du}+\Gamma^a{}_{bc}\,\frac{dx^c}{du}\,\frac{d\xi^b}{du}-\Gamma^a{}_{de}\,\frac{dx^d}{du}\,\frac{d\xi^e}{du}+\xi^d\,\frac{dx^b}{du}\,\frac{dx^c}{du}\,\partial_{x^d}\Gamma^a{}_{bc}-\xi^c\,\frac{dx^b}{du}\,\frac{dx^d}{du}\,\partial_{x^d}\Gamma^a{}_{bc}$$

3rd and reindexed&symmetrized 4th terms cancel

$$-\Gamma^a{}_{bc}\,\xi^c\,\frac{d^2x^b}{dudu}-\Gamma^a{}_{de}\,\Gamma^e{}_{bc}\,\xi^c\,\frac{dx^b}{du}\,\frac{dx^d}{du}+\xi^d\,\frac{dx^b}{du}\,\frac{dx^c}{du}\,\partial_{x^d}\Gamma^a{}_{bc}-\xi^c\,\frac{dx^b}{du}\,\frac{dx^d}{du}\,\partial_{x^d}\Gamma^a{}_{bc}$$

Use the second equation to reduce the 1st term

$$-\Gamma^a{}_{de}\,\Gamma^e{}_{bc}\,\xi^c\,\frac{dx^b}{du}\,\frac{dx^d}{du}+\Gamma^a{}_{bc}\,\Gamma^b{}_{de}\,\xi^c\,\frac{dx^d}{du}\,\frac{dx^e}{du}+\xi^d\,\frac{dx^b}{du}\,\frac{dx^c}{du}\,\partial_{x^d}\Gamma^a{}_{bc}-\xi^c\,\frac{dx^b}{du}\,\frac{dx^d}{du}\,\partial_{x^d}\Gamma^a{}_{bc}$$

Reindex 2nd and 3rd term and factor

$$-\Gamma^a{}_{de}\,\Gamma^e{}_{bc}\,\xi^c\,\frac{dx^b}{du}\,\frac{dx^d}{du}+\Gamma^a{}_{ec}\,\Gamma^e{}_{db}\,\xi^c\,\frac{dx^b}{du}\,\frac{dx^d}{du}+\xi^d\,\frac{dx^b}{du}\,\frac{dx^c}{du}\,\partial_{x^d}\Gamma^a{}_{bc}-\xi^c\,\frac{dx^b}{du}\,\frac{dx^d}{du}\,\partial_{x^d}\Gamma^a{}_{bc}$$

$$-\Gamma^a{}_{de}\,\Gamma^e{}_{bc}\,\xi^c\,\frac{dx^b}{du}\,\frac{dx^d}{du}+\Gamma^a{}_{ec}\,\Gamma^e{}_{db}\,\xi^c\,\frac{dx^b}{du}\,\frac{dx^d}{du}-\xi^c\,\frac{dx^b}{du}\,\frac{dx^d}{du}\,\partial_{x^d}\Gamma^a{}_{bc}+\xi^c\,\frac{dx^b}{du}\,\frac{dx^d}{du}\,\partial_{x^c}\Gamma^a{}_{bd}$$

$$-\xi^c\,\frac{dx^b}{du}\,\frac{dx^d}{du}\left(\Gamma^a{}_{de}\,\Gamma^e{}_{bc}-\Gamma^a{}_{ec}\,\Gamma^e{}_{db}+\partial_{x^d}\Gamma^a{}_{bc}-\partial_{x^c}\Gamma^a{}_{bd}\right)$$

Symmetrize Γ and use Riemann definition rule

$$-R^a{}_{bdc}\,\xi^c\,\frac{dx^b}{du}\,\frac{dx^d}{du}$$

c) Reassemble the terms

$$\frac{D^2\xi^a}{dudu}-R^a{}_{bdc}\,\xi^c\,\frac{dx^b}{du}\,\frac{dx^d}{du}==0$$

Reindex 2nd term and use antisymmetry of R. We get finally the *equation of geodesic deviation*, eq.(3.35),p.111:

$$\frac{D^2\xi^a}{dudu}-R^a{}_{cdb}\,\xi^b\,\frac{dx^c}{du}\,\frac{dx^d}{du}==0$$

$$\boxed{\frac{D^2\xi^a}{dudu}+R^a{}_{cbd}\,\xi^b\,\frac{dx^c}{du}\,\frac{dx^d}{du}==0}$$

Spezialfall 2D-Mannigfaltigkeit (special case 2D-manifold):

```
oldindices = BaseIndices;
DeclareBaseIndices[{1, 2}]
eqn[3, 35][[1]] == zerou[a]
% // EinsteinSum[] // EinsteinArray[];
% /. Ruddd[a_, a_, _, _] → 0;
% // SymmetrizeSlots[R, 4, {-1, {3, 4}}] // Simplify
DeclareBaseIndices[oldindices]
```

$$\frac{D^2\xi^a}{dudu}+R^a{}_{cbd}\,\xi^b\,\frac{dx^c}{du}\,\frac{dx^d}{du}==\text{zero}^a$$

$$\left\{\frac{D^2\xi^1}{dudu}+R^1{}_{212}\,\frac{dx^2}{du}\left(-\xi^2\,\frac{dx^1}{du}+\xi^1\,\frac{dx^2}{du}\right)==0,\ \frac{D^2\xi^2}{dudu}+R^2{}_{112}\,\frac{dx^1}{du}\left(-\xi^2\,\frac{dx^1}{du}+\xi^1\,\frac{dx^2}{du}\right)==0\right\}$$

3.5 EINSTEIN's field equations p. 112 - 114
3.6 Einstein's equation compared with Poisson's equation p. 115 - 116

```
Needs["TGeneralRelativity1`GeneralRelativity`"]
$PrePrint =.
labs = {x, δ, g, Γ};
DeclareBaseIndices[{0, 1, 2, 3}]
DefineTensorShortcuts[
 {{x, y, ξ, u}, 1},
 {{δ, g, G, R, T, η, h}, 2},
 {{Γ}, 3},
 {{R, Γ, g}, 4}]
DeclareTensorSymmetries[g, 2, {1, {1, 2}}]
DeclareTensorSymmetries[R, 2, {1, {1, 2}}]
DeclareTensorSymmetries[T, 2, {1, {1, 2}}]
DeclareTensorSymmetries[g, 3, {1, {1, 2}}]
DeclareTensorSymmetries[g, 4, {{1, {1, 2}}, {1, {3, 4}}}]
RiemannToChristoffelRule =
 LHSSymbolsToPatterns[{a, b, c, d}]@(Ruddd[d, a, b, c] → -Γudd[d, c, e] Γudd[e, a, b] +
     Γudd[d, b, e] Γudd[e, a, c] - PartialD[Γudd[d, a, b], c] + PartialD[Γudd[d, a, c], b])
(* Check *)RiemannRule[[2]] == %[[2]] // ExpandPartialD[labs]
ChristoffelUpToMetricRule = LHSSymbolsToPatterns[{a, b, c}]@(Γudd[a, b, c] →
    1/2 guu[a, d] (PartialD[gdd[d, c], b] + PartialD[gdd[b, d], c] - PartialD[gdd[b, c], d]))
(* Check *)ChristoffelDownRule[[2]] == gdd[a, e] (%[[2]] /. a → e) // MetricSimplify[g] //
 Simplify
ΓΓRule =
 (Tensor[Γ, List[a_, _, _], List[_, b_, c_]] Tensor[Γ, List[d_, _, _], List[_, e_, f_]] →
   (ChristoffelUpToMetricRule[[2]] /. d → r)
    (ChristoffelUpToMetricRule[[2]] /. {d → s, a → d, b → e, c → f}))
PDChristoffelUpToMetricRule = LHSSymbolsToPatterns[{ι}]@
  (PartialD[#, ι] & /@ChristoffelUpToMetricRule)
SetAttributes[c, Constant]
```

$R^{d\_}{}_{a\_b\_c\_} \to -\Gamma^{d}{}_{ce}\,\Gamma^{e}{}_{ab} + \Gamma^{d}{}_{be}\,\Gamma^{e}{}_{ac} - \Gamma^{d}{}_{ab,c} + \Gamma^{d}{}_{ac,b}$

True

$\Gamma^{a\_}{}_{b\_c\_} \to \frac{1}{2}\, g^{ad}\left(-g_{bc,d} + g_{bd,c} + g_{dc,b}\right)$

True

$\Gamma^{a\_}{}_{\_b\_c\_}\,\Gamma^{d\_}{}_{\_e\_f\_} \to \frac{1}{4}\, g^{ar}\, g^{ds}\left(-g_{bc,r} + g_{br,c} + g_{rc,b}\right)\left(-g_{ef,s} + g_{es,f} + g_{sf,e}\right)$

$\Gamma^{a\_}{}_{b\_c\_,\iota\_} \to \frac{1}{2}\left(g^{ad}{}_{,\iota}\left(-g_{bc,d} + g_{bd,c} + g_{dc,b}\right) + g^{ad}\left(-g_{bc,d,\iota} + g_{bd,c,\iota} + g_{dc,b,\iota}\right)\right)$

Classical Poisson's equation

("As is the case with the vector products discussed above, the common differential operations in three dimensions are defined in terms of Cartesian coordinates. If you are working in another coordinate system and you wish to compute these quantities, you must, in principle, first transform into the Cartesian system and then do the calculation. When you specify the coordinate system in functions like Laplacian, Grad, and so on, this transformation is done automatically.", *Mathematica*)

```
<< Calculus`VectorAnalysis`
Print["Laplacian Δ: a) definition (in Cartesian and spherical coordinates)"]
Laplacian[V[x, y, z], Cartesian[x, y, z]]
L = Laplacian[V[r, θ, ϕ], Spherical[r, θ, ϕ]] // Simplify
Print["b) linearity"]
Expand /@ (Laplacian[a f[r, θ, ϕ] + b g[r, θ, ϕ], Spherical[r, θ, ϕ]] ==
    a Laplacian[f[r, θ, ϕ], Spherical[r, θ, ϕ]] + b Laplacian[ g[r, θ, ϕ], Spherical[r, θ, ϕ]])
Print["c) special case: spherical symmetric V"]
(L /. V^(_,m_,n_)[r, θ, ϕ] :> 0 /; (m > 0 || n > 0)) // Simplify
```

Laplacian Δ: a) definition (in Cartesian and spherical coordinates)

$V^{(0,0,2)}[x, y, z] + V^{(0,2,0)}[x, y, z] + V^{(2,0,0)}[x, y, z]$

$$\frac{1}{r^2}\left(\mathrm{Csc}[\theta]^2 V^{(0,0,2)}[r, \theta, \phi] + \mathrm{Cot}[\theta] V^{(0,1,0)}[r, \theta, \phi] + V^{(0,2,0)}[r, \theta, \phi] + 2 r V^{(1,0,0)}[r, \theta, \phi] + r^2 V^{(2,0,0)}[r, \theta, \phi]\right)$$

b) linearity

True

c) special case: spherical symmetric V

$$\frac{2 V^{(1,0,0)}[r, \theta, \phi]}{r} + V^{(2,0,0)}[r, \theta, \phi]$$

```
Print["Example: gravitational potential and field for a homogenous sphere (Newton)"]
Print["gravitational potential V(r) (a.u.)"]
V[r_] = (r^2 (1 - UnitStep[r - 1]) + (3 - 2 r^-1) UnitStep[r - 1]) / 6 - 1 / 2
Simplify[V[r]]
% // FullForm
Print["V(r→∞)=", Limit[V[r], r → Infinity]]
Print["Laplacian ΔV(r)~ρ(r)"]
LV[r_] = Laplacian[V[r], Spherical[r, θ, ϕ]] // Simplify
% // FullForm
Print["gravitational field g(r)"]
g[r_] = -Grad[V[r], Spherical[r, θ, ϕ]] // Simplify
```

Example: gravitational potential and field for a homogenous sphere (Newton)

gravitational potential V(r) (a.u.)

$$-\frac{1}{2} + \frac{1}{6}\left(r^2 (1 - \mathrm{UnitStep}[-1 + r]) + \left(3 - \frac{2}{r}\right) \mathrm{UnitStep}[-1 + r]\right)$$

$$\begin{cases} -\frac{1}{3 r} & r \geq 1 \\ \frac{1}{6}\left(-3 + r^2\right) & \text{True} \end{cases}$$

Piecewise[List[List[Times[Rational[-1, 3], Power[r, -1]], GreaterEqual[r, 1]]],
 Times[Rational[1, 6], Plus[-3, Power[r, 2]]]]

V(r→∞)=0

Laplacian ΔV(r)~ρ(r)

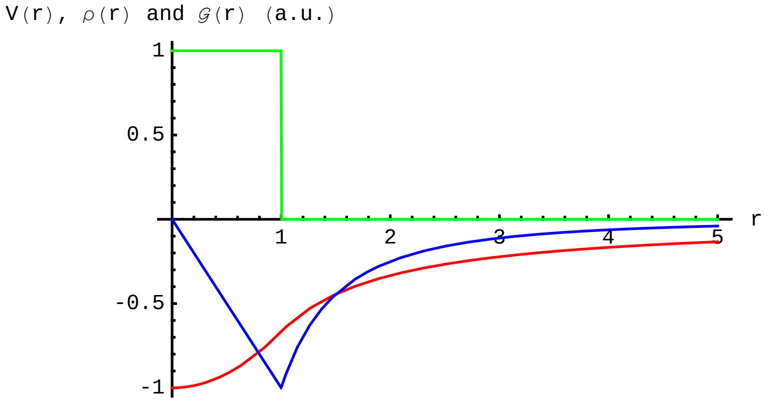
$$\begin{cases} 1 & r < 1 \end{cases}$$

Piecewise[List[List[1, Less[r, 1]]], 0]

gravitational field g(r)

$$\left\{\begin{cases} -\frac{1}{3r^2} & r \geq 1 \\ -\frac{r}{3} & \text{True} \end{cases}, 0, 0\right\}$$

```
Plot[Evaluate[{2 V[r], LV[r], 3 𝒢[r][[1]]}],
  {r, 0, 5}, PlotStyle → {Hue[0], Hue[1 / 3], Hue[2 / 3]},
  AxesLabel → {"r", "V(r), ρ(r) and 𝒢(r) (a.u.)"}, ImageSize → 72 × 5];
```



★ Plot of gravitational potential V(r) (red), density ρ(r) (green) and gravitational field $\mathcal{G}$(r) (blue) for a homogenous sphere (Newton). ★

▪ I) Einstein's equation (Einsteinsche Gleichung, Albert Einstein, 25. November 1915, Preußischen Akademie der Wissenschaft, Berlin)

```
(eqn[3, 38] = Ruu[μ, ν] - 1 / 2 Tensor[R] guu[μ, ν] == κ Tuu[μ, ν]) // FrameBox // DisplayForm
Print["where ", eqn[3, 38][[1]] == Guu[μ, ν]]
```

$$\boxed{-\frac{1}{2} R\, g^{\mu\nu} + R^{\mu\nu} == \kappa\, T^{\mu\nu}}$$

where $-\frac{1}{2} R\, g^{\mu\nu} + R^{\mu\nu} == G^{\mu\nu}$

I) G^{ab} is sym., $G^{ab}{}_{;b} == 0$ and G^{ab} is function of max. 2nd order derivatives of $g^{\mu\nu}$.

An alternative form of Einstein's equation is

```
(eqn[3, 39] = Ruu[μ, ν] == κ (Tuu[μ, ν] - 1 / 2 Tensor[T] guu[μ, ν])) // FrameBox // DisplayForm
Print["where"]
Tensor[T] == Tud[μ, μ]
Print["Special case: Empty spacetime (T^μν = 0) field equations"]
(eqn[3, 40] = Ruu[μ, ν] == 0) // FrameBox // DisplayForm
```

$$\boxed{R^{\mu\nu} == \kappa\left(-\frac{1}{2} T\, g^{\mu\nu} + T^{\mu\nu}\right)}$$

where

$T == T^{\mu}{}_{\mu}$

Special case: Empty spacetime $(T^{\mu\nu} = 0)$ field equations

$$\boxed{R^{\mu\nu} == 0}$$

The structure of Einstein's equation (Die Struktur der Einsteinschen Gleichung)

John Baez: "In 4 dimensions, it takes 20 numbers to specify the curvature at each point. 10 of these numbers are captured by the *Ricci tensor*, while the remaining 10 are captured by the *Weyl tensor*."

Ted Bunn: "There's extra information in the [electric and magnetic] fields beyond just what the sources of the fields can tell you. After all, you could have an electromagnetic wave passing by. It needn't have any source, but it still alters the fields. So in electromagnetism, knowing all about the sources isn't enough to specify the fields. In general relativity, knowing all about the sources (the stress-energy tensor T) isn't enough to tell you all about the curvature. In both cases, you can supplement the source information with some extra initial conditions to get a unique solution."

```
Print["(General) Einstein's equation:"]
(*EFE Einstein*)Guu[μ, ν] == eqn[3, 38][[2]]
(*EFE curvature scalar&Ricci*)eqn[3, 38]
EFERicci =
 % /. {Tensor[R] → guu[a, b] Rdd[a, b], Ruu[μ, ν] -> guu[a, μ] guu[b, ν] Rdd[a, b]} // Simplify
EFERiemann = (% // Simplify) /. Rdd[a_, b_] → Ruddd[c, a, b, c]
EFEΓ = % /. RiemannToChristoffelRule
% /. ΓΓRule /. PDChristoffelUpToMetricRule // SymmetrizeSlots[];
MapAt[SimplifyTensorSum, %, 1];
EFEMetric = (MapAt[FullSimplify, %, 1] /. r → f // SymmetrizeSlots[])
```

(*General*) *Einstein's equation:*

$$G^{\mu\nu} == \kappa\, T^{\mu\nu}$$

$$-\frac{1}{2} R\, g^{\mu\nu} + R^{\mu\nu} == \kappa\, T^{\mu\nu}$$

$$\frac{1}{2}\left(2 g^{a\mu} g^{b\nu} - g^{ab} g^{\mu\nu}\right) R_{ab} == \kappa\, T^{\mu\nu}$$

$$\frac{1}{2}\left(2 g^{a\mu} g^{b\nu} - g^{ab} g^{\mu\nu}\right) R^{c}{}_{abc} == \kappa\, T^{\mu\nu}$$

$$\frac{1}{2}\left(2 g^{a\mu} g^{b\nu} - g^{ab} g^{\mu\nu}\right)\left(-\Gamma^{c}{}_{ce}\Gamma^{e}{}_{ab} + \Gamma^{c}{}_{be}\Gamma^{e}{}_{ac} - \Gamma^{c}{}_{ab,c} + \Gamma^{c}{}_{ac,b}\right) == \kappa\, T^{\mu\nu}$$

$$\frac{1}{8}\left(2 g^{a\mu} g^{b\nu} - g^{ab} g^{\mu\nu}\right)\Big(2 g^{cd}{}_{,c}\left(g_{ab,d} - g_{ad,b} - g_{bd,a}\right) + 2 g^{cd}{}_{,b}\left(-g_{ac,d} + g_{ad,c} + g_{cd,a}\right) + g^{cd}\Big(g^{ef}\Big(-\left(g_{ac,e} - g_{ae,c} - g_{ce,a}\right)\left(g_{bd,f} - g_{bf,d} + g_{df,b}\right) + \left(g_{ab,e} - g_{ae,b} - g_{be,a}\right)\left(g_{cd,f} - g_{cf,d} + g_{df,c}\right)\Big) + 2\left(g_{ab,c,d} - g_{ac,b,d} - g_{bd,a,c} + g_{cd,a,b}\right)\Big)\Big) == \kappa\, T^{\mu\nu}$$

```
oldindices = BaseIndices;
DeclareBaseIndices[{1, 2, 3}]
Print[NDim, "-dimensional Einstein's equation:"]
Guu[μ, ν] == eqn[3, 38][[2]]
eqn[3, 38]
% // EinsteinSum[] // EinsteinArray[] // SymmetrizeSlots[] // Flatten // Union // ColumnForm
EFERicci
% // EinsteinSum[] // EinsteinArray[] // SymmetrizeSlots[] // Flatten;
MapAt[Simplify, #, 1] & /@ % // Union // ColumnForm
DeclareBaseIndices[oldindices]
```

3-*dimensional Einstein's equation:*

$$G^{\mu\nu} == \kappa\, T^{\mu\nu}$$

$$-\frac{1}{2} R\, g^{\mu\nu} + R^{\mu\nu} == \kappa\, T^{\mu\nu}$$

$$-\frac{1}{2} R\, g^{11} + R^{11} == \kappa\, T^{11}$$
$$-\frac{1}{2} R\, g^{12} + R^{12} == \kappa\, T^{12}$$
$$-\frac{1}{2} R\, g^{13} + R^{13} == \kappa\, T^{13}$$
$$-\frac{1}{2} R\, g^{22} + R^{22} == \kappa\, T^{22}$$
$$-\frac{1}{2} R\, g^{23} + R^{23} == \kappa\, T^{23}$$
$$-\frac{1}{2} R\, g^{33} + R^{33} == \kappa\, T^{33}$$

$$\frac{1}{2}\left(2\, g^{a\mu}\, g^{b\nu} - g^{ab}\, g^{\mu\nu}\right) R_{ab} == \kappa\, T^{\mu\nu}$$

$$\frac{1}{2}\left(-g^{11} g^{23} R_{11} + 2 g^{13} g^{22} R_{12} + 2 g^{12}\left(g^{13} R_{11} + g^{33} R_{13}\right) + g^{22} g^{23} R_{22} + 2 g^{22} g^{33} R_{23} + g^{23} g^{33} R_{33}\right) == \kappa\, T^{23}$$
$$\frac{1}{2}\left(g^{11}\left(g^{12} R_{11} + 2 g^{22} R_{12} + 2 g^{23} R_{13}\right) + 2 g^{13}\left(g^{22} R_{23} + g^{23} R_{33}\right) + g^{12}\left(g^{22} R_{22} - g^{33} R_{33}\right)\right) == \kappa\, T^{12}$$
$$\frac{1}{2}\left(\left(g^{11}\right)^2 R_{11} + \left(g^{12}\right)^2 R_{22} + 2 g^{12} g^{13} R_{23} + \left(g^{13}\right)^2 R_{33} + g^{11}\left(g^{12} R_{12} + g^{13} R_{13} - \frac{1}{2} g^{22} R_{22} - g^{23} R_{23} - \frac{1}{2} g^{33} R_{33}\right) == \kappa\right.$$
$$\frac{1}{2}\left(g^{11}\left(g^{13} R_{11} + 2 g^{23} R_{12} + 2 g^{33} R_{13}\right) + 2 g^{12}\left(g^{23} R_{22} + g^{33} R_{23}\right) + g^{13}\left(-g^{22} R_{22} + g^{33} R_{33}\right)\right) == \kappa\, T^{13}$$
$$\left(g^{12}\right)^2 R_{11} + g^{12}\left(g^{22} R_{12} + 2 g^{23} R_{13}\right) + \frac{1}{2}\left(-g^{11} g^{22} R_{11} - 2 g^{13} g^{22} R_{13} + \left(g^{22}\right)^2 R_{22} + 2 g^{22} g^{23} R_{23} + 2\left(g^{23}\right)^2 R_{33} -\right.$$
$$\left(g^{13}\right)^2 R_{11} + g^{13}\left(2 g^{23} R_{12} + g^{33} R_{13}\right) + \frac{1}{2}\left(-g^{11} g^{33} R_{11} - 2 g^{12} g^{33} R_{12} + 2\left(g^{23}\right)^2 R_{22} - g^{22} g^{33} R_{22} + 2 g^{23} g^{33} R_{23} +\right.$$

(Quite complicate! And the Ricci tensor is not yet fully written out as a function of the metric tensor...)

■ II) Support for Einstein's equation by comparing the equation of geodesic deviation with its Newtonian counterpart (empty spacetime)

GR: equation of geodesic deviation with proper time τ as affine parameter:

```
AbsoluteD[ξu[μ], {τ, τ}] == -Ruddd[μ, σ, ν, ρ] ξu[ν] TotalD[xu[σ], τ] TotalD[xu[ρ], τ]
Print["GR tidal tensor of differential acceleration K_GR:"]
Coefficient[%%[[2]], -ξu[ν]];
K_GR = MapAt[IndexChange[{{ν, ρ}, {ρ, ν}}, -1], %, 1]
```

$$\frac{D^2 \xi^{\mu}}{d\tau d\tau} == -R^{\mu}{}_{\sigma\nu\rho}\, \xi^{\nu}\, \frac{dx^{\rho}}{d\tau}\, \frac{dx^{\sigma}}{d\tau}$$

GR tidal tensor of differential acceleration K_{GR}:

$$-R^{\mu}{}_{\sigma\rho\nu}\, \frac{dx^{\rho}}{d\tau}\, \frac{dx^{\sigma}}{d\tau}$$

Newton: we consider two particles moving under gravity on nearby paths given by coordinates x and y in space. Here η = diag(1,1,1).

```
oldindices = BaseIndices;
DeclareBaseIndices[{1, 2, 3}]
DefineTensorShortcuts[η, 2]
SetTensorValueRules[ηuu[i, j], DiagonalMatrix[{1, 1, 1}]]
{Neqn1 = TotalD[yu[i], {t, t}] == -ηuu[i, k] PartialD[{y, δ, η, Γ}][Tensor[V], yu[k]],
 Neqn2 = TotalD[xu[i], {t, t}] == -ηuu[i, k] PartialD[{x, δ, η, Γ}][Tensor[V], xu[k]]}
Inner[Subtract, Neqn1, Neqn2, Equal]
% /. HoldPattern[TotalD[a_, d_] - TotalD[b_, d_]] → TotalD[NestedTensor[a - b], d] //
 Simplify
step1 = % /. yu[i] - xu[i] → ξu[i] // UnnestTensor
Print["For the derivative on the y curve
   we expand about the corresponding point on the x curve"]
derivativerule =
```

```
 PartialD[{y, δ, η, Γ}][Tensor[V], yu[k_]] → PartialD[{x, δ, η, Γ}][Tensor[V], xu[k]] +
    PartialD[{x, δ, η, Γ}][Tensor[V], {xu[j], xu[k]}] ξu[j]
step1 /. derivativerule
Print["Newtonian tidal tensor of differential acceleration K_N:"]
Coefficient[%%[[2]], -ξu[j]]
Print["Its trace is the Laplacian of the gravitational potential ΔV  (η=diag(1,1,1)):"]
%% /. i → j // EinsteinSum[] // ToArrayValues[]
ClearTensorValues[ηuu[i, j]]
DeclareBaseIndices[oldindices]
```

$$\left\{\frac{d^2 y^i}{dt\,dt} == -\eta^{ik}\,\partial_{y^k} V,\ \frac{d^2 x^i}{dt\,dt} == -\eta^{ik}\,\partial_{x^k} V\right\}$$

$$-\frac{d^2 x^i}{dt\,dt} + \frac{d^2 y^i}{dt\,dt} == \eta^{ik}\,\partial_{x^k} V - \eta^{ik}\,\partial_{y^k} V$$

$$\frac{d^2\left(-x^i + y^i\right)}{dt\,dt} == \eta^{ik}\left(\partial_{x^k} V - \partial_{y^k} V\right)$$

$$\frac{d^2 \xi^i}{dt\,dt} == \eta^{ik}\left(\partial_{x^k} V - \partial_{y^k} V\right)$$

For the derivative on the y curve we expand about the corresponding point on the x curve

$$\partial_{y^{k\_}} V \to \xi^j\,\partial_{x^j,x^k} V + \partial_{x^k} V$$

$$\frac{d^2 \xi^i}{dt\,dt} == -\eta^{ik}\,\xi^j\,\partial_{x^j,x^k} V$$

Newtonian tidal tensor of differential acceleration K_N:

$$\eta^{ik}\,\partial_{x^j,x^k} V$$

Its trace is the Laplacian of the gravitational potential ΔV (η=diag(1,1,1)):

$$\partial_{x^1,x^1} V + \partial_{x^2,x^2} V + \partial_{x^3,x^3} V$$

Now the empty space field equation of Newtonian gravitation is $\Delta V = 0$, or equivalently Trace(K_N) = 0.
This suggests that in empty spacetime we should have Trace(K_{GR}) = 0...

```
(K_GR /. ν → μ) == 0
% /. Ruddd[a_, b_, c_, a_] → Rdd[b, c] // Simplify
Print["Since this should hold for arbitrary tangent vectors to geodesics"]
r = Table[a[i, j], {i, 1, 4}, {j, 1, 4}];
v1 = Table[va[i], {i, 1, 4}];
v2 = Table[vb[i], {i, 1, 4}];
r /. Flatten[SolveAlways[v1.r.v2 == 0, Union[v1, v2]]]
Rdd[σ, ρ] == 0
```

$$-R^{\mu}{}_{\sigma\rho\mu}\,\frac{dx^\rho}{d\tau}\,\frac{dx^\sigma}{d\tau} == 0$$

$$R_{\sigma\rho}\,\frac{dx^\rho}{d\tau}\,\frac{dx^\sigma}{d\tau} == 0$$

Since this should hold for arbitrary tangent vectors to geodesics

{{0, 0, 0, 0}, {0, 0, 0, 0}, {0, 0, 0, 0}, {0, 0, 0, 0}}

$$R_{\sigma\rho} == 0$$

This accords with Einstein's empty spacetime field equations.

■ III) Einstein's equation compared with Poisson's Equation (with matter)

The object here is to show that the Einstein field equations agree with the weak gravity Newtonian equations when matter is present.

As in Section 2.7, we will use a nearly Cartesian coordinate system with $g_{\mu\nu} = \eta_{\mu\nu} + h_{\mu\nu}$, where $h_{\mu\nu}$ and $h_{\mu\nu,\rho}$ are small, and the *extended* (!?) quasi-static condition should hold: $h_{\mu\nu,0} \ll h_{\mu\nu,i}$ and $\partial_{x^0} \Gamma^i{}_{0i}$ small.

```
Print["Stress tensor for a perfect fluid"]
Tuu[μ, ν] == (Tensor[ρ] + Tensor[P] / c^2) uu[μ] uu[ν] - Tensor[P] guu[μ, ν]
Print["Stress tensor approximation for
   a low speed (v<<c), low pressure (P/c^2<<ρ) perfect fluid."]
Tdd[μ, ν] == ρ ud[μ] ud[ν];
Tddapprox = LHSSymbolsToPatterns[{μ, ν}][Rule @@ %]
Print["Raising index and contracting"]
guu[ν, μ] # & /@ %%%
% // MetricSimplify[g]
% /. {Tud[a_, a_] → Tensor[T], uu[a_] ud[a_] → c^2};
Tapprox = Rule @@ %
```

Stress tensor for a perfect fluid

$$T^{\mu\nu} == -P\, g^{\mu\nu} + \left(\frac{P}{c^2} + \rho\right) u^\mu u^\nu$$

Stress tensor approximation for a low speed (v<<c), low pressure (P/c²<<ρ) perfect fluid.

$$T_{\mu\_\nu\_} \to \rho\, u_\mu u_\nu$$

Raising index and contracting

$$g^{\nu\mu} T_{\mu\nu} == \rho\, g^{\nu\mu} u_\mu u_\nu$$

$$T^\nu{}_\nu == \rho\, u_\nu u^\nu$$

$$T \to c^2 \rho$$

```
Print["Einstein's equation eq.(3.39) in covariant form"]
Rdd[μ, ν] == κ (Tdd[μ, ν] - 1 / 2 Tensor[T] gdd[μ, ν])
Print["Using the stress tensor approximation"]
%% /. Tddapprox /. Tapprox
Print["Taking the 00 Ricci tensor component"]
%% /. {μ → 0, ν → 0}
{ud[0] → c, gdd[0, 0] → 1};
Print["Using ", %]
(eqn[3, 47] = %%% /. %%)
```

Einstein's equation eq.(3.39) in covariant form

$$R_{\mu\nu} == \kappa\left(-\frac{1}{2} T\, g_{\mu\nu} + T_{\mu\nu}\right)$$

Using the stress tensor approximation

$$R_{\mu\nu} == \kappa\left(-\frac{1}{2} c^2 \rho\, g_{\mu\nu} + \rho\, u_\mu u_\nu\right)$$

Taking the 00 Ricci tensor component

$$R_{00} == \kappa \left(-\frac{1}{2} c^2 \rho\, g_{00} + \rho\, (u_0)^2\right)$$

Using $\{u_0 \to c,\ g_{00} \to 1\}$

$$R_{00} == \frac{1}{2} c^2 \kappa\, \rho$$

```
Print["Riemann tensor"]
Equal @@ RiemannRule
Print["Contracting to obtain the 00 Ricci tensor component"]
%% /. Thread[{a, b, c, d, e} → {0, 0, μ, μ, ν}]
% /. Ruddd[a_, b_, c_, a_] → Rdd[b, c]
Print["With small h_{μν,ρ}, the Γ·Γ are small"]
%% /. Γudd[a_, _, _] Γudd[b_, _, _] → 0
Print["Expand into temporal and spatial parts and simplify"]
%% // PartialSum[0, {i}]
%%% // EinsteinSum[]
Print["Using the extended quasi-static approximation:"]
%%% /. PartialD[labs][_, xu[0]] → 0
Print["We use (following eq.(2.78), sec. 2.7):"]
Γrule[j_] = Γudd[i_, 0, 0] → -1 / 2 ηuu[i, j] PartialD[labs][hdd[0, 0], xu[j]]
%%% /. Γrule[j]
Print["Minkowsky metric η is constant"]
eqn1 = %% // NondependentPartialD[{η, x}]
```

Riemann tensor

$$R^{d}{}_{abc} == -\Gamma^{d}{}_{ce}\, \Gamma^{e}{}_{ab} + \Gamma^{d}{}_{be}\, \Gamma^{e}{}_{ac} - \partial_{x^c} \Gamma^{d}{}_{ab} + \partial_{x^b} \Gamma^{d}{}_{ac}$$

Contracting to obtain the 00 Ricci tensor component

$$R^{\mu}{}_{00\mu} == -\Gamma^{\mu}{}_{\mu\nu}\, \Gamma^{\nu}{}_{00} + \Gamma^{\mu}{}_{0\nu}\, \Gamma^{\nu}{}_{0\mu} - \partial_{x^\mu} \Gamma^{\mu}{}_{00} + \partial_{x^0} \Gamma^{\mu}{}_{0\mu}$$

$$R_{00} == -\Gamma^{\mu}{}_{\mu\nu}\, \Gamma^{\nu}{}_{00} + \Gamma^{\mu}{}_{0\nu}\, \Gamma^{\nu}{}_{0\mu} - \partial_{x^\mu} \Gamma^{\mu}{}_{00} + \partial_{x^0} \Gamma^{\mu}{}_{0\mu}$$

With small $h_{\mu\nu,\rho}$, the $\Gamma\cdot\Gamma$ are small

$$R_{00} == -\partial_{x^\mu} \Gamma^{\mu}{}_{00} + \partial_{x^0} \Gamma^{\mu}{}_{0\mu}$$

Expand into temporal and spatial parts and simplify

$$R_{00} == -\partial_{x^i} \Gamma^{i}{}_{00} + \partial_{x^0} \Gamma^{i}{}_{0i}$$

$$R_{00} == -\partial_{x^1} \Gamma^{1}{}_{00} + \partial_{x^0} \Gamma^{1}{}_{01} - \partial_{x^2} \Gamma^{2}{}_{00} + \partial_{x^0} \Gamma^{2}{}_{02} - \partial_{x^3} \Gamma^{3}{}_{00} + \partial_{x^0} \Gamma^{3}{}_{03}$$

Using the extended quasi-static approximation:

$$R_{00} == -\partial_{x^i} \Gamma^{i}{}_{00}$$

We use (following eq.(2.78), sec. 2.7):

$$\Gamma^{i\_}{}_{00} \to -\frac{1}{2} \eta^{ij}\, \partial_{x^j} h_{00}$$

$$R_{00} == \frac{1}{2} \left(\eta^{ij}\, \partial_{x^i, x^j} h_{00} + \partial_{x^j} h_{00}\, \partial_{x^i} \eta^{ij}\right)$$

Minkowsky metric η is constant

$$R_{00} == \frac{1}{2} \eta^{ij} \partial_{x^i,x^j} h_{00}$$

We combine now the two results for R_{00}.

```
-eqn1[[2]] == -eqn[3, 47][[2]]
hdd[0, 0] → 2 Tensor[V] / c^2;
Print["Using ", %]
%%% /. %%;
c^2 # &/@%
κ → -8 π G / c^4;
Print["With ", %, "..."]
%%% /. %%
```

$$-\frac{1}{2} \eta^{ij} \partial_{x^i,x^j} h_{00} == -\frac{1}{2} c^2 \kappa \rho$$

Using $h_{00} \to \frac{2 V}{c^2}$

$$-\eta^{ij} \partial_{x^i,x^j} V == -\frac{1}{2} c^4 \kappa \rho$$

With $\kappa \to -\frac{8 G \pi}{c^4}$...

$$-\eta^{ij} \partial_{x^i,x^j} V == 4 G \pi \rho$$

...we obtain the Poisson equation for Newtonian gravity.

3.7 The Schwarzschild solution p. 116 - 119

```
Needs["TGeneralRelativity1`GeneralRelativity`"]
$PrePrint =.
labs = {x, δ, g, Γ};
DeclareBaseIndices[{0, 1, 2, 3}]
DefineTensorShortcuts[
 {{x, dx}, 1},
 {{δ, g, η, h}, 2},
 {{Γ}, 3}]
SetTensorValues[δud[i, j], IdentityMatrix[NDim]]
SetAttributes[c, Constant]
varnames = {c t, r, θ, ϕ};
dvarnames = Dt[%]; (* useSchwarzschild will apply UseCoordinates for both x and dx. *)
useSchwarzschild = Composition[UseCoordinates[%, dx], UseCoordinates[%%, x]];
```

Note: In deriving the Schwarzschild metric I'm using a little different path than FN and David Park.

Schwarzschild's (exact) solution (1916) of Einstein's equation considers the case
(a) That the field is static. (But: (b) and (c) ⟹ (a), Birkhoff, 1923!)
(b) That the field is spherically symmetric.
(c) That spacetime is empty outside a (spherically symmetric) body of mass M.
(d) That spacetime is asymptotically flat.

DANGER: Suggestive labels for coordinates ahead!

(e) That spacetime is coordinatized by $\{x^0, x^i\} \stackrel{!}{=} \{c\,t, r, \theta, \phi\}$, where $c\ t$ is a timelike coordinate (Schwarzschild coordinates).

```
Print["Schwarzschild's ansatz for conditions (a)-(e) (proof of uniqueness?):"]
cmetric = DiagonalMatrix[{A[r], -B[r], -r^2, -r^2 Sin[θ]^2}];
gdd[μ, ν] == (% // MatrixForm)
metric = cmetric // CoordinatesToTensors[varnames];
SetMetricValueRules[g, metric]
gdd[μ, ν] == (gdd[μ, ν] // ToArrayValues[] // MatrixForm)
Print["Line element with assumed form of metric"]
ds^2 == gdd[μ, ν] dxu[μ] dxu[ν]
% // ToArrayValues[]
(eqn[3, 51] = % // useSchwarzschild) // TraditionalForm
Print["A surface with constant c t and r has line element
   dσ^2 and area element dS and so has the geometry of a sphere S:"]
(eqn[3, 52] = (dσ^2 == -eqn[3, 51][[2]] /. {Dt[t] → 0, Dt[r] → 0})) // TraditionalForm
dS == (eqn[3, 52][[2]] /. Plus → Times // Sqrt // PowerExpand) // TraditionalForm
Print["∫_S dS=", ∫_{-π}^{π} ∫_0^π %[[2]] / (Dt[θ] Dt[ϕ]) dθ dϕ]
```

Schwarzschild's *ansatz* for conditions (a)-(e) (proof of uniqueness?):

$$g_{\mu\nu} == \begin{pmatrix} A[r] & 0 & 0 & 0 \\ 0 & -B[r] & 0 & 0 \\ 0 & 0 & -r^2 & 0 \\ 0 & 0 & 0 & -r^2 \operatorname{Sin}[\theta]^2 \end{pmatrix}$$

$$g_{\mu\nu} == \begin{pmatrix} A[x^1] & 0 & 0 & 0 \\ 0 & -B[x^1] & 0 & 0 \\ 0 & 0 & -(x^1)^2 & 0 \\ 0 & 0 & 0 & -\operatorname{Sin}[x^2]^2 (x^1)^2 \end{pmatrix}$$

Line element with assumed form of metric

$$ds^2 == dx^\mu \, dx^\nu \, g_{\mu\nu}$$

$$ds^2 == A[x^1] (dx^0)^2 - B[x^1] (dx^1)^2 - (dx^2)^2 (x^1)^2 - \operatorname{Sin}[x^2]^2 (dx^3)^2 (x^1)^2$$

$$ds^2 = -B(r)\,(d\,r)^2 + c^2 A(r)\,(d\,t)^2 - r^2\,(d\,\theta)^2 - r^2\,(d\,\phi)^2 \sin^2(\theta)$$

A surface with constant c t and r has line
  element $d\sigma^2$ and area element dS and so has the geometry of a sphere S:

$$d\sigma^2 = r^2\,(d\,\theta)^2 + r^2\,(d\,\phi)^2 \sin^2(\theta)$$

$$dS = r^2\, d\,\theta\, d\,\phi \sin(\theta)$$

$$\int_S dS = 4\,\pi\, r^2$$

Comments:

(a): The trial solution is static because none of the metric components depend upon c t.

(b): The trial solution is spherically symmetric. The area of the sphere S is $4\pi r^2$, but if B[r]≠1, then r is not the radius (= distance from center), but a "radial" coordinate.

(c): For empty spacetime we must set the Ricci tensor equal to zero.

(d): Asymptotic flatness requires that as $r \to \infty$, $A[r] \to 1$ and $B[r] \to 1$. Then Schwarzschild metric converges to the flat space metric (in spherical coordinates).

```
Print["Calculate the Christoffel symbols in
   preparation for calculating the covariant Riemann tensor."]
SetChristoffelValueRules[xu[i], metric, Γ]
SelectedTensorRules[Γ, Γudd[_, a_, b_] /; OrderedQ[{a, b}]]
```

```
% // UseCoordinates[varnames] // TableForm
Print["Calculate the covariant Riemann
   tensor in preparation for calculating the Ricci tensor."]
riemanndown = CalculateRiemannd[labs, Identity, Simplify];
Short[%, 7]
Print["Calculate the Ricci tensor Rμν."]
(*The following calculates the Ricci and other tensors,
but we are only interested in the Ricci tensor.*)
{riemannup, ricci, scalarcurve, einstein} = CalculateRRRG[g, riemanndown];
ricci // MatrixForm
```

Calculate the Christoffel symbols in preparation for calculating the covariant Riemann tensor.

$$\left\{\Gamma^{0}{}_{01} \to \frac{A'[x^1]}{2\,A[x^1]},\ \Gamma^{1}{}_{00} \to \frac{A'[x^1]}{2\,B[x^1]},\ \Gamma^{1}{}_{11} \to \frac{B'[x^1]}{2\,B[x^1]},\ \Gamma^{1}{}_{22} \to -\frac{x^1}{B[x^1]},\ \Gamma^{1}{}_{33} \to -\frac{\mathrm{Sin}[x^2]^2\,x^1}{B[x^1]},\right.$$
$$\left.\Gamma^{2}{}_{12} \to (x^1)^{-1},\ \Gamma^{2}{}_{33} \to -\mathrm{Cos}[x^2]\,\mathrm{Sin}[x^2],\ \Gamma^{3}{}_{13} \to (x^1)^{-1},\ \Gamma^{3}{}_{23} \to \mathrm{Cot}[x^2]\right\}$$

$\Gamma^{0}{}_{01} \to \frac{A'[r]}{2\,A[r]}$

$\Gamma^{1}{}_{00} \to \frac{A'[r]}{2\,B[r]}$

$\Gamma^{1}{}_{11} \to \frac{B'[r]}{2\,B[r]}$

$\Gamma^{1}{}_{22} \to -\frac{r}{B[r]}$

$\Gamma^{1}{}_{33} \to -\frac{r\,\mathrm{Sin}[\theta]^2}{B[r]}$

$\Gamma^{2}{}_{12} \to \frac{1}{r}$

$\Gamma^{2}{}_{33} \to -\mathrm{Cos}[\theta]\,\mathrm{Sin}[\theta]$

$\Gamma^{3}{}_{13} \to \frac{1}{r}$

$\Gamma^{3}{}_{23} \to \mathrm{Cot}[\theta]$

Calculate the covariant Riemann tensor in preparation for calculating the Ricci tensor.

$$\left\{\{\{\{0,0,0,0\},\{0,0,0,0\},\{0,0,0,0\},\{0,0,0,0\}\},\ \ll 1 \gg,\right.$$
$$\left\{\left\{0,0,-\frac{x^1\,A'[x^1]}{2\,B[x^1]},0\right\},\{0,0,0,0\},\left\{\frac{x^1\,A'[x^1]}{2\,B[x^1]},0,0,0\right\},\{0,0,0,0\}\right\},$$
$$\left\{\left\{0,0,0,-\frac{\mathrm{Sin}[x^2]^2\,x^1\,A'[x^1]}{2\,B[x^1]}\right\},\{0,0,0,0\},\{0,0,0,0\},\left\{\frac{\mathrm{Sin}[x^2]^2\,x^1\,A'[x^1]}{2\,B[x^1]},0,0,0\right\}\right\}\},$$
$$\left.\{\ll 1 \gg\},\ \{\{\ll 1 \gg\},\{\ll 1 \gg\},\{\ll 1 \gg\},\{\ll 1 \gg\}\},\ \ll 1 \gg\right\}$$

Calculate the Ricci tensor $R_{\mu\nu}$.

$$\begin{pmatrix} \frac{A'[x^1]}{B[x^1]\,x^1} - \frac{\frac{A'[x^1]^2}{A[x^1]} + \frac{A'[x^1]\,B'[x^1]}{B[x^1]} - 2\,A''[x^1]}{4\,B[x^1]} & 0 & 0 & 0 \\ 0 & \frac{B'[x^1]}{B[x^1]\,x^1} + \frac{\frac{A'[x^1]^2}{A[x^1]} + \frac{A'[x^1]\,B'[x^1]}{B[x^1]} - 2\,A''[x^1]}{4\,A[x^1]} & 0 & 0 \\ 0 & 0 & \frac{-1+B[x^1]}{B[x^1]} - \frac{x^1\,A'[x^1]}{2\,A[x^1]\,B[x^1]} + \frac{x^1\,B'[x^1]}{2\,B[x^1]^2} & 0 \\ 0 & 0 & 0 & \frac{(-1+B[x^1]}{B[} \end{pmatrix}$$

Only the diagonal terms of $R_{\mu\nu}$ are nonzero. We set $R_{\mu\nu} = 0$ (empty space), use coordinate symbols, extract the diagonal terms as equations and solve them for `A[r]` and `B[r]`.

```
{eqn[3, 54], eqn[3, 55], eqn[3, 56], eqn[3, 57]};
(Tr[Simplify[ricci // UseCoordinates[varnames]], List] == {0, 0, 0, 0} // Thread //
    Simplify);
{%%, %};
Transpose[%] // TableForm
MapThread[Set, %%];
Print["Note: eqn[3,57] is the same as Sin(θ)eqn[3,56]. No new info!"]
Sin[θ] (eqn[3, 56][[1]] - eqn[3, 56][[2]]) == (eqn[3, 57][[1]] - eqn[3, 57][[2]])
Print["If we take eqn[3,54]-eqn[3,55] we obtain..."]
Inner[Subtract, eqn[3, 54], eqn[3, 55], Equal] // Simplify;
# B[r] r & /@ %
%[[1]] == HoldForm[∂r (A[r] B[r])]
% // ReleaseHold
Print["We get, taking into account the asymptotic values of A and B..."]
A[r] B[r] == const
A[r] B[r] == 1
Brule = LHSSymbolsToPatterns[{r}]@Solve[%, B[r]][[1,1]]
Print["Eliminate B'[r] and A''[r] from eqn[3,54-55-56], substitute for
    B[r] from above and solve the ODE (RS is an integration constant)..."]
Eliminate[{eqn[3, 54], eqn[3, 55], eqn[3, 56]}, {B'[r], A''[r]}]
First[%] /. Brule // Simplify
Arule = LHSSymbolsToPatterns[{r}]@DSolve[{%}, A[r], r][[1,1]] /. C[1] → -RS
Print["Check the solutions A[r] and B[r] in eqn[3,54-55-56]:"]
Arule /. Rule → Set; Brule /. Rule → Set;
{eqn[3, 54], eqn[3, 55], eqn[3, 56]} // Simplify
metric /. Brule /. Arule;
SetMetricValueRules[g, %]
Print["Schwarzschild line element with parameter RS:"]
ds^2 == gdd[μ, ν] dxu[μ] dxu[ν] // ToArrayValues[] // useSchwarzschild // TraditionalForm
```

eqn[3, 54]	$A'[r]\left(-\frac{4}{r}+\frac{A'[r]}{A[r]}+\frac{B'[r]}{B[r]}\right) == 2\,A''[r]$
eqn[3, 55]	$\frac{A'[r]^2}{A[r]}+\frac{4\,A[r]\,B'[r]}{r\,B[r]}+\frac{A'[r]\,B'[r]}{B[r]} == 2\,A''[r]$
eqn[3, 56]	$2\,B[r]+\frac{r\,B'[r]}{B[r]} == 2+\frac{r\,A'[r]}{A[r]}$
eqn[3, 57]	$\mathrm{Sin}[\theta]\left(-2+2\,B[r]-\frac{r\,A'[r]}{A[r]}+\frac{r\,B'[r]}{B[r]}\right) == 0$

Note: eqn[3,57] is the same as Sin(θ)eqn[3,56]. No new info!

True

If we take eqn[3,54]-eqn[3,55] we obtain...

$B[r]\,A'[r]+A[r]\,B'[r] == 0$

$B[r]\,A'[r]+A[r]\,B'[r] == \partial_r\,(A[r]\,B[r])$

True

We get, taking into account the asymptotic values of A and B...

$A[r]\,B[r] == \mathrm{const}$

$A[r]\,B[r] == 1$

$B[r\_] \to \frac{1}{A[r]}$

Eliminate B'[r] and A''[r] from eqn[3,54-55-56], substitute
for B[r] from above and solve the ODE (R_S is an integration constant)...

$$A'[r] == \frac{A[r]\,(-1+B[r])}{r} \;\&\&\; r \neq 0 \;\&\&\; A[r] \neq 0 \;\&\&\; B[r] \neq 0$$

$$A'[r] == \frac{1 - A[r]}{r}$$

$$A[r\_] \to 1 - \frac{R_S}{r}$$

Check the solutions A[r] and B[r] in eqn[3,54-55-56]:

{True, True, True}

Schwarzschild line element with parameter R_S:

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

The last task is to determine the value of R_S. This is done comparing h_{00} to $\frac{2\,V[r]}{c^2} = -\frac{2\,G\,M}{c^2\,r}$ in the approximation for small $h_{\nu\mu} = g_{\nu\mu} - \eta_{\nu\mu}$ and small M. (See sec. 2.7 and 2.8 for more details.)

```
Print["gμν in Schwarzschild coordinates:"]
gdd[μ, ν] // ToArrayValues[]
Print["some guessed ημν in Schwarzschild coordinates:"]
Eta = DiagonalMatrix[{1, -1, -r^2, -r^2 Sin[θ]^2}];
SetMetricValueRules[η, % // CoordinatesToTensors[varnames]]
ηdd[μ, ν] // ToArrayValues[]
Print["hμν=gμν-ημν in Schwarzschild coordinates:"]
H = ToArrayValues[][gdd[μ, ν] - ηdd[μ, ν]] // Simplify
(
 Unprotect[D];
 D[fun_, c t] := c^-1 D[fun, t];
 Protect[D];
)
"transformation from Schwarzschild coordinates to new coordinates" ==
 {c t, r Sin[θ] Cos[ϕ], r Sin[θ] Sin[ϕ], r Cos[θ]}
Λ = Outer[D, %[[2]], varnames] // Simplify;
InvΛ = Inverse[Λ] // Simplify;
Print["hμν in new coordinates:"]
Transpose[InvΛ].H.InvΛ // Simplify;
SetTensorValues[hdd[μ, ν], %]
%% // UseCoordinates[varnames] // MatrixForm
Print["hμν → 0 for large r"]
Limit[%%, r → Infinity]
Print["ημν in new coordinates:"]
Transpose[InvΛ].Eta.InvΛ // Simplify
Print["This implies that the new coordinates are nearly Cartesian for large r.\n",
 "Small M means that r behaves as radial distance.\n",
"So we are allowed to identify h00 as the Newtonian gravitational potential:"]
(hdd[0, 0] // UseCoordinates[varnames]) == 2 V[r] / c^2 /. V[r] → -G M / r
Print["Solve for RS"]
RSrule = Solve[%%, RS][[1,1]]
```

$g_{\mu\nu}$ in Schwarzschild coordinates:

$$\left\{\left\{1-\frac{R_S}{x^1}, 0, 0, 0\right\}, \left\{0, -\frac{1}{1-\frac{R_S}{x^1}}, 0, 0\right\}, \left\{0, 0, -\left(x^1\right)^2, 0\right\}, \left\{0, 0, 0, -\mathrm{Sin}\left[x^2\right]^2 \left(x^1\right)^2\right\}\right\}$$

some guessed $\eta_{\mu\nu}$ in Schwarzschild coordinates:

$$\left\{\{1, 0, 0, 0\}, \{0, -1, 0, 0\}, \left\{0, 0, -\left(x^1\right)^2, 0\right\}, \left\{0, 0, 0, -\mathrm{Sin}\left[x^2\right]^2 \left(x^1\right)^2\right\}\right\}$$

$h_{\mu\nu}=g_{\mu\nu}-\eta_{\mu\nu}$ in Schwarzschild coordinates:

$$\left\{\left\{-\frac{R_S}{x^1}, 0, 0, 0\right\}, \left\{0, \frac{R_S}{R_S - x^1}, 0, 0\right\}, \{0, 0, 0, 0\}, \{0, 0, 0, 0\}\right\}$$

transformation from Schwarzschild coordinates to new coordinates ==
$\{c\,t, r\,\mathrm{Cos}[\phi]\,\mathrm{Sin}[\theta], r\,\mathrm{Sin}[\theta]\,\mathrm{Sin}[\phi], r\,\mathrm{Cos}[\theta]\}$

$h_{\mu\nu}$ in new coordinates:

$$\begin{pmatrix} -\frac{R_S}{r} & 0 & 0 & 0 \\ 0 & \frac{\mathrm{Cos}[\phi]^2\,\mathrm{Sin}[\theta]^2\,R_S}{-r+R_S} & \frac{\mathrm{Cos}[\phi]\,\mathrm{Sin}[\theta]^2\,\mathrm{Sin}[\phi]\,R_S}{-r+R_S} & \frac{\mathrm{Cos}[\theta]\,\mathrm{Cos}[\phi]\,\mathrm{Sin}[\theta]\,R_S}{-r+R_S} \\ 0 & \frac{\mathrm{Cos}[\phi]\,\mathrm{Sin}[\theta]^2\,\mathrm{Sin}[\phi]\,R_S}{-r+R_S} & \frac{\mathrm{Sin}[\theta]^2\,\mathrm{Sin}[\phi]^2\,R_S}{-r+R_S} & \frac{\mathrm{Cos}[\theta]\,\mathrm{Sin}[\theta]\,\mathrm{Sin}[\phi]\,R_S}{-r+R_S} \\ 0 & \frac{\mathrm{Cos}[\theta]\,\mathrm{Cos}[\phi]\,\mathrm{Sin}[\theta]\,R_S}{-r+R_S} & \frac{\mathrm{Cos}[\theta]\,\mathrm{Sin}[\theta]\,\mathrm{Sin}[\phi]\,R_S}{-r+R_S} & \frac{\mathrm{Cos}[\theta]^2\,R_S}{-r+R_S} \end{pmatrix}$$

$h_{\mu\nu} \to 0$ for large r

$$\{\{0, 0, 0, 0\}, \{0, 0, 0, 0\}, \{0, 0, 0, 0\}, \{0, 0, 0, 0\}\}$$

$\eta_{\mu\nu}$ in new coordinates:

$$\{\{1, 0, 0, 0\}, \{0, -1, 0, 0\}, \{0, 0, -1, 0\}, \{0, 0, 0, -1\}\}$$

This implies that the new coordinates are nearly Cartesian for large r.
Small M means that r behaves as radial distance.
So we are allowed to identify h_{00} as the Newtonian gravitational potential:

$$-\frac{R_S}{r} == -\frac{2\,G\,M}{c^2\,r}$$

Solve for R_S

$$R_S \to \frac{2\,G\,M}{c^2}$$

Finally...

```
gdd[μ, ν] // ToArrayValues[];
% /. RSrule;
SetMetricValueRules[g, %];
gdd[μ, ν] == (%% // UseCoordinates[varnames] // MatrixForm)
(eqn[3, 59] = ds² == gdd[μ, ν] dxu[μ] dxu[ν] // ToArrayValues[] // useSchwarzschild) //
    TraditionalForm // FrameBox // DisplayForm
```

$$g_{\mu\nu} == \begin{pmatrix} 1-\frac{2\,G\,M}{c^2\,r} & 0 & 0 & 0 \\ 0 & -\frac{1}{1-\frac{2\,G\,M}{c^2\,r}} & 0 & 0 \\ 0 & 0 & -r^2 & 0 \\ 0 & 0 & 0 & -r^2\,\mathrm{Sin}[\theta]^2 \end{pmatrix}$$

$$\boxed{ds^2 = -\frac{(dr)^2}{1-\frac{2\,G\,M}{c^2\,r}} + c^2\left(1-\frac{2\,G\,M}{c^2\,r}\right)(dt)^2 - r^2\,(d\theta)^2 - r^2\,(d\phi)^2\,\sin^2(\theta)}$$

This is the standard form of the Schwarzschild metric (→ *Schwarzschild geometry*). $\frac{2\,G\,M}{c^2} = R_S$ is the Schwarzschild radius of the mass M.

```
(* The Schwarzschild radius of some objects. *)
RS[M_(*kg*)](*m*) = RSrule[[2]] /. {G → 6.674×10^-11, c → 299 792 458};
data = {
   {"1/2 solar mass neutron star", mSun/2, 10^4},
   {"white dwarf (~Sirius B)", mSun, 5.750 10^6},
   {"Sun", mSun = 1.989 10^30 (*kg*), 6.960 10^8 (*m*)},
   {"Earth", 5.976 10^24, 6.3710 10^6}, {"me (october 2011)", mme = 77, (3 mme 10^-3 / (4 π))^(1/3)},
   {"proton", 1.67×10^-27, Sqrt[0.74] 10^-15}};
RS[#] & /@ (#[[2]] & /@ data);
RS[#] & /@ (#[[2]] & /@ data);
% / (#[[3]] & /@ data);
Prepend[Transpose[{(#[[1]] & /@ data), %%, %}],
  {"Object", "RS (m)", "RS/rsurface"}] // TableForm
```

Object	R_S (m)	$R_S/r_{surface}$
$\frac{1}{2}$ solar mass neutron star	1477.	0.1477
white dwarf (~Sirius B)	2953.99	0.000513738
Sun	2953.99	4.24424×10^{-6}
Earth	0.00887535	1.39309×10^{-9}
me (october 2011)	1.14358×10^{-25}	4.33309×10^{-25}
proton	2.48023×10^{-54}	2.8832×10^{-39}

Chapter 4: Physics in the vicinity of a massive object

4.0 Introduction p. 123

FN: "Turning *M* up introduces curvature, so that spacetime is no longer flat, and there is no reason to assume that the coordinates have the simple physical meanings they had in flat spacetime."

Ranges of the Schwarzschild coordinates :

$$t \in (-\infty, +\infty)$$
$$\theta \in [0, \pi],\ \phi \in [0, 2\pi) \text{ with } \phi + \text{integer } 2\pi \equiv \phi$$
$$x(r_B, R_S), +\infty) \text{ where } r_B \text{ is the boundary of the object}$$

In the next subsections we investigate the relationship between coordinates and physically observable quantities in the framework of the static spacetime given by the Schwarzschild solution.

4.1 Length and time p. 124

"Chuck Norris doesn't wear a watch, HE decides what time it is."

```
Needs["TensorCalculus3`Tensorial`"]
$PrePrint =.
labs = {x, δ, g, Γ};
DeclareBaseIndices[{0, 1, 2, 3}]
```

```
DefineTensorShortcuts[
 {{x, dx}, 1},
 {{δ, g}, 2},
 {{Γ}, 3}]
SetAttributes[c, Constant]
varnames = {c t, r, θ, ϕ};
dvarnames = Dt[%];
useSchwarzschild = Composition @@ MapThread[UseCoordinates[#1, #2] &, {{%, %%}, {dx, x}}];
box = Composition[DisplayForm, FrameBox, TraditionalForm];
(* Def. of Schwarzschild metric *)
DiagonalMatrix[{1 - 2 G M / (c^2 r), - (1 - 2 G M / (c^2 r))^-1, - r^2, - r^2 Sin[θ]^2}] /. (M → m c^2 / G);
SetMetricValueRules[g, % // CoordinatesToTensors[varnames]];
```

---

1) We rewrite the spacetime line element in the Schwarzschild metric as function of observable quantities.

```
Print["Spacetime line element in Schwarzschild metric eq.(4.1),p.124"]
ds^2 == c^2 dτ^2 == gdd[μ, ν] dxu[μ] dxu[ν]
(eqn[4, 1] = %[[{1, 3}]] // ToArrayValues[] // useSchwarzschild) // box
Print["- Space line element at fixed time (sign change!) eq.(4.2),p.124"]
dl^2 == (-gdd[μ, ν] dxu[μ] dxu[ν] // ToArrayValues[{1, 2, 3}] // useSchwarzschild) ==
 HoldForm[dR^2 + dL^2]
Print["-- tangential line element eq.(4.4),p.125"]
dL^2 == -gdd[μ, ν] dxu[μ] dxu[ν] // ToArrayValues[{2, 3}] // Simplify // useSchwarzschild
(eqn[4, 4] = Sqrt /@ % // PowerExpand) // box
Print["-- radial line element eq.(4.5),p.125"]
dR^2 == -gdd[μ, ν] dxu[μ] dxu[ν] // ToArrayValues[{1}] // useSchwarzschild
%[[1]] == HoldForm[Evaluate[Series[%[[2]] / Dt[r]^2, {r, ∞, 3}]]] HoldForm[Dt[r]^2] //
 TraditionalForm
(eqn[4, 5] = Sqrt /@ %% // PowerExpand) // box
Print["- Proper time line element at fixed point in space eq.(4.6),p.127"]
c^2 dT^2 == gdd[μ, ν] dxu[μ] dxu[ν] // ToArrayValues[{0}] // useSchwarzschild
%[[1]] == HoldForm[Evaluate[Series[%[[2]] / Dt[t]^2, {r, ∞, 3}]]] HoldForm[Dt[t]^2] //
 TraditionalForm
(eqn[4, 6] = Sqrt[# / c^2] & /@ %% // PowerExpand) // box
Print["Hence the spacetime line element in the Schwarzschild metric is:"]
ds^2 == HoldForm[c^2 dT^2 - (dR^2 + dL^2)]
% /. ({eqn[4, 1], eqn[4, 4], eqn[4, 5], eqn[4, 6]} /. Equal → Rule) // ReleaseHold // Simplify
```

Spacetime line element in Schwarzschild metric eq.(4.1),p.124

$$ds^2 == c^2 d\tau^2 == dx^\mu dx^\nu g_{\mu\nu}$$

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

- Space line element at fixed time (sign change!) eq.(4.2),p.124

$$dl^2 == \frac{Dt[r]^2}{1-\frac{2m}{r}} + r^2 Dt[\theta]^2 + r^2 Dt[\phi]^2 Sin[\theta]^2 == dR^2 + dL^2$$

-- tangential line element eq.(4.4),p.125

$$dL^2 == r^2 \left(Dt[\theta]^2 + Dt[\phi]^2 Sin[\theta]^2\right)$$

$$dL = r\sqrt{(d\theta)^2 + (d\phi)^2 \sin^2(\theta)}$$

-- radial line element eq.(4.5),p.125

$$dR^2 == \frac{Dt[r]^2}{1 - \frac{2m}{r}}$$

$$dR^2 = (dr)^2\left(1 + \frac{2m}{r} + 4m^2\left(\frac{1}{r}\right)^2 + 8m^3\left(\frac{1}{r}\right)^3 + O\left(\left(\frac{1}{r}\right)^4\right)\right)$$

$$dR = \frac{dr}{\sqrt{1 - \frac{2m}{r}}}$$

- Proper time line element at fixed point in space eq.(4.6),p.127

$$c^2\, dT^2 == c^2\left(1 - \frac{2m}{r}\right) Dt[t]^2$$

$$c^2\, dT^2 = (dt)^2\left(c^2 - \frac{2(c^2 m)}{r} + O\left(\left(\frac{1}{r}\right)^4\right)\right)$$

$$dT = \sqrt{1 - \frac{2m}{r}}\, dt$$

Hence the spacetime line element in the Schwarzschild metric is:

$$ds^2 == c^2\, dT^2 - \left(dR^2 + dL^2\right)$$

True

Coordinate distance of concentric spheres: surface S(r) < S(r+Δr), ΔS = S(r+Δr)-S(r). Δr≠ΔR !

```
Reduce[{ΔS == 4 π (r + Δr)^2 - 4 π r^2 && Δr > 0 && r > 0 && ΔS > 0}, Δr][[3]]
```

$$\Delta r == -r + \frac{\sqrt{4\pi r^2 + \Delta S}}{2\sqrt{\pi}}$$

Coordinate distance of concentric circles: circumference C(r) < C(r+Δr), ΔC = C(r+Δr)-C(r). Δr≠ΔR ! We can infer coordinate distances Δr from circumference differences ΔC.

```
Reduce[{ΔC == 2 π (r + Δr) - 2 π r}, Δr]
```

$$\Delta r == \frac{\Delta C}{2\pi}$$

■ Examples 4.1.2 p. 128; Exercise 4.1 p. 129.

(a) **Short stick** (differential approximation $\Delta R \approx (1 - 2m/r)^{-1/2}\,\Delta r$). If a stick of length 1 m lies radially in the field of a star where m/r is 10^{-2}, what *coordinate distance* does it take up?

```
eqn[4, 5]
Solve[% /. {m / r → 10^-2, dR → 1}] // N
res = %[[1, 1, 2]];
```

$$dR == \frac{Dt[r]}{\sqrt{1-\frac{2m}{r}}}$$

```
{{Dt[r] → 0.989949}}
```

(b) **Long stick** (exact integral for ΔR). A long stick is lying radially in the field of a spherical object of Schwarzschild radius 2m. If the r coordinates of its ends are at r1 and r2 (2 m < r1 < r2), what is its *length* ΔR?

```
$Assumptions = {0 < m, 0 < r1, r1 > 2 m, r1 < r2, r1 < r};
HoldForm[∫_r1^r2 1 / √(1 - 2 m / r) ⅆr];
Print["ΔR = ", %, " = "]
ReleaseHold[%%]
```

$$\Delta R \ = \int_{r1}^{r2} \frac{1}{\sqrt{1-\frac{2m}{r}}} \, dr \ =$$

$$-\sqrt{r1\,(-2m+r1)}+\sqrt{r2\,(-2m+r2)}-2\,m\,\mathrm{ArcTanh}\left[\sqrt{\frac{r1}{-2m+r1}}\right]+2\,m\,\mathrm{ArcTanh}\left[\sqrt{\frac{r2}{-2m+r2}}\right]$$

We must apply a rather tricky transformation to obtain the result in the same form as FN. Different *Mathematica* versions evaluate the integral to different forms, so the following code is system-dependent. The final result should be $\Delta R = -\sqrt{r1\,(-2m+r1)}+\sqrt{r2\,(-2m+r2)}+2\,m\,\mathrm{Log}\left[\left(\sqrt{r2}+\sqrt{-2m+r2}\right)\big/\left(\sqrt{r1}+\sqrt{-2m+r1}\right)\right]$.

```
% // TrigToExp;
expr = % //. {m Log[a_] - m Log[b_] → m Log[a / b], m Log[a_] + m Log[b_] → m Log[a b]}
(√r1 + √(-2 m + r1)) (√r2 + √(-2 m + r2)) √((-2 m + r1) (-2 m + r2) / m) ;
Numerator[expr[[3, 2, 1]]] % // FullSimplify;
%[[4]] Simplify[%[[{1, 2, 3}]]];
Denominator[expr[[3, 2, 1]]] %%% // FullSimplify;
%[[2]] Simplify[%[[{1, 3, 4}]]];
ΔR = expr /. Log[_] → Log[%%% / %] // FullSimplify
Print["With m = G M/c^2 we get eq.(4.7),p.128:"]
%% /. m → G M / c^2
Print["Flat space limit for M→0"]
Limit[%%, M → 0]
```

$$-\sqrt{r1\,(-2m+r1)}+\sqrt{r2\,(-2m+r2)}+m\,\mathrm{Log}\left[\frac{\left(1-\sqrt{\frac{r1}{-2m+r1}}\right)\left(1+\sqrt{\frac{r2}{-2m+r2}}\right)}{\left(1+\sqrt{\frac{r1}{-2m+r1}}\right)\left(1-\sqrt{\frac{r2}{-2m+r2}}\right)}\right]$$

$$-\sqrt{r1\,(-2m+r1)}+\sqrt{r2\,(-2m+r2)}+2\,m\,\mathrm{Log}\left[\frac{\sqrt{r2}+\sqrt{-2m+r2}}{\sqrt{r1}+\sqrt{-2m+r1}}\right]$$

```
With m = G M/c^2 we get eq.(4.7),p.128:
```

$$-\sqrt{r1\left(-\frac{2\,G\,M}{c^2}+r1\right)}+\sqrt{r2\left(-\frac{2\,G\,M}{c^2}+r2\right)}+\frac{2\,G\,M\,\mathrm{Log}\left[\frac{\sqrt{r2}+\sqrt{-\frac{2GM}{c^2}+r2}}{\sqrt{r1}+\sqrt{-\frac{2GM}{c^2}+r1}}\right]}{c^2}$$

```
Flat space limit for M→0
```

$-r1+r2$

Check differential approximation in example 4.1.2(a) with the exact formula for ΔR calculated in example 4.1.2(b):

```
ΔR /. {m → 10^-2, r1 → 1 - res / 2, r2 → 1 + res / 2}
1 + {-res, +res} / 2
FindRoot[(ΔR /. {m → 10^-2, r1 → %[[1]]}) == 1, {r2, 1}]
```

```
1.001

{0.505025, 1.49497}

{r2 → 1.49398}
```

Extra space *ES* between the orbits of Earth and Venus in the gravitational field of the Sun. (**Extra Raum** *ES* zwischen den Bahnen von Erde und Venus im Sonnenschwerefeld.):

```
data = {m → G M / c^2, G → 6.674*^-11, c → 299 792 458,
   M → (*M_Sun=*) 1.989*^30, r1 → (*r_Venus=*) 108.2*^9, r2 → (*r_Earth=*) 149.6*^9};
ES = (ΔR - (r2 - r1)) Meter //. data
ES / (ΔR Meter) //. data
ES / (c Meter / Second) /. data
```

```
478.523 Meter

1.15585 × 10^-8

1.59618 × 10^-6 Second
```

4.2 Radar sounding (*Shapiro-Effekt*) p. 129

```
Needs["TensorCalculus3`Tensorial`"]
$PrePrint =.
labs = {x, δ, g, Γ};
DeclareBaseIndices[{0, 1, 2, 3}]
DefineTensorShortcuts[
 {{x, dx}, 1},
 {{δ, g}, 2},
 {{Γ}, 3}]
SetAttributes[c, Constant]
varnames = {c t, r, θ, ϕ};
dvarnames = Dt[%];
useSchwarzschild = Composition[UseCoordinates[%, dx], UseCoordinates[%%, x]];
(* Def. of Schwarzschild metric *)
DiagonalMatrix[{1 - 2 G M / (c^2 r), - (1 - 2 G M / (c^2 r))^-1, -r^2, -r^2 Sin[θ]^2}] /. (M → m c^2 / G);
SetMetricValueRules[g, % // CoordinatesToTensors[varnames]];
```

When Venus and Earth are line up with the sun a radar pulse is bounced off Venus from Earth. The proper time lapse $\Delta\tau$ for the whole trip as measured by the observer on Earth is compared with the expected value $\Delta\tilde{\tau}$ from classical theory. (Irwin I. Shapiro, December 1964, "Fourth Test of General Relativity", *Physical Review Letters* **13**: 789–791; first detection in 1968.)

A) **General relativity:** two-way travel time $\Delta\tau$

```
Print["Spacetime line element in Schwarzschild metric"]
Dt[s]^2 == c^2 Dt[τ]^2 == gdd[μ, ν] dxu[μ] dxu[ν]
(% // ToArrayValues[] // useSchwarzschild) // Rest
Print[
 "For fixed θ and ϕ (inferior conjunction) and using dτ == 0 for a radar pulse we get"]
(%% /. {Dt[τ] → 0, Dt[θ] → 0, Dt[ϕ] → 0})
Print[
 "Solve for the coordinate speed of light in the radial direction c[r](Note: 2m < r)"]
Map[# / Dt[t] &, Solve[%%, Dt[r]] // Flatten, {2}]
(c[r] = %[[1,2]]) == -%[[2,2]] // Simplify
Print["The two-way travel time from Earth (r1) to
   Venus (r2) and back again as measured by the t coordinate is..."]
Δt == HoldForm[∫_r1^r2 -c[r]^-1 ⅆr + ∫_r2^r1 c[r]^-1 ⅆr]
Print[
 "The observer on Earth will measure the elapsed proper time Δτ by his clock at r1 so
   we have to use time warping (eq.(4.6),p.127) and get finally eq.(4.9),p.130..."]
Δτ == Sqrt[1 - 2 m / r1] Δt
% /. Δt → %%%[[2]]
eqn[4, 9] = (Assuming[0 < 2 m < r2 < r1, % // ReleaseHold] // Simplify) /.
   (2 m Log[a_] - 2 m Log[b_] → 2 m Log[a / b])
Print["...and to first order in m (eq.(4.10a),p.130):"]
eqn[4, 10 a] = Series[eqn[4, 9][[2]], {m, 0, 1}] // Normal;
Δτ ≈ HoldForm[2 / c] Collect[Simplify[% / (2 / c)], m]
```

Spacetime line element in Schwarzschild metric

$$Dt[s]^2 == c^2\, Dt[\tau]^2 == dx^{\mu}\, dx^{\nu}\, g_{\mu\nu}$$

$$c^2\, Dt[\tau]^2 == -\frac{Dt[r]^2}{1-\frac{2m}{r}} + c^2\left(1-\frac{2m}{r}\right) Dt[t]^2 - r^2\, Dt[\theta]^2 - r^2\, Dt[\phi]^2\, Sin[\theta]^2$$

For fixed θ and ϕ (inferior conjunction) and using dτ == 0 for a radar pulse we get

$$0 == -\frac{Dt[r]^2}{1-\frac{2m}{r}} + c^2\left(1-\frac{2m}{r}\right) Dt[t]^2$$

Solve for the coordinate speed of light in the radial direction c[r](Note: 2m < r)

$$\left\{\frac{Dt[r]}{Dt[t]} \to c\left(1-\frac{2m}{r}\right),\ \frac{Dt[r]}{Dt[t]} \to \frac{c\,(2m-r)}{r}\right\}$$

True

The two-way travel time from Earth (r1) to
Venus (r2) and back again as measured by the t coordinate is...

$$\Delta t == \int_{r1}^{r2} -\frac{1}{c[r]}\, dr + \int_{r2}^{r1} \frac{1}{c[r]}\, dr$$

The observer on Earth will measure the elapsed proper time Δτ by his clock at r1
so we have to use time warping (eq.(4.6),p.127) and get finally eq.(4.9),p.130...

$$\Delta\tau == \sqrt{1-\frac{2m}{r1}}\ \Delta t$$

$$\Delta\tau == \sqrt{1-\frac{2\,m}{r1}}\left(\int_{r1}^{r2}-\frac{1}{c[r]}\,dr+\int_{r2}^{r1}\frac{1}{c[r]}\,dr\right)$$

$$\Delta\tau == \frac{2\sqrt{1-\frac{2\,m}{r1}}\left(r1-r2+2\,m\,\mathrm{Log}\left[\frac{-2\,m+r1}{-2\,m+r2}\right]\right)}{c}$$

...and to first order in m (eq.(4.10a),p.130):

$$\Delta\tau \approx \frac{2}{c}\left(r1-r2+m\left(-1+\frac{r2}{r1}+2\,\mathrm{Log}\left[\frac{r1}{r2}\right]\right)\right)$$

B) **Classical theory:** two-way travel time $\Delta\tilde{\tau}$

I depart here from FN. In the *flat* spacetime of the classical theory there is no warping, hence r1-r2 is the true distances between Earth and Venus as given by Euclidean geometry and t measures the absolute time.

```
Print["Round-trip time Δτ̃"]
Δ τ̃ == (eqn[4, 10 b] = 2 (r1 - r2) / c)
```

Round-trip time $\Delta\tilde{\tau}$

$$\Delta\tilde{\tau} == \frac{2\,(r1-r2)}{c}$$

C) Hence the GR-induced delay $\Delta\tau - \Delta\tilde{\tau}$ is to first order in m:

```
2 m / c;
Δτ - Δ τ̃ ≈ % Simplify[Expand[eqn[4, 10 a] - eqn[4, 10 b]] / %]
Δτ - Δ τ̃ ≈ (eqn[4, 11] = %[[2]] /. m → G M / c^2) // TraditionalForm // FrameBox // DisplayForm
```

$$\Delta\tau - \Delta\tilde{\tau} \approx \frac{2\,m\left(-1+\frac{r2}{r1}+2\,\mathrm{Log}\left[\frac{r1}{r2}\right]\right)}{c}$$

$$\boxed{\Delta\tau - \Delta\tilde{\tau} \approx \frac{2\,G\,M\left(\frac{r2}{r1}+2\log\left(\frac{r1}{r2}\right)-1\right)}{c^3}}$$

Let's calculate how large an effect this would be for a measurement with Earth and Venus in inferior conjunction.

```
data =
  {m → G M / c^2, G → 6.674*^-11, c → 299 792 458, M → 1.989*^30, r1 → 149.6*^9, r2 → 108.2*^9};
Print["GR pulse time delay Δτ from eq.(4.10a)"]
NumberForm[eqn[4, 10 a] Second, 10] //. data
% / (60 Second / Minute)
Print["Classical pulse time delay Δτ̃"]
NumberForm[eqn[4, 10 b] Second, 10] //. data
% / (60 Second / Minute)
Print["Δτ-Δτ̃ and c(Δτ-Δτ̃)"]
eqn[4, 11] Second /. data
% c Meter / Second //. data
Print["An estimate of the required measurement precision on Δτ is..."]
(Δτ - Δ τ̃)/Δτ == eqn[4, 11]/eqn[4, 10 a] //. data
```

GR pulse time delay Δτ from eq.(4.10a)

276.1910745 Second

4.60318 Minute

Classical pulse time delay $\Delta\tilde{\tau}$

276.1910708 Second

4.60318 Minute

$\Delta\tau-\Delta\tilde{\tau}$ and $c(\Delta\tau-\Delta\tilde{\tau})$

3.6579×10^{-6} Second

1096.61 Meter

An estimate of the required measurement precision on $\Delta\tau$ is...

$$\frac{\Delta\tau-\Delta\tilde{\tau}}{\Delta\tau} == 1.32441\times10^{-8}$$

4.3 Spectral Shift p. 131

```
Needs["TensorCalculus3`Tensorial`"]
$PrePrint =.
labs = {x, δ, g, Γ};
DeclareBaseIndices[{0, 1, 2, 3}]
DefineTensorShortcuts[
 {{x, dx}, 1},
 {{δ, g}, 2},
 {{Γ}, 3}]
SetAttributes[c, Constant]
```

- **1) General case:** Suppose that in a *static* spacetime a signal is sent from an emitter **E** at a fixed point, that it travels along a null geodesic and is received by a receiver **R** at a fixed point.

```
Print["Line element in spacetime:"]
ds^2 == gdd[μ, ν] dxu[μ] dxu[ν]
Print["Line element along a null geodesic:"]
%% /. ds → 0
Print["Null geodesic with affine parametrization x^μ(u), u ∈ [u_E,u_R]. Divide by du^2."]
%% /. dxu[i_] :→ TotalD[xu[i], u]
Print["Expand into temporal and spatial parts."]
MapAt[PartialSum[0, {i, j}], %%, 2]
Print["In a suitably defined coordinate system the mixed metric components
   g_i0 of a static spacetime are zero (Landau,vol.II,chap.10,par.88)."]
%% /. {gdd[i, 0] → 0, gdd[0, j] → 0}
Print["Solve for the coordinate time component.(⚠: √(a^2))"]
Reverse[%%];
# - Part[%, 1, 2] & /@ %
# / gdd[0, 0] & /@ %
√# & /@ %
MapAt[PowerExpand, %, 1]
Print["Substitute variable x^0(u)=c t(u) and simplify."]
%% /. Tensor[x, List[0], List[Void]] → c t[u]
# / c & /@ %
Print["Integrate. We get the coordinate time of travel:"]
Integrate[#, {u, u"E", uR}] & /@ %%
```

Line element in spacetime:

$$ds^2 == dx^{\mu}\, dx^{\nu}\, g_{\mu\nu}$$

Line element along a null geodesic:

$$0 == dx^{\mu}\, dx^{\nu}\, g_{\mu\nu}$$

Null geodesic with affine parametrization $x^{\mu}(u)$, $u \in [u_E, u_R]$. Divide by du^2.

$$0 == g_{\mu\nu} \frac{dx^{\mu}}{du} \frac{dx^{\nu}}{du}$$

Expand into temporal and spatial parts.

$$0 == g_{00} \left(\frac{dx^0}{du}\right)^2 + g_{i0} \frac{dx^0}{du} \frac{dx^i}{du} + g_{0j} \frac{dx^0}{du} \frac{dx^j}{du} + g_{ij} \frac{dx^i}{du} \frac{dx^j}{du}$$

In a suitably defined coordinate system the mixed metric
components g_{i0} of a static spacetime are zero (Landau,vol.II,chap.10,par.88).

$$0 == g_{00} \left(\frac{dx^0}{du}\right)^2 + g_{ij} \frac{dx^i}{du} \frac{dx^j}{du}$$

Solve for the coordinate time component.(⚠: $\sqrt{a^2}$)

$$g_{00} \left(\frac{dx^0}{du}\right)^2 == -g_{ij} \frac{dx^i}{du} \frac{dx^j}{du}$$

$$\left(\frac{dx^0}{du}\right)^2 == -\frac{g_{ij} \frac{dx^i}{du} \frac{dx^j}{du}}{g_{00}}$$

$$\sqrt{\left(\frac{dx^0}{du}\right)^2} == \sqrt{-\frac{g_{ij} \frac{dx^i}{du} \frac{dx^j}{du}}{g_{00}}}$$

$$\frac{dx^0}{du} == \sqrt{-\frac{g_{ij} \frac{dx^i}{du} \frac{dx^j}{du}}{g_{00}}}$$

Substitute variable $x^0(u)=c\ t(u)$ and simplify.

$$c\, t'[u] == \sqrt{-\frac{g_{ij} \frac{dx^i}{du} \frac{dx^j}{du}}{g_{00}}}$$

$$t'[u] == \frac{\sqrt{-\frac{g_{ij} \frac{dx^i}{du} \frac{dx^j}{du}}{g_{00}}}}{c}$$

Integrate. We get the coordinate time of travel:

$$-t[u_E]+t[u_R]==\int_{u_E}^{u_R}\frac{\sqrt{-\frac{g_{ij}\frac{dx^i}{du}\frac{dx^j}{du}}{g_{00}}}}{c}\,du$$

The coordinate time delay $t_R - t_E$ between emission and reception depends only on the chosen spatial path between the two spatial points. (We don't actually care about the time of travel so we don't have to evaluate the integral.) For two signals 1 and 2 travelling on the same path we have $t_{R1} - t_{E1} = t_{R2} - t_{E2}$, hence we get always $\Delta t_R \equiv t_{R2} - t_{R1} = t_{E2} - t_{E1} \equiv \Delta t_E$.

```
Print["Proper time delays between two signals in R and E:\n{ΔτR,ΔτE}=",
 {ΔτR = c^-1 Sqrt[gdd[0, 0]R] ΔtR,
  Δτ"E" = c^-1 Sqrt[gdd[0, 0]"E"] Δt"E"}]
Print["Gravitational spectral shift in a static spacetime:"]
νR / ν"E" == (dummy = HoldForm[ n / ΔτR] / HoldForm[ n / Δτ"E"]) ==
   (dummy // ReleaseHold) /. ΔtR → Δt"E" // FrameBox // DisplayForm
```

Proper time delays between two signals in R and E:

$$\{\Delta\tau_R,\Delta\tau_E\}=\left\{\frac{\Delta t_R\sqrt{g_{00R}}}{c},\ \frac{\Delta t_E\sqrt{g_{00E}}}{c}\right\}$$

Gravitational spectral shift in a static spacetime:

$$\boxed{\frac{\nu_R}{\nu_E}==\frac{\frac{n}{\Delta\tau_R}}{\frac{n}{\Delta\tau_E}}==\frac{\sqrt{g_{00E}}}{\sqrt{g_{00R}}}}$$

- **2) Special case:** Schwarzschild spacetime

```
Print["Spectral shift in Schwarzschild spacetime:\nνR/νE ="]
(SpectralShift = (νR = Sqrt[1 - 2 G M / (c^2 r"E")]) / (ν"E" = Sqrt[1 - 2 G M / (c^2 rR)])) ==
  (SpectralShift /. M → m c^2 / G) ≈ Simplify[Series[SpectralShift, {M, 0, 1}]]
Print["Fractional frequency shift in Schwarzschild spacetime:\n(νR-νE)/νE ="]
(FractionalFrequencyShift = νR / ν"E" - 1) == (FractionalFrequencyShift /. M → m c^2 / G) ≈
 FullSimplify[Series[FractionalFrequencyShift, {M, 0, 2}]]
```

Spectral shift in Schwarzschild spacetime:
ν_R/ν_E =

$$\left(\frac{\sqrt{1-\frac{2GM}{c^2 r_E}}}{\sqrt{1-\frac{2GM}{c^2 r_R}}}==\frac{\sqrt{1-\frac{2m}{r_E}}}{\sqrt{1-\frac{2m}{r_R}}}\right)\approx 1+\frac{G\left(-\frac{1}{r_E}+\frac{1}{r_R}\right)M}{c^2}+O[M]^2$$

Fractional frequency shift in Schwarzschild spacetime:
$(\nu_R-\nu_E)/\nu_E$ =

$$\left(-1+\frac{\sqrt{1-\frac{2GM}{c^2 r_E}}}{\sqrt{1-\frac{2GM}{c^2 r_R}}}==-1+\frac{\sqrt{1-\frac{2m}{r_E}}}{\sqrt{1-\frac{2m}{r_R}}}\right)\approx\frac{G\left(-\frac{1}{r_E}+\frac{1}{r_R}\right)M}{c^2}+\frac{G^2(r_E-r_R)(3r_E+r_R)M^2}{2c^4 r_E^2 r_R^2}+O[M]^3$$

Note: If you measure very carefully (to $O(M^2)$ or higher), the gravitational redshift is more then a test of the equivalence principle!

3) Some examples of spectral shifts in a Schwarzschild spacetime

```
Print["Dati approssimativi sulle nane bianche da (approximate white dwarf
   data from): Vittorio Castellani, Astrofisica stellare, p. 270-271."]
const = {G → 6.674 10^-11, c → 299 792 458};
Msun = 1.989 10^30; Rsun = 6.960 10^8;
Print[
 "From fig. 6.21, p.271, we get {log(M/M⊙), log(R/R⊙)} for Sirius B and 40 Eri B:"]
{{0.8 - 1.4 / 49 × 28 // Chop,
  -3.5 + 2 / 69.5 50.5},
 {0.8 - 1.4 / 49 × 40,
  -3.5 + 2 / 69.5 59}}
Print["{M/M⊙, R/R⊙}:"]
{WhiteDwarfSiriusB = 10^%%[[1]],
 WhiteDwarf40EriB = 10^%%[[2]]}
Print["{M, R}:"]
{Msun kilogram, Rsun / 1000 kilometer} # & /@ %%
Print["Check: velocity v for the equivalent Doppler shift."]
v == (G/c^2) (Mr Msun)/(Rr Rsun) c / 1000 kilometer / second /. const
{%[[2]] /. Thread[{Mr, Rr} → WhiteDwarfSiriusB],
 %[[2]] /. Thread[{Mr, Rr} → WhiteDwarf40EriB]}
Print["This agrees only roughly (?) with the results given by Castellani."]
%% / {91 kilometer / second , 22 kilometer / second } 100 "%"
```

Dati approssimativi sulle nane bianche da (approximate white
  dwarf data from): Vittorio Castellani, *Astrofisica stellare*, p. 270-271.

From fig. 6.21, p.271, we get {log(M/M⊙), log(R/R⊙)} for Sirius B and 40 Eri B:

{{0, -2.04676}, {-0.342857, -1.80216}}

{M/M⊙, R/R⊙}:

{{1, 0.0089792}, {0.454091, 0.0157704}}

{M, R}:

$\{\{1.989\times 10^{30}\ \text{kilogram}, 6249.52\ \text{kilometer}\}, \{9.03187\times 10^{29}\ \text{kilogram}, 10\,976.2\ \text{kilometer}\}\}$

Check: velocity v for the equivalent Doppler shift.

$$v == \frac{0.636196\ \text{kilometer Mr}}{\text{Rr second}}$$

$$\left\{\frac{70.8522\ \text{kilometer}}{\text{second}}, \frac{18.3186\ \text{kilometer}}{\text{second}}\right\}$$

This agrees only roughly (?) with the results given by Castellani.

{77.8596 %, 83.2663 %}

```
TableForm[{
  {"object", "νR/νE", "(νR-νE)/νE"},
  {"⊙ (1962)",
   SpectralShift /. (data = const ⋃ {M → Msun, r"E" → Rsun, rR → ∞}),
   FractionalFrequencyShift /. data},
  {"White dwarf 40 Eri B",
   SpectralShift /. (data =
      const ⋃ {M → WhiteDwarf40EriB[[1]] Msun, r"E" → WhiteDwarf40EriB[[2]] Rsun, rR → ∞}),
   FractionalFrequencyShift /. data},
  {"White dwarf Sirius B (Adams,1925)",
```

```
    SpectralShift /. (data = const ⋃
        {M → WhiteDwarfSiriusB[[1]] Msun, r"E" → WhiteDwarfSiriusB[[2]] Rsun, rR → ∞}),
    FractionalFrequencyShift /. data}, {"⊕ Pound-Rebka experiment,1960",
    SpectralShift /.
     (data = const ⋃ {M → 5.976 10^24, r"E" → (Rearth = 6.3710 10^6), rR → Rearth + 22.5}),
    FractionalFrequencyShift /. data},
  {"?: Exercise 4.3, p.131",
    SpectralShift /. (data = const ⋃ {M → 10^30, r"E" → 10^6, rR → ∞}),
    FractionalFrequencyShift /. data}}]
```

object	ν_R/ν_E	$(\nu_R-\nu_E)/\nu_E$
⊙ (1962)	0.999998	-2.12212×10^{-6}
White dwarf 40 Eri B	0.999939	−0.0000611061
White dwarf Sirius B (Adams,1925)	0.999764	−0.000236366
⊕ Pound-Rebka experiment,1960	1.	-2.44249×10^{-15}
?: *Exercise 4.3, p.131*	0.999257	−0.000742859

"Although the Global Positioning System (GPS) is not designed as a test of fundamental physics, it must account for the gravitational redshift in its timing system, and physicists have analyzed timing data from the GPS to confirm other tests. When the first satellite was launched, some engineers resisted the prediction that a noticeable gravitational time dilation would occur, so the first satellite was launched without the clock adjustment that was later built into subsequent satellites. It showed the predicted shift of 38 microseconds per day. This rate of discrepancy is sufficient to substantially impair function of GPS within hours if not accounted for." (*Wikipedia*, 'Tests of general relativity', 2009)

Addendum: The Hafele-Keating experiment (Heuristische ex post Machbarkeitsstudie des Hafele-Keating-Experiments)

Hafele, J.; Keating, R. (July 14, 1972). "Around the world atomic clocks: observed relativistic time gains". Science 177 (4044): 168–170.

Abstract. *Four cesium beam clocks flown around the world on commercial jet flights during October 1971, once eastward and once westward, recorded directionally dependent time differences which are in good agreement with predictions of conventional relativity theory. Relative to the atomic time scale of the U.S. Naval Observatory, the flying clocks lost 59 ± 10 nanoseconds during the eastward trip and gained 273 ± 7 nanoseconds during the westward trip, where the errors are the corresponding standard deviations. These results provide an unambiguous empirical resolution of the famous clock "paradox" with macroscopic clocks.*

A very simple model in the framework of the Schwarzschild metric (neglecting the Earth spin effects implied by the Kerr metric) can account for the experimental data given in the abstract. We assume that two airplanes travel in opposite directions along the equator with constant height h and velocity v. (The Earth is a crazy planet, but not a black hole after all, so we will neglect the difference between coordinate distances and actual distances.)

M = Earth mass, R = Earth radius
A: reference point at ground on the equator *corotating* with the Earth (plays the role of the U.S. Naval Observatory)
B: common round trip start and stop point at height h above A (*corotating* with the Earth)
D: reference point at ground on the equator *fixed* in space
C: reference point at height h above D *fixed* in space
O: east-flying airplane
W: west-flying airplane
ω angular velocity of the Earth rotation measured by D
vA: velocity of A measured by D
vB: velocity of B measured by C

$vO \overset{!}{=} vB + v$, $vW \overset{!}{=} vB - v$: velocity of the east(O)-/west(W)-flying airplane measured by C (definitions). If vB and v are small, then v is nearly exactly the speed of the airplanes relative to B.
tA, tC, tD, tO, tW: round trip time as measured by the various observers
$\Delta tO \overset{!}{=} tO - tA$, $\Delta tW \overset{!}{=} tW - tA$: differences of measured round trip time (definitions)

```
data = {G → 6.674 10^-11, c → 299 792 458, RS → 2 G M / c^2, M → 5.976 × 10^24, R → 6.3710 10^6,
   ω → 7.292 10^-5, ΔtOexp → -59 10^-9, ΔtWexp → +273 10^-9}; (* all data are in SI units *)
mps2kmph = 60 × 60 / 1000 kilometer / hour;

σ[h_] = Sqrt[1 - RS / (R + h)] / Sqrt[1 - RS / R];
(* general relativistic time warping factor Sqrt[g00(R+h)/g00(R)] *)
γ[v_] = 1 / Sqrt[1 - v^2 / c^2];
vB = ω / σ[h] (R + h);
vO = vB + v;
vW = vB - v;
tC = 2 π (R + h) / v;
tO = tC / γ[vO];
tW = tC / γ[vW];
Print["tD = ", tD = tC / σ[h]]
Print["vA = ", (vA = ω R) {meter / second, mps2kmph} /. data]
Print["tA = ", tA = tD / γ[vA]]
Print["ΔtO[h,v] = ", ΔtO[h_, v_] = tO - tA]
Print["ΔtW[h,v] = ", ΔtW[h_, v_] = tW - tA]
```

$$tD = \frac{2\pi(h+R)\sqrt{1-\frac{R_S}{R}}}{v\sqrt{1-\frac{R_S}{h+R}}}$$

$$vA = \left\{\frac{464.573\,\text{meter}}{\text{second}}, \frac{1672.46\,\text{kilometer}}{\text{hour}}\right\}$$

$$tA = \frac{2\pi(h+R)\sqrt{1-\frac{R^2\omega^2}{c^2}}\sqrt{1-\frac{R_S}{R}}}{v\sqrt{1-\frac{R_S}{h+R}}}$$

$$\Delta tO[h,v] = -\frac{2\pi(h+R)\sqrt{1-\frac{R^2\omega^2}{c^2}}\sqrt{1-\frac{R_S}{R}}}{v\sqrt{1-\frac{R_S}{h+R}}} + \frac{2\pi(h+R)\sqrt{1-\frac{\left(v+\frac{(h+R)\,\omega\sqrt{1-\frac{R_S}{R}}}{\sqrt{1-\frac{R_S}{h+R}}}\right)^2}{c^2}}}{v}$$

$$\Delta tW[h,v] = -\frac{2\pi(h+R)\sqrt{1-\frac{R^2\omega^2}{c^2}}\sqrt{1-\frac{R_S}{R}}}{v\sqrt{1-\frac{R_S}{h+R}}} + \frac{2\pi(h+R)\sqrt{1-\frac{\left(-v+\frac{(h+R)\,\omega\sqrt{1-\frac{R_S}{R}}}{\sqrt{1-\frac{R_S}{h+R}}}\right)^2}{c^2}}}{v}$$

```
hh = 10 000;
NSolve[ΔtO[hh, v] == ΔtOexp //. data];
Print["h = ", hh meter, ",  v = ", %[[2, 1, 2]] mps2kmph]
{ΔtO[hh, v], ΔtW[hh, v]} 10^9 ns /. %%[[2]] //. data;
```

```
Print["Our model: ", %]
{ΔtOexp, ΔtWexp} 10^9 ns //. data;
Print["Real data: ", %]
%%%% / %% 100 "%"
```

$h = 10\,000\ \text{meter},\ v = \frac{793.909\ \text{kilometer}}{\text{hour}}$

```
Our model: {-59.0226 ns, 356.114 ns}

Real data: {-59 ns, 273 ns}

{100.038 %, 130.445 %}
```

The chosen cruising height (10000 m) and speed (ca. 800 km/h) are well within the capabilities of an airliner of the '70s, so the experimental data is reproduced with a rather good approximation by our crude model.

4.4 General particle motion (Including photons) p. 136

```
Needs["TensorCalculus3`Tensorial`"]
$PrePrint =.
labs = {x, δ, g, Γ};
DeclareBaseIndices[{0, 1, 2, 3}]
DefineTensorShortcuts[
 {{x, zero, dx}, 1},
 {{δ, g}, 2},
 {{Γ}, 3}]
DeclareZeroTensor[zero]
SetAttributes[c, Constant]
varnames = {c t, r, θ, ϕ};
useSchwarzschild = UseCoordinates[%, x];
```

In the sections 4.5 and 4.6 we need the orbital equations for massive and massless particles in the Schwarzschild metric (SSM). These equations are here deduced in four steps (without using variational methods as do FN).

1) We set up the geodetic equations in the SSM.

```
Print["Schwarzschild metric (SSM) with m=G M/c^2: ", gdd[i, j]]
DiagonalMatrix[{1 - 2 G M c^-2 r^-1, -(1 - 2 G M c^-2 r^-1)^-1, -r^2, -r^2 Sin[θ]^2}] /. (M → m c^2 / G);
SSM = % // CoordinatesToTensors[varnames];
SetMetricValueRules[g, SSM];
SelectedTensorRules[g, gdd[a_, b_]] // useSchwarzschild
Print["Coordinates (i=0,1,2,3) ", xu[i]]
xu[i] // ToArrayValues[] // useSchwarzschild
Print[
 "Christoffel symbols in the SSM (nonzero, nonduplicate up symbols) ", Γudd[i, j, k] ]
SetChristoffelValueRules[xu[i], SSM, Γ, Simplify[#] &]
SelectedTensorRules[Γ, Γudd[_, a_, b_] /; OrderedQ[{a, b}]] // useSchwarzschild
Print["Affinely parametrized geodesic equation
   in the SSM; w is any affine parameter along the geodesic."]
TotalD[xu[i], {w, w}] + Γudd[i, j, k] TotalD[xu[j], w] TotalD[xu[k], w] == zerou[i]
arule = {t → t[w], r → r[w], θ → θ[w], ϕ → ϕ[w]};
GeodesicEqn = %% // ToArrayValues[] // useSchwarzschild /. arule;
Print["Explicitely:"]
%% // TableForm // TraditionalForm
```

Schwarzschild metric (SSM) with m=G M/c^2: g_{ij}

$$\left\{g_{00}\to 1-\frac{2\,m}{r},\ g_{11}\to -\frac{1}{1-\frac{2\,m}{r}},\ g_{22}\to -r^2,\ g_{33}\to -r^2\,\text{Sin}[\theta]^2\right\}$$

Coordinates (i=0,1,2,3) x^i

$\{c\,t,\ r,\ \theta,\ \phi\}$

Christoffel symbols in the SSM (nonzero, nonduplicate up symbols) $\Gamma^i{}_{jk}$

$$\left\{\Gamma^0{}_{01}\to -\frac{m}{2\,m\,r-r^2},\ \Gamma^1{}_{00}\to \frac{m\,(-2\,m+r)}{r^3},\ \Gamma^1{}_{11}\to \frac{m}{2\,m\,r-r^2},\ \Gamma^1{}_{22}\to 2\,m-r,\right.$$
$$\left.\Gamma^1{}_{33}\to (2\,m-r)\,\text{Sin}[\theta]^2,\ \Gamma^2{}_{12}\to \frac{1}{r},\ \Gamma^2{}_{33}\to -\frac{1}{2}\,\text{Sin}[2\,\theta],\ \Gamma^3{}_{13}\to \frac{1}{r},\ \Gamma^3{}_{23}\to \text{Cot}[\theta]\right\}$$

Affinely parametrized geodesic equation in the SSM; w is any affine parameter along the geodesic.

$$\frac{d^2x^i}{dw\,dw}+\Gamma^i{}_{jk}\,\frac{dx^j}{dw}\,\frac{dx^k}{dw}==\text{zero}^i$$

Explicitely:

$$c\,t''(w)-\frac{2\,c\,m\,r'(w)\,t'(w)}{2\,m\,r(w)-r(w)^2}=0$$
$$\frac{m\,r'(w)^2}{2\,m\,r(w)-r(w)^2}+\frac{c^2\,m\,(r(w)-2\,m)\,t'(w)^2}{r(w)^3}+(2\,m-r(w))\,\theta'(w)^2+(2\,m-r(w))\sin^2(\theta(w))\,\phi'(w)^2+r''(w)=0$$
$$-\frac{1}{2}\sin(2\,\theta(w))\,\phi'(w)^2+\frac{2\,r'(w)\,\theta'(w)}{r(w)}+\theta''(w)=0$$
$$\frac{2\,r'(w)\,\phi'(w)}{r(w)}+2\cot(\theta(w))\,\theta'(w)\,\phi'(w)+\phi''(w)=0$$

2) Note that (abc)' = a'bc + ab'c + abc', so multiplication of the first and the forth geodesic equation with a *guessed* integration factor and integration in the variable w leads to two useful constant of motions. (Eyebrow raised? Then resort to variational methods and guess $\mathcal{L}$...) The third geodesic equation shows that in the SSM too all orbits will be in flat planes through the origin.

```
Print["a) Constant of motion from the first geodesic equation."]
(GeodesicEqn[[1, 1]] c^-1 (1 - 2 m / r[w]) // Simplify) == 0
Integrate[%[[1]], w];
Print["eq.(4.22),p.136"]
Collect[%% // Expand, t′[w]] == "const" == k
trule = Solve[%[[1]] == %[[3]], t′[w]][[1, 1]]
Print["b) Constant of motion from the forth geodesic equation."]
(GeodesicEqn[[4, 1]] r[w]^2 Sin[θ[w]]^2 // Expand) == 0
Print["compare with eq.(4.23),p.136"]
Simplify[Integrate[%%[[1]], w]] == "const" == h
ϕrule = Solve[%[[1]] == %[[3]], ϕ′[w]][[1, 1]]
Print["c) Spherical symmetry allows us to choose a coordinate system with θ[0]=π/2
   and θ'[0]=0 as our initial condition. Then the third geodesic equation
   implies that θ[w]=π/2=const, hence the orbit lays in the equatorial plane."]
GeodesicEqn[[3]] /. w → 0
% /. {θ[0] → π / 2, θ'[0] → 0}
```

a) Constant of motion from the first geodesic equation.

$$\frac{2\,m\,r'[w]\,t'[w]+r[w]\,(-2\,m+r[w])\,t''[w]}{r[w]^2}==0$$

eq.(4.22),p.136

$$\left(1-\frac{2\,m}{r[w]}\right)t'[w]==\text{const}==k$$

$$t'[w] \to -\frac{k\, r[w]}{2m - r[w]}$$

b) Constant of motion from the forth geodesic equation.

$$2\, r[w]\, \mathrm{Sin}[\theta[w]]^2\, r'[w]\, \phi'[w] + 2\, \mathrm{Cos}[\theta[w]]\, r[w]^2\, \mathrm{Sin}[\theta[w]]\, \theta'[w]\, \phi'[w] + r[w]^2\, \mathrm{Sin}[\theta[w]]^2\, \phi''[w] == 0$$

compare with eq.(4.23),p.136

$$r[w]^2\, \mathrm{Sin}[\theta[w]]^2\, \phi'[w] == \mathrm{const} == h$$

$$\phi'[w] \to \frac{h\, \mathrm{Csc}[\theta[w]]^2}{r[w]^2}$$

c) Spherical symmetry allows us to choose a coordinate system with
$\theta[0]=\pi/2$ and $\theta'[0]=0$ as our initial condition. Then the third geodesic equation
implies that $\theta[w]=\pi/2=$const, hence the orbit lays in the equatorial plane.

$$\frac{2\, r'[0]\, \theta'[0]}{r[0]} - \frac{1}{2} \mathrm{Sin}[2\,\theta[0]]\, \phi'[0]^2 + \theta''[0] == 0$$

$$\theta''[0] == 0$$

3) From the definition of the spacetime line element ds we obtain:

```
Print[
 "Spacetime line element; ds=c dτ for massive particles, ds=0 for massless particles"]
ds² == gdd[i, j] dxu[i] dxu[j]
Print["Line element in derivative form;
   K=(ds/dw)²=const≠0 for massive particles, K=0 for massless particles"]
HoldForm[ds/dw ds/dw] == K == gdd[i, j] TotalD[xu[i], w] TotalD[xu[j], w]
Print["Expanding and substituting variables for SSM (compare with eq.(4.24),p.136)"]
eqn[4, 24] = Reverse[Rest[%%]] // ToArrayValues[] // useSchwarzschild /. arule
```

Spacetime line element; ds=c dτ for massive particles, ds=0 for massless particles

$$ds^2 == dx^i\, dx^j\, g_{ij}$$

Line element in derivative form;
K=(ds/dw)^2=const≠0 for massive particles, K=0 for massless particles

$$\frac{ds\, ds}{dw\, dw} == K == g_{ij} \frac{dx^i}{dw} \frac{dx^j}{dw}$$

Expanding and substituting variables for SSM (compare with eq.(4.24),p.136)

$$-\frac{r'[w]^2}{1-\frac{2m}{r[w]}} + c^2 \left(1 - \frac{2m}{r[w]}\right) t'[w]^2 - r[w]^2\, \theta'[w]^2 - r[w]^2\, \mathrm{Sin}[\theta[w]]^2\, \phi'[w]^2 == K$$

4) Some manipulation of the above equation and using the properties derived in 2) leads finally to the orbital equations for massive and massless particles in the SSM.

```
Print["Divide by ϕ′[w]^2"]
Expand[# / ϕ′[w]^2] & /@ eqn[4, 24]
Print["Substitute r′[w]^2/ ϕ′[w]^2 by r′[ϕ]^2"]
%% /. r′[w]^2 / ϕ′[w]^2 → Dt[r[ϕ], ϕ]^2
Print["Replacing t'[w] and ϕ'[w] using the constant of motions derived above"]
%% /. {trule, ϕrule}
Print["Choose a coordinate system with the
   orbit laying in the equatorial plane: θ[w]=π/2=const; simplify"]
%% /. θ → (π / 2 &);
Simplify[(1 - 2 m / r[w]) #] & /@ %;
Expand /@ %
Print["Substitute with reparametrization r[w]→1/u[ϕ] and simplify"]
%% /. r[w] → 1 / u[ϕ] /. Dt[r[ϕ], ϕ] → Dt[1 / u[ϕ], ϕ];
Expand[-u[ϕ]^4 #] & /@ %;
step1 = # - %[[2]] & /@ %
Print["a) Orbital equation for massive particles in SSM. Using
   proper time τ=w as affine parameter implies K=c^2. eq.(4.25),p.137:"]
step1 /. K → c^2;
c^2 (1 - k^2) / h^2 → -En
%% /. (Expand /@ %)
(eqn[4, 25] = % /. m → G M / c^2) // FrameBox // DisplayForm
Print["b) Orbital equation for massless particles in SSM. K=0. eq.(4.39),p.142:"]
c^2 k^2 / h^2 → F
step1 /. K → 0 /. %
(eqn[4, 39] = % /. m → G M / c^2) // FrameBox // DisplayForm
```

Divide by ϕ′[w]^2

$$-r[w]^2 \operatorname{Sin}[\theta[w]]^2 - \frac{r'[w]^2}{\left(1-\frac{2m}{r[w]}\right)\phi'[w]^2} + \frac{c^2\, t'[w]^2}{\phi'[w]^2} - \frac{2 c^2 m\, t'[w]^2}{r[w]\,\phi'[w]^2} - \frac{r[w]^2\,\theta'[w]^2}{\phi'[w]^2} == \frac{K}{\phi'[w]^2}$$

Substitute r′[w]^2/ ϕ′[w]^2 by r′[ϕ]^2

$$-r[w]^2 \operatorname{Sin}[\theta[w]]^2 - \frac{r'[\phi]^2}{1-\frac{2m}{r[w]}} + \frac{c^2\, t'[w]^2}{\phi'[w]^2} - \frac{2 c^2 m\, t'[w]^2}{r[w]\,\phi'[w]^2} - \frac{r[w]^2\,\theta'[w]^2}{\phi'[w]^2} == \frac{K}{\phi'[w]^2}$$

Replacing t'[w] and ϕ'[w] using the constant of motions derived above

$$-r[w]^2 \operatorname{Sin}[\theta[w]]^2 - \frac{2 c^2 k^2 m\, r[w]^5 \operatorname{Sin}[\theta[w]]^4}{h^2\,(2m - r[w])^2} + \frac{c^2 k^2\, r[w]^6 \operatorname{Sin}[\theta[w]]^4}{h^2\,(2m - r[w])^2} - \frac{r'[\phi]^2}{1-\frac{2m}{r[w]}} - \frac{r[w]^6 \operatorname{Sin}[\theta[w]]^4\,\theta'[w]^2}{h^2} == \frac{K\, r[w]^4 \operatorname{Sin}[\theta[w]]^4}{h^2}$$

Choose a coordinate system with the orbit laying in the equatorial plane: θ[w]=π/2=const; simplify

$$2 m\, r[w] - r[w]^2 + \frac{c^2 k^2\, r[w]^4}{h^2} - r'[\phi]^2 == -\frac{2 K m\, r[w]^3}{h^2} + \frac{K\, r[w]^4}{h^2}$$

Substitute with reparametrization r[w]→1/u[ϕ] and simplify

$$-\frac{c^2 k^2}{h^2} + \frac{K}{h^2} - \frac{2 K m\, u[\phi]}{h^2} + u[\phi]^2 - 2 m\, u[\phi]^3 + u'[\phi]^2 == 0$$

a) *Orbital equation for massive particles in SSM.*
Using proper time τ=w as affine parameter implies K=c^2. eq.(4.25),p.137:

$$\frac{c^2\left(1-k^2\right)}{h^2}\to -En$$

$$-En-\frac{2c^2\,m\,u[\phi]}{h^2}+u[\phi]^2-2\,m\,u[\phi]^3+u'[\phi]^2==0$$

$$\boxed{-En-\frac{2\,G\,M\,u[\phi]}{h^2}+u[\phi]^2-\frac{2\,G\,M\,u[\phi]^3}{c^2}+u'[\phi]^2==0}$$

b) *Orbital equation for massless particles in SSM.* K=0. eq.(4.39),p.142:

$$\frac{c^2\,k^2}{h^2}\to F$$

$$-F+u[\phi]^2-2\,m\,u[\phi]^3+u'[\phi]^2==0$$

$$\boxed{-F+u[\phi]^2-\frac{2\,G\,M\,u[\phi]^3}{c^2}+u'[\phi]^2==0}$$

"Make things as simple as possible, but not simpler."

4.5 Perihelion advance p. 144

We follow the Møller argument to derive the advance of the perihelion and consider here only *planetary motion*, this means bound nearly elliptical orbits around the Sun.

Wegen des zusätzlichen r^{-3}-Terms ist die Gravitation in der ART stärker anziehend als die klassische Newtonsche Gravitation. Die Bahnkurven der Planeten um die Sonne sind nur noch näherungsweise geschlossene Ellipsen mit großer Halbachse $\mathcal{A}$ und Exzentrizität e, auf denen r periodisch zwischen dem Aphel r1 = $\mathcal{A}(1+e)$ = 1/u1 und dem Perihel r2 = $\mathcal{A}(1-e)$ = 1/u2 hin und her pendelt. Es wird ein etwas größerer Winkel als 2π von Perihel zu Perihel durchlaufen; das Perihel verschiebt sich in Umlaufrichtung nach vorne.

1) We derive an expression for u'(ϕ). (Note: inverse radial coordinate u = 1 / r.)

```
Print["Orbital equation for massive particles
   in the Schwarzschild metric, eq.(4.25) with m=G M/c²"]
(-En - (2 c² m u[ϕ])/h² + u[ϕ]² - 2 m u[ϕ]³ + u′[ϕ]² == 0)
Print["Rearrange and put ϵ=2 m, but only in the u³ term (the 'GR correction' term):"]
(# - Most[%%[[1]]] &/@%%) /. (2 m u[ϕ]³ → ϵ u[ϕ]³)
Print["Solving the cubic equation for u at turning
   points (where u'=0) yields three solutions u1 (aphelion) < u2
   (perihelion) and u3 (≈ 1/(2 m) in our context). Vieta states:"]
u1 + u2 + u3 == -Coefficient[%%[[2]], u[ϕ]²] / Coefficient[%%[[2]], u[ϕ]³]
Print["Differential equation in terms of degree three u polynomial and its roots"]
u′[ϕ]² == ϵ (u - u1) (u - u2) (u - u3)
Solve[%%%, u3] // Simplify // Flatten;
Print["Substituting ", %[[1]]]
%%% /. %%
Print["Taking the square root of both sides; u'>0 on (ϕ[u1],ϕ[u2])"]
Sqrt[#] &/@%%;
step1 = MapAt[PowerExpand, %, {1}]
```

Orbital equation for massive particles in the Schwarzschild metric, eq.(4.25) with m=G M/c^2

$$-En - \frac{2 c^2 m u[\phi]}{h^2} + u[\phi]^2 - 2 m u[\phi]^3 + u'[\phi]^2 == 0$$

Rearrange and put ϵ=2 m, but only in the u^3 term (the 'GR correction' term):

$$u'[\phi]^2 == En + \frac{2 c^2 m u[\phi]}{h^2} - u[\phi]^2 + \epsilon u[\phi]^3$$

Solving the cubic equation for u at turning points (where u'=0) yields three solutions u1 (aphelion) < u2 (perihelion) and u3 ($\approx \frac{1}{2 m}$ in our context). Vieta states:

$$u1 + u2 + u3 == \frac{1}{\epsilon}$$

Differential equation in terms of degree three u polynomial and its roots

$$u'[\phi]^2 == (u - u1)(u - u2)(u - u3)\epsilon$$

Substituting $u3 \to -u1 - u2 + \frac{1}{\epsilon}$

$$u'[\phi]^2 == (u - u1)(u - u2)\left(u + u1 + u2 - \frac{1}{\epsilon}\right)\epsilon$$

Taking the square root of both sides; u'>0 on (ϕ[u1],ϕ[u2])

$$u'[\phi] == \sqrt{(u - u1)(u - u2)\left(u + u1 + u2 - \frac{1}{\epsilon}\right)\epsilon}$$

2) From u'(ϕ) we derive now a suitable approximate expression for ϕ'(u) in the case of weak relativistic perturbation.

```
1 / # & /@ step1
Print[
 "Power series expansion for 1/u′[ϕ] about the point ϵ=0 to first order in ϵ; simplify"]
Series[Part[%%, 2], {ϵ, 0, 1}]
1 / u'[ϕ] ≈ (% // Normal // Together)
Print["We have 1/u'[ϕ]=ϕ'[u], so on (u1,u2)"]
ϕ'[u] ≈ %%[[2]]
Print["Change of variables"]
{α == 1 / 2 (u1 + u2), β == 1 / 2 (u2 - u1)}
u1u2rules = Solve[%, {u1, u2}] // Flatten
step2 = MapAt[Expand, %%%% /. %, {2, 2, 1}] // Simplify
```

$$\frac{1}{u'[\phi]} == \frac{1}{\sqrt{(u-u1)\ (u-u2)\ \left(u+u1+u2-\frac{1}{\epsilon}\right)\ \epsilon}}$$

Power series expansion for 1/u′[ϕ] about the point ϵ=0 to first order in ϵ; simplify

$$\frac{1}{\sqrt{(-u+u1)\ (u-u2)}} - \frac{((u-u1)\ (u+u1+u2))\ \epsilon}{2\left((-u+u1)\ \sqrt{(-u+u1)\ (u-u2)}\right)} + O[\epsilon]^2$$

$$\frac{1}{u'[\phi]} \approx \frac{2+u\,\epsilon+u1\,\epsilon+u2\,\epsilon}{2\sqrt{(-u+u1)\ (u-u2)}}$$

We have 1/u'[ϕ]=ϕ'[u], so on (u1,u2)

$$\phi'[u] \approx \frac{2+u\,\epsilon+u1\,\epsilon+u2\,\epsilon}{2\sqrt{(-u+u1)\ (u-u2)}}$$

Change of variables

$$\left\{\alpha == \frac{u1+u2}{2},\ \beta == \frac{1}{2}\ (-u1+u2)\right\}$$

$$\{u1 \to \alpha-\beta,\ u2 \to \alpha+\beta\}$$

$$\phi'[u] \approx \frac{2+u\,\epsilon+2\,\alpha\,\epsilon}{2\sqrt{-u^2+2\,u\,\alpha-\alpha^2+\beta^2}}$$

3) Finally, to calculate $\Delta\phi$ from the above equation we integrate the rhs from aphelion u1 to perihelion u2. (We substitute u1 and u2 in terms of α and β.)

```
HoldForm[∫_{u1=α-β}^{u2=α+β} ϕ′[u] ⅆu] ==
 Integrate[Part[step2, 2], {u, u1, u2} /. u1u2rules, Assumptions → 0 < β]
Print["Multiplying by 2 for a complete orbit"]
2 %%[[2]] // Distribute
Print["The advance of the perihelion in one orbit is"]
Δϕ == %% - 2 π
Print["Substituting for α and ϵ, u=1/r, we get eq.(4.45)"]
(eqn[4, 45] = %% //. {α → (u1 + u2) / 2, ϵ → 2 m, u1 → 1 / r1, u2 → 1 / r2}) // FrameBox //
 DisplayForm
Print["Substituing m"]
eqn[4, 45] /. m → G M / c^2
```

$$\int_{u1=\alpha-\beta}^{u2=\alpha+\beta} \phi'[u]\,\mathrm{d}u == \pi + \frac{3\,\pi\,\alpha\,\epsilon}{2}$$

Multiplying by 2 for a complete orbit

$2\pi + 3\pi\alpha\epsilon$

The advance of the perihelion in one orbit is

$\Delta\phi == 3\pi\alpha\epsilon$

Substituting for α and ϵ, u=1/r, we get eq.(4.45)

$$\Delta\phi == 3\,m\,\pi\left(\frac{1}{r1} + \frac{1}{r2}\right)$$

Substituing m

$$\Delta\phi == \frac{3\,G\,M\,\pi\left(\frac{1}{r1} + \frac{1}{r2}\right)}{c^2}$$

4) We can express M, r1 and r2 in terms of classical elliptical orbit parameters getting Einstein's approximate formula for the perihelion advance.

```
Print["If 𝒜 is the semimajor axis, e is the eccentricity and applying
   Kepler's third law with T meaning the sidereal orbit period, we get"]
{r1 → 𝒜 (1 + e), r2 → 𝒜 (1 - e), M → 4 π^2 𝒜^3 / (G T^2)}
(eqn[4, 45 a] = ((%%% /. % // Simplify) /. (-1 + e^2) → -Hold[1 - e^2]) // ReleaseHold) //
  FrameBox // DisplayForm
```

If 𝒜 is the semimajor axis, e is the eccentricity and applying Kepler's third law with T meaning the sidereal orbit period, we get

$$\left\{r1 \to \mathcal{A}\,(1+e),\ r2 \to \mathcal{A}\,(1-e),\ M \to \frac{4\,\pi^2\,\mathcal{A}^3}{G\,T^2}\right\}$$

$$\Delta\phi == \frac{24\,\pi^3\,\mathcal{A}^2}{c^2\,T^2\,(1-e^2)}$$

Dies ist genau die Formel (113) auf Seite 95 in Albert Einsteins *Grundzüge der Relativitätstheorie*.

```
Print["Perihelion advance of Mercury - theory (orbital data: NASA 2010)"]
eqn[4, 45 a][[2]] /. {c → 299792458, 𝒜 → 5.791 × 10^10 , e → 0.2056, T → 87.969 × 24 × 60 × 60};
% 100 / (87.969 / 365.256);
% 360 / (2 π) × 60 × 60 ArcSeconds / Century
```

Perihelion advance of Mercury - theory (orbital data: NASA 2010)

$$\frac{42.9823\ \text{ArcSeconds}}{\text{Century}}$$

"Dieser Ausdruck liefert die Erklärung für die seit hundert Jahren (seit Le Verrier) bekannte Perihelbewegung des Planeten Merkur von etwa 42" in hundert Jahren, welche die theoretische Astronomie bisher nicht in befriedigender Weise zu deuten vermochte." Albert Einstein

4.6 Bending of light p. 146

We consider a photon originating from infinity in the direction $\phi = 0$, travelling in the equatorial plane ($\theta = \pi/2$), passing a massive object (mass M, lying on the origin) and going off to infinity in the direction $\phi = \pi + \alpha$, where α is a deflection angle to be determined ($\alpha = 0$ in flat spacetime where M = 0). Let be r_0 the radial coordinate of the point on the path nearest to the origin and u=1/r the inverse radial coordinate.

```
Print["Orbital equation for massless particles in the Schwarzschild metric, eq.(4.39)"]
eqn[4, 39] = (-F + u[ϕ]^2 - (2 G M u[ϕ]^3)/c^2 + u'[ϕ]^2 == 0)
Print["Put ϵ = 2GM/c^2. We consider ϵ u^3
   a relativistic correction to the flat spacetime equation."]
eqn[4, 39, b] = %% /. M → ϵ c^2 / (2 G)
Print["If u0=1/r0 is the point of closest approach then u'[u0]=0."]
%% /. {u'[ϕ] → 0, u[ϕ] → u0}
Print["Solve for F and substitute into the general equation"]
Frule = Solve[%%, F][[1, 1]]
(eqn[4, 49] = eqn[4, 39, b] /. Frule)
SetAttributes[{ϵ, u0}, Constant]
```

Orbital equation for massless particles in the Schwarzschild metric, eq.(4.39)

$$-F + u[\phi]^2 - \frac{2 G M u[\phi]^3}{c^2} + u'[\phi]^2 == 0$$

Put ϵ = 2GM/c^2. We consider ϵ u^3 a relativistic correction to the flat spacetime equation.

$$-F + u[\phi]^2 - \epsilon u[\phi]^3 + u'[\phi]^2 == 0$$

If u0=1/r_0 is the point of closest approach then u'[u0]=0.

$$-F + u0^2 - u0^3 \epsilon == 0$$

Solve for F and substitute into the general equation

$$F \to u0^2 - u0^3 \epsilon$$

$$-u0^2 + u0^3 \epsilon + u[\phi]^2 - \epsilon u[\phi]^3 + u'[\phi]^2 == 0$$

We will solve this equation by a perturbation method. The equation should have a solution close to the flat spacetime solution: $u[\phi] == u0 \operatorname{Sin}[\phi] + \epsilon v[\phi]$. We substitute into the equation and work to first order in ϵ.

```
Print["Substituting perturbed solution"]
u[ϕ] → u0 Sin[ϕ] + ϵ v[ϕ]
eqn[4, 49] /. {%, Dt[#, ϕ] & /@ %}
Print["Working to first order in ϵ"]
Series[#, {ϵ, 0, 1}] & /@ %% // Normal
Print[
 "Solve the differential equation for v[ϕ] (integration constant A) and simplify."]
DSolve[%%, v[ϕ], ϕ]
%[[1, 1, 2]] /. C[1] → A // Expand
step1 = %[[2]] + Factor[%[[{1, 3, 4}]]]
Print["We fix A by requiring that the photon originates from infinity in the
   direction ϕ=0, so u[0]=0. Hence impose v[0]=0, solve for A and substitute"]
step1 == 0 /. ϕ → 0
Solve[%, A][[1,1]]
```

```
step1 = step1 /. %
Print["Substitute the above into the perturbed u[ϕ] solution"]
step1 = u0 Sin[ϕ] + ϵ step1
```

Substituting perturbed solution

$u[\phi] \to u0\, Sin[\phi] + \epsilon\, v[\phi]$

$-u0^2 + u0^3 \epsilon + (u0\, Sin[\phi] + \epsilon\, v[\phi])^2 - \epsilon\, (u0\, Sin[\phi] + \epsilon\, v[\phi])^3 + (u0\, Cos[\phi] + \epsilon\, v'[\phi])^2 == 0$

Working to first order in ϵ

$\epsilon \left(u0^3 - u0^3\, Sin[\phi]^3 + 2\, u0\, Sin[\phi]\, v[\phi] + 2\, u0\, Cos[\phi]\, v'[\phi]\right) == 0$

Solve the differential equation for v[ϕ] (integration constant A) and simplify.

$\left\{\left\{v[\phi] \to C[1]\, Cos[\phi] + Cos[\phi] \left(\frac{1}{2} u0^2\, Cos[\phi] + \frac{1}{2} u0^2\, Sec[\phi] - \frac{1}{2} u0^2\, Tan[\phi]\right)\right\}\right\}$

$\frac{u0^2}{2} + A\, Cos[\phi] + \frac{1}{2} u0^2\, Cos[\phi]^2 - \frac{1}{2} u0^2\, Sin[\phi]$

$A\, Cos[\phi] + \frac{1}{2} u0^2 \left(1 + Cos[\phi]^2 - Sin[\phi]\right)$

We fix A by requiring that the photon originates from infinity in
the direction ϕ=0, so u[0]=0. Hence impose v[0]=0, solve for A and substitute

$A + u0^2 == 0$

$A \to -u0^2$

$-u0^2\, Cos[\phi] + \frac{1}{2} u0^2 \left(1 + Cos[\phi]^2 - Sin[\phi]\right)$

Substitute the above into the perturbed u[ϕ] solution

$\epsilon \left(-u0^2\, Cos[\phi] + \frac{1}{2} u0^2 \left(1 + Cos[\phi]^2 - Sin[\phi]\right)\right) + u0\, Sin[\phi]$

If M ≠ 0 the photon will no longer depart at the angle π as in flat spacetime but rather at the angle $\pi + \alpha$. We evaluate this under the assumptions that α and ϵ are small.

```
Print["Put ϕ=π+α in the expression for u[ϕ]"]
step1 /. ϕ → π + α
Print["Since α is small, expand to first order in α"]
Series[%%, {α, 0, 1}] // Normal
Print["Set u to zero when the photon approaches infinity and solve for α"]
%% == 0
Solve[%, α][[1,1]]
Print["Expand to first order in ϵ"]
(Equal @@ MapAt[Normal[Series[#, {ϵ, 0, 1}]] &, %%, 2])
Print["Subsitute ϵ and u0. We get finally eq.(4.51), p.149."]
%% /. ϵ → 2 m /. u0 → 1 / r0
(eqn[4, 51] = % /. m → G M / c^2) // FrameBox // DisplayForm
```

Put ϕ=π+α in the expression for u[ϕ]

$-u0\, Sin[\alpha] + \epsilon \left(u0^2\, Cos[\alpha] + \frac{1}{2} u0^2 \left(1 + Cos[\alpha]^2 + Sin[\alpha]\right)\right)$

Since α is small, expand to first order in α

$$2\, u0^2\, \epsilon + \alpha \left(-u0 + \frac{u0^2\, \epsilon}{2}\right)$$

Set u to zero when the photon approaches infinity and solve for α

$$2\, u0^2\, \epsilon + \alpha \left(-u0 + \frac{u0^2\, \epsilon}{2}\right) == 0$$

$$\alpha \to -\frac{4\, u0\, \epsilon}{-2 + u0\, \epsilon}$$

Expand to first order in ϵ

$$\alpha == 2\, u0\, \epsilon$$

Subsitute ϵ and u0. We get finally eq.(4.51), p.149.

$$\alpha == \frac{4\, m}{r_0}$$

$$\boxed{\alpha == \frac{4\, G\, M}{c^2\, r_0}}$$

So in its flight past a massive object (mass M) with impact parameter r_0 the photon is deflected through the angle α.

```
Print["Deflection of light passing the Sun at grazing incidence - theory"]
eqn[4, 51][[2]] /. {G → 6.67384 × 10^-11, c → 299 792 458, M → 1.9891 × 10^30, r0 → 6.9599 × 10^8 };
% 360 / (2 π) × 60 × 60 ArcSecond
```

Deflection of light passing the Sun at grazing incidence - theory

1.75095 ArcSecond

"Der bisher provisorisch ermittelte Wert liegt zwischen 0,9 und 1,8 Bogensekunden. Die Theorie fordert 1,7."
Albert Einstein, *Prüfung der allgemeinen Relativitätstheorie*, Die Naturwissenschaften, 7, 1919, S. 776.

4.7 Geodesic effect p. 149

coming soon...

4.8 Black holes p. 152

coming soon...

4.9 Other coordinate systems p. 157

coming soon...

4.10 Rotating objects; the Kerr solution p. 167

coming soon...

Chapter 5: Gravitational radiation

Tech note: This chapter was written with *Mathematica* 12.

In[•]:=
```
Print["This system is:"]
{"ProductIDName", "ProductVersion"} /. $ProductInformation
ReadList["!ver", String][[1]]
{$MachineType, $ProcessorType, $ByteOrdering, $SystemCharacterEncoding}
```

```
This system is:
```

Out[•]=
```
{Mathematica, 12.1.0 for Microsoft Windows (64-bit) (March 14, 2020)}
```

Out[•]=
```
Microsoft Windows [Versione 10.0.18363.1016]
```

Out[•]=
```
{PC, x86-64, -1, WindowsANSI}
```

5.0 Introduction p. 169-170

FN: "We shall see in Section 5.3 that it is the second time derivative of the *second moment of the mass distribution* of the source that produces the radiation, showing that it is predominantly *quadrupole*."

5.1 What wiggles? p. 170-173

In[•]:=
```
Needs["Notation`"]
Notation[ h̄ ⟺ hb]
(* Grafikhinweis: h̄ ist Overscript[h,"-\[NegativeMediumSpace]-"] *)

Needs["TGeneralRelativity1`GeneralRelativity`"]
$PrePrint=.
DeclareBaseIndices[{0, 1, 2, 3}]
MyRed = StyleForm[Superscript[#, "′"], FontColor → RGBColor[1, 0, 0]] &;
DeclareIndexFlavor[{red, MyRed}]
labs = {x, δ, g, Γ};
DefineTensorShortcuts[
  {{x, ξ}, 1},
  {{g, η, δ, R, T, h, hb, x, ξ, X}, 2},
  {{Γ, h}, 3},
  {{R, h}, 4}];
SetTensorValueRules[ηdd[i, j], DiagonalMatrix[{1, -1, -1, -1}]]
SetTensorValueRules[ηuu[i, j], DiagonalMatrix[{1, -1, -1, -1}]]
SetTensorValues[δud[i, j], IdentityMatrix[NDim]]

(* One more little adjustment...*)
(
 Unprotect[PartialD];
 PartialD[l_][T_, {c t, c t}] := HoldForm[c^-2] × PartialD[l][T, {t, t}];
```

```
 Protect[PartialD];
)
```

--- **Get**: Cannot open Graphics`Colors`.

--- **Needs**: Context Graphics`Colors` was not created when Needs was evaluated.

--- **SetDelayed**: Tag Symmetric in Symmetric[args__ /; Length[{args}] > 1][term_] is Protected.

--- **SetDelayed**: Tag Symmetric in Symmetric[ind_][term_] is Protected.

Tech note:

- *Mathematica* 12 complains of some problems when loading the Tensorial package (see previous error messages) which are not disturbing here, but which should be easy to fix anyway.
- The colored tensor indices are unambiguous and very useful to prevent mistakes, but I add primes nonetheless (see the `MyRed` code), so that the notation survives a black and white printout or similar incidents...

Utilization note: David Park: "Since we are working with Cartesian coordinates we can freely raise and lower indices with partial derivatives. But Tensorial doesn't allow that because in general it is incorrect. To overcome this limitation we use the following functions [`ToNonDifTensor`, `ToDifTensor`] from the GeneralRelativity subpackage that convert [ordinary] `Dif` indices into [plain] indices and vice versa. Then partial differentiations are understood for all slot numbers beyond a fixed slot. This gives us the freedom to raise and lower indices on those slots."

Example

ordinary notation: $T_{\alpha\beta,\gamma}{}^{,\delta}$ $\longleftrightarrow$ plain notation: $T_{\alpha\beta\gamma}{}^{\delta}$ and partial differentiations understood

In[ ]:=
```
? ToNonDifTensor
```

Out[ ]=

Symbol

ToNonDifTensor[label][expr] will convert all partial differentiation indices in tensors with the specified label to ordinary indices. i.e., Dif[x] → x

In[ ]:=
```
? ToDifTensor
```

Out[ ]=

Symbol

ToDifTensor[label, firstdifslot][expr] will convert all tensors in expr with the given label so that all indices from the firstdifslot number on will be partial differentiation indices. i.e., non Void indices x go to Dif[x].

▲ 1) A wave equation for gravitational radiation

We will assume that over extensive regions of spacetime there exist *nearly Cartesian coordinate systems* in which the metric $g_{\mu\nu}$ is

In[ ]:=
```
gdd[μ, ν] == ηdd[μ, ν] + hdd[μ, ν];
RowBox[{%[[1]], "=", %[[2, 2]], "+", %[[2, 1]]}] // FrameBox // DisplayForm
```

Out[∘]//DisplayForm=

$$g_{\mu\nu} = \eta_{\mu\nu} + h_{\mu\nu}$$

where the *flat space metric in Cartesian coordinates* $\eta_{\mu\nu}$ is given by

```
In[∘]:= ηdd[μ, ν] == (ToArrayValues[][ηdd[μ, ν]] // MatrixForm)
```

Out[∘]= $\eta_{\mu\nu} == \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}$

and where $h_{\mu\nu}$ is the *perturbation*. All quantities (including the relevant derivatives) having the kernel letter h are small; products of them are ignored. Suffixes (including indices with partial derivatives) are raised and lowered using $\eta^{\mu\nu}$ and $\eta_{\mu\nu}$ rather than $g^{\mu\nu}$ and $g_{\mu\nu}$. So, in the following "equality" means "exact equality" or "equality up to the first order in small quantities".

We validate first an approximation for the up metric $g^{\mu\nu}$ (see also FN solution to Exercise 2.7.1, p.265):

```
In[∘]:= Print["With the ansatz"]
guu[μ, ν] == ηuu[μ, ν] - huu[μ, ν];
RowBox[{%[[1]], "=", %[[2, 2]], "-", %[[2, 1, 2]]}] // FrameBox // DisplayForm
Print["we get"]
gdd[μ, ν] × guu[σ, μ]
(ηdd[μ, ν] + hdd[μ, ν]) (ηuu[σ, μ] - huu[σ, μ])
% // Expand
Print["Simplify metric and neglect h products:"]
%% // MetricSimplify[η]
% /. HoldPattern[Tensor[h, __] × Tensor[h, __]] → 0
Print["But this is the Kronecker delta:"]
ηdd[μ, ν] × ηuu[σ, μ] == δud[σ, ν]
% // MetricSimplify[η]
ToArrayValues[][%%[[1]]] == ToArrayValues[][%%[[2]]]
```

With the *ansatz*

Out[∘]//DisplayForm=

$$g^{\mu\nu} = \eta^{\mu\nu} - h^{\mu\nu}$$

we get

Out[∘]= $g_{\mu\nu}\, g^{\sigma\mu}$

Out[∘]= $(h_{\mu\nu} + \eta_{\mu\nu})\ (-h^{\sigma\mu} + \eta^{\sigma\mu})$

Out[∘]= $-h_{\mu\nu}\, h^{\sigma\mu} - h^{\sigma\mu}\, \eta_{\mu\nu} + h_{\mu\nu}\, \eta^{\sigma\mu} + \eta_{\mu\nu}\, \eta^{\sigma\mu}$

Simplify metric and neglect h products:

Out[∘]= $-h_{\mu\nu}\, h^{\sigma\mu} + \eta^{\sigma}{}_{\nu}$

Out[∘]= $\eta^{\sigma}{}_{\nu}$

But this is the Kronecker delta:

Out[∘]= $\eta_{\mu\nu}\, \eta^{\sigma\mu} == \delta^{\sigma}{}_{\nu}$

Out[∘]= $\eta^{\sigma}{}_{\nu} == \delta^{\sigma}{}_{\nu}$

Out[∘]= True

Our *ansatz* for $g^{\mu\nu}$ works up to the first order!

We will need the (approximated) Christoffel symbols of the second kind $\Gamma^{\mu}{}_{\nu\sigma}$ associated with the metric $g_{\mu\nu}$:

```
In[•]:= Print["Definition"]
        Γudd[μ, ν, σ] ==
         1 / 2 guu[μ, β] (PartialD[gdd[σ, β], ν] + PartialD[gdd[ν, β], σ] - PartialD[gdd[ν, σ], β])
        Print["Substituting and expanding"]
        %%[[2]] /. {
          guu[a_, b_] → ηuu[a, b] - huu[a, b],
          PartialD[gdd[a_, b_], c_] → PartialD[ηdd[a, b] + hdd[a, b], c]}
        % // Expand
        Print["We neglect h products and η derivatives are zero:"]
        %% /. HoldPattern[Tensor[h, __] × Tensor[h, __]] → 0
        % /. Tensor[η, _, List[__, Dif[_]]] → 0
        Print["Switch to plain notation and simplify metric"]
        %% // ToNonDifTensor[h]
        % // MetricSimplify[η] // Simplify
        Print["Switching back to ordinary notation gets eq.(5.2),p.171"]
        Γudd[μ, ν, σ] == (%% // ToDifTensor[h, 3])
        Print["Turn into a rule"]
        eqn[5, 2] = Rule @@ %% // LHSSymbolsToPatterns[{μ, ν, σ}]
```

Definition

Out[•]= $\Gamma^{\mu}{}_{\nu\sigma} == \frac{1}{2} g^{\mu\beta} (g_{\nu\beta,\sigma} - g_{\nu\sigma,\beta} + g_{\sigma\beta,\nu})$

Substituting and expanding

Out[•]= $\frac{1}{2} (-h^{\mu\beta} + \eta^{\mu\beta}) (h_{\nu\beta,\sigma} - h_{\nu\sigma,\beta} + h_{\sigma\beta,\nu} + \eta_{\nu\beta,\sigma} - \eta_{\nu\sigma,\beta} + \eta_{\sigma\beta,\nu})$

Out[•]= $-\frac{1}{2} h^{\mu\beta} h_{\nu\beta,\sigma} + \frac{1}{2} h^{\mu\beta} h_{\nu\sigma,\beta} - \frac{1}{2} h^{\mu\beta} h_{\sigma\beta,\nu} + \frac{1}{2} h_{\nu\beta,\sigma} \eta^{\mu\beta} - \frac{1}{2} h_{\nu\sigma,\beta} \eta^{\mu\beta} + \frac{1}{2} h_{\sigma\beta,\nu} \eta^{\mu\beta} - \frac{1}{2} h^{\mu\beta} \eta_{\nu\beta,\sigma} + \frac{1}{2} \eta^{\mu\beta} \eta_{\nu\beta,\sigma} + \frac{1}{2} h^{\mu\beta} \eta_{\nu\sigma,\beta} - \frac{1}{2} \eta^{\mu\beta} \eta_{\nu\sigma,\beta} - \frac{1}{2} h^{\mu\beta} \eta_{\sigma\beta,\nu} + \frac{1}{2} \eta^{\mu\beta} \eta_{\sigma\beta,\nu}$

We neglect h products and η derivatives are zero:

Out[•]= $\frac{1}{2} h_{\nu\beta,\sigma} \eta^{\mu\beta} - \frac{1}{2} h_{\nu\sigma,\beta} \eta^{\mu\beta} + \frac{1}{2} h_{\sigma\beta,\nu} \eta^{\mu\beta} - \frac{1}{2} h^{\mu\beta} \eta_{\nu\beta,\sigma} + \frac{1}{2} \eta^{\mu\beta} \eta_{\nu\beta,\sigma} + \frac{1}{2} h^{\mu\beta} \eta_{\nu\sigma,\beta} - \frac{1}{2} \eta^{\mu\beta} \eta_{\nu\sigma,\beta} - \frac{1}{2} h^{\mu\beta} \eta_{\sigma\beta,\nu} + \frac{1}{2} \eta^{\mu\beta} \eta_{\sigma\beta,\nu}$

Out[•]= $\frac{1}{2} h_{\nu\beta,\sigma} \eta^{\mu\beta} - \frac{1}{2} h_{\nu\sigma,\beta} \eta^{\mu\beta} + \frac{1}{2} h_{\sigma\beta,\nu} \eta^{\mu\beta}$

Switch to plain notation and simplify metric

Out[•]= $\frac{1}{2} h_{\nu\beta\sigma} \eta^{\mu\beta} - \frac{1}{2} h_{\nu\sigma\beta} \eta^{\mu\beta} + \frac{1}{2} h_{\sigma\beta\nu} \eta^{\mu\beta}$

Out[•]= $\frac{1}{2} (-h_{\nu\sigma}{}^{\mu} + h_{\nu}{}^{\mu}{}_{\sigma} + h_{\sigma}{}^{\mu}{}_{\nu})$

Switching back to ordinary notation gets eq.(5.2),p.171

Out[•]= $\Gamma^{\mu}{}_{\nu\sigma} == \frac{1}{2} (-h_{\nu\sigma}{}^{,\mu} + h_{\nu}{}^{\mu}{}_{,\sigma} + h_{\sigma}{}^{\mu}{}_{,\nu})$

Turn into a rule

Out[∘]= $\Gamma^{\mu}{}_{\nu\_\sigma\_} \to \frac{1}{2}\left(-h_{\nu\sigma}{}^{,\mu} + h_{\nu}{}^{\mu}{}_{,\sigma} + h_{\sigma}{}^{\mu}{}_{,\nu}\right)$

Note: The expression

```
In[∘]:=   Tensor[h, {a, Void, Dif[c], Void}, {Void, b, Void, Dif[d]}] // TraditionalForm
```

Out[∘]//TraditionalForm=

$$h^{a}{}_{b}{}^{,c}{}_{,d}$$

means

```
In[∘]:=   PartialD[labs][hud[a, b], {xd[c], xu[d]}] // TraditionalForm
```

Out[∘]//TraditionalForm=

$$\frac{\partial^2 h^{a}{}_{b}}{\partial x^{d}\, \partial x_{c}}$$

Hence the first two slots of $h^{a}{}_{b}{}^{,c}{}_{,d}$ are symmetric because $h_{\mu\nu}$ is symmetric, the last two slots are symmetric because the partial derivatives are supposed to be commutative. Tensor A^{ik} is called symmetric if $A^{ik} == A^{ki}$; in this case we have $A^{i}{}_{k} == A_{k}{}^{i} == A^{i}_{k}$.

Exercise. Prove that  +  =  , but sometimes not!

Einstein's field equations in covariant form (see FN eq.(3.38), p.113) are...

```
In[∘]:=  (eqn[3, 38, cov] = Rdd[μ, ν] - 1/2 Tensor[R] × gdd[μ, ν] == κ Tdd[μ, ν]) // FrameBox // DisplayForm
```

Out[∘]//DisplayForm=

$$\boxed{-\frac{1}{2} R\, g_{\mu\nu} + R_{\mu\nu} == \kappa\, T_{\mu\nu}}$$

...so on the left hand side we need the *curvature scalar* R and the *Ricci tensor* $R_{\mu\nu}$.

We calculate first the Ricci tensor $R_{\mu\nu}$...

```
In[∘]:= Print["We start from the Riemann tensor"]
        Equal @@ RiemannRule;
        Fold[IndexChange[#2][#1] &, %, {{a, μ}, {b, ν}, {c, α}, {d, β}}]
        Print["Contracting the Riemann tensor to form the Ricci tensor Rμν"]
        %% /. β → α
        % /. Ruddd[α, μ, ν, α] → Rdd[μ, ν]
        Print["Substituting Rμν with eqn[5,2] and throwing out h products"]
        %%[[2]] //. eqn[5, 2] // Expand
        % /. HoldPattern[Tensor[h, __] × Tensor[h, __]] → 0;
        % // Factor
        Print["Converting the partial derivatives to plain notation"]
        %% // ToNonDifTensor[h]
        % /. {PartialD[labs][hddu[a_, b_, c_], xu[d_]] → hddud[a, b, c, d],
          PartialD[labs][hdud[a_, b_, c_], xu[d_]] → hdudd[a, b, c, d]}
        Print["Symmetrizing on last two slots"]
        %% // SymmetrizeSlots[h, 4, {1, {3, 4}}]
        Print["Symmetrizing on first two slots"]
        %% // SymmetrizeSlots[h, 4, {1, {1, 2}}]
```

```
Print["Swap indices in the first and third term"]
MapAt[UpDownSwap[α], %%, {{2, 1}, {2, 3}}]
Print["Switch back to ordinary notation"]
(Ricci = Rdd[μ, ν] == (%% // ToDifTensor[h, 3])) // FrameBox // DisplayForm
Print["Turn into a rule"]
eqn[5, 3] = (Rule @@ Ricci) // LHSSymbolsToPatterns[{μ, ν}]
```

We start from the Riemann tensor

Out[∘]= $R^{\beta}{}_{\mu\nu\alpha} == -\Gamma^{e}{}_{\mu\nu}\,\Gamma^{\beta}{}_{\alpha e} + \Gamma^{e}{}_{\mu\alpha}\,\Gamma^{\beta}{}_{\nu e} + \partial_{x^{\nu}}\Gamma^{\beta}{}_{\mu\alpha} - \partial_{x^{\alpha}}\Gamma^{\beta}{}_{\mu\nu}$

Contracting the Riemann tensor to form the Ricci tensor $R_{\mu\nu}$

Out[∘]= $R^{\alpha}{}_{\mu\nu\alpha} == -\Gamma^{e}{}_{\mu\nu}\,\Gamma^{\alpha}{}_{\alpha e} + \Gamma^{e}{}_{\mu\alpha}\,\Gamma^{\alpha}{}_{\nu e} + \partial_{x^{\nu}}\Gamma^{\alpha}{}_{\mu\alpha} - \partial_{x^{\alpha}}\Gamma^{\alpha}{}_{\mu\nu}$

Out[∘]= $R_{\mu\nu} == -\Gamma^{e}{}_{\mu\nu}\,\Gamma^{\alpha}{}_{\alpha e} + \Gamma^{e}{}_{\mu\alpha}\,\Gamma^{\alpha}{}_{\nu e} + \partial_{x^{\nu}}\Gamma^{\alpha}{}_{\mu\alpha} - \partial_{x^{\alpha}}\Gamma^{\alpha}{}_{\mu\nu}$

Substituting $R_{\mu\nu}$ with eqn[5,2] and throwing out h products

Out[∘]= $$\frac{1}{4} h_{\mu}{}^{e}{}_{,\nu}\, h_{\alpha e}{}^{,\alpha} + \frac{1}{4} h_{\nu}{}^{e}{}_{,\mu}\, h_{\alpha e}{}^{,\alpha} - \frac{1}{4} h_{\mu\nu}{}^{,e}\, h_{\alpha e}{}^{,\alpha} - \frac{1}{4} h_{\alpha}{}^{e}{}_{,\mu}\, h_{\nu e}{}^{,\alpha} - \frac{1}{4} h_{\mu}{}^{e}{}_{,\alpha}\, h_{\nu e}{}^{,\alpha} + \frac{1}{4} h_{\mu\alpha}{}^{,e}\, h_{\nu e}{}^{,\alpha} - \frac{1}{4} h_{\mu}{}^{e}{}_{,\nu}\, h_{e}{}^{\alpha}{}_{,\alpha} - \frac{1}{4} h_{\nu}{}^{e}{}_{,\mu}\, h_{e}{}^{\alpha}{}_{,\alpha} + \frac{1}{4} h_{\mu\nu}{}^{,e}\, h_{e}{}^{\alpha}{}_{,\alpha} + \frac{1}{4} h_{\alpha}{}^{e}{}_{,\mu}\, h_{e}{}^{\alpha}{}_{,\nu} + \frac{1}{4} h_{\mu}{}^{e}{}_{,\alpha}\, h_{e}{}^{\alpha}{}_{,\nu} - \frac{1}{4} h_{\mu\alpha}{}^{,e}\, h_{e}{}^{\alpha}{}_{,\nu} - \frac{1}{4} h_{\mu}{}^{e}{}_{,\nu}\, h_{\alpha}{}^{\alpha}{}_{,e} - \frac{1}{4} h_{\nu}{}^{e}{}_{,\mu}\, h_{\alpha}{}^{\alpha}{}_{,e} + \frac{1}{4} h_{\mu\nu}{}^{,e}\, h_{\alpha}{}^{\alpha}{}_{,e} + \frac{1}{4} h_{\alpha}{}^{e}{}_{,\mu}\, h_{\nu}{}^{\alpha}{}_{,e} + \frac{1}{4} h_{\mu}{}^{e}{}_{,\alpha}\, h_{\nu}{}^{\alpha}{}_{,e} - \frac{1}{4} h_{\mu\alpha}{}^{,e}\, h_{\nu}{}^{\alpha}{}_{,e} - \frac{1}{2}\partial_{x^{\nu}} h_{\mu\alpha}{}^{,\alpha} + \frac{1}{2}\partial_{x^{\alpha}} h_{\mu\nu}{}^{,\alpha} + \frac{1}{2}\partial_{x^{\nu}} h_{\alpha}{}^{\alpha}{}_{,\mu} + \frac{1}{2}\partial_{x^{\nu}} h_{\mu}{}^{\alpha}{}_{,\alpha} - \frac{1}{2}\partial_{x^{\alpha}} h_{\mu}{}^{\alpha}{}_{,\nu} - \frac{1}{2}\partial_{x^{\alpha}} h_{\nu}{}^{\alpha}{}_{,\mu}$$

Out[∘]= $$\frac{1}{2}\left(-\partial_{x^{\nu}} h_{\mu\alpha}{}^{,\alpha} + \partial_{x^{\alpha}} h_{\mu\nu}{}^{,\alpha} + \partial_{x^{\nu}} h_{\alpha}{}^{\alpha}{}_{,\mu} + \partial_{x^{\nu}} h_{\mu}{}^{\alpha}{}_{,\alpha} - \partial_{x^{\alpha}} h_{\mu}{}^{\alpha}{}_{,\nu} - \partial_{x^{\alpha}} h_{\nu}{}^{\alpha}{}_{,\mu}\right)$$

Converting the partial derivatives to plain notation

Out[∘]= $$\frac{1}{2}\left(-\partial_{x^{\nu}} h_{\mu\alpha}{}^{\alpha} + \partial_{x^{\alpha}} h_{\mu\nu}{}^{\alpha} + \partial_{x^{\nu}} h_{\alpha}{}^{\alpha}{}_{\mu} + \partial_{x^{\nu}} h_{\mu}{}^{\alpha}{}_{\alpha} - \partial_{x^{\alpha}} h_{\mu}{}^{\alpha}{}_{\nu} - \partial_{x^{\alpha}} h_{\nu}{}^{\alpha}{}_{\mu}\right)$$

Out[∘]= $$\frac{1}{2}\left(-h_{\mu\alpha}{}^{\alpha}{}_{\nu} + h_{\mu\nu}{}^{\alpha}{}_{\alpha} + h_{\alpha}{}^{\alpha}{}_{\mu\nu} + h_{\mu}{}^{\alpha}{}_{\alpha\nu} - h_{\mu}{}^{\alpha}{}_{\nu\alpha} - h_{\nu}{}^{\alpha}{}_{\mu\alpha}\right)$$

Symmetrizing on last two slots

Out[∘]= $$\frac{1}{2}\left(-h_{\mu\alpha}{}^{\alpha}{}_{\nu} + h_{\mu\nu}{}^{\alpha}{}_{\alpha} + h_{\alpha}{}^{\alpha}{}_{\mu\nu} - h_{\nu}{}^{\alpha}{}_{\alpha\mu}\right)$$

Symmetrizing on first two slots

Out[∘]= $$\frac{1}{2}\left(-h_{\alpha\mu}{}^{\alpha}{}_{\nu} + h_{\mu\nu}{}^{\alpha}{}_{\alpha} + h_{\alpha}{}^{\alpha}{}_{\mu\nu} - h^{\alpha}{}_{\nu\alpha\mu}\right)$$

Swap indices in the first and third term

Out[∘]= $$\frac{1}{2}\left(h_{\mu\nu}{}^{\alpha}{}_{\alpha} + h^{\alpha}{}_{\alpha\mu\nu} - h^{\alpha}{}_{\mu\alpha\nu} - h^{\alpha}{}_{\nu\alpha\mu}\right)$$

Switch back to ordinary notation

Out[∘]//DisplayForm=

$$\boxed{R_{\mu\nu} == \frac{1}{2}\left(h_{\mu\nu}{}^{,\alpha}{}_{,\alpha} + h^{\alpha}{}_{\alpha,\mu,\nu} - h^{\alpha}{}_{\mu,\alpha,\nu} - h^{\alpha}{}_{\nu,\alpha,\mu}\right)}$$

Turn into a rule

Out[∘]= $$R_{\mu\_\nu\_} \to \frac{1}{2}\left(h_{\mu\nu}{}^{,\alpha}{}_{,\alpha} + h^{\alpha}{}_{\alpha,\mu,\nu} - h^{\alpha}{}_{\mu,\alpha,\nu} - h^{\alpha}{}_{\nu,\alpha,\mu}\right)$$

...and then we calculate the curvature scalar R

In[∘]:=
```
Print["We start from the Ricci tensor Rμν"]
Ricci
```

```
Print["Contracting on both indices to produce the curvature scalar R"]
guu[μ, ν] # & /@ %%
Print["Substituting the curvature scalar and using η instead of g"]
%% /. guu[μ, ν] × Rdd[μ, ν] → Tensor[R] /. guu[μ, ν] → ηuu[μ, ν]
Print["Switch to plain notation and contracting"]
%% // ToNonDifTensor[h]
% // MapLevelParts[(# // Expand // MetricSimplify[η]) &, {2, {2, 3}}]
Print["Performing dummy swap on first term"]
MapAt[UpDownSwap[ν], %%, {{2, 2, 1}}]
Print["Performing index change on third term and simplification"]
MapAt[IndexChange[{{ν, α}, {α, ν}}], %%, {{2, 2, 3}}]
Expand /@ %
Print["Index renaming"]
%% // IndexChange[{ν, β}]
Print["Switch back to ordinary notation"]
%% // ToDifTensor[h, 3];
% // FrameBox // DisplayForm
Print["Compare with eq.(5.4),p.171."]
Print["Turn into a rule"]
eqn[5, 4] = Rule @@ %%%%
```

We start from the Ricci tensor $R_{\mu\nu}$

Out[∘]= $R_{\mu\nu} == \frac{1}{2}\left(h_{\mu\nu}{}'^{\alpha}{}_{,\alpha} + h^{\alpha}{}_{\alpha,\mu,\nu} - h^{\alpha}{}_{\mu,\alpha,\nu} - h^{\alpha}{}_{\nu,\alpha,\mu}\right)$

Contracting on both indices to produce the curvature scalar R

Out[∘]= $g^{\mu\nu} R_{\mu\nu} == \frac{1}{2} g^{\mu\nu}\left(h_{\mu\nu}{}'^{\alpha}{}_{,\alpha} + h^{\alpha}{}_{\alpha,\mu,\nu} - h^{\alpha}{}_{\mu,\alpha,\nu} - h^{\alpha}{}_{\nu,\alpha,\mu}\right)$

Substituting the curvature scalar and using η instead of g

Out[∘]= $R == \frac{1}{2}\left(h_{\mu\nu}{}'^{\alpha}{}_{,\alpha} + h^{\alpha}{}_{\alpha,\mu,\nu} - h^{\alpha}{}_{\mu,\alpha,\nu} - h^{\alpha}{}_{\nu,\alpha,\mu}\right)\eta^{\mu\nu}$

Switch to plain notation and contracting

Out[∘]= $R == \frac{1}{2}\left(h_{\mu\nu}{}^{\alpha}{}_{\alpha} + h^{\alpha}{}_{\alpha\mu\nu} - h^{\alpha}{}_{\mu\alpha\nu} - h^{\alpha}{}_{\nu\alpha\mu}\right)\eta^{\mu\nu}$

Out[∘]= $R == \frac{1}{2}\left(-h^{\alpha}{}_{\nu\alpha}{}^{\nu} + h^{\alpha}{}_{\alpha}{}^{\nu}{}_{\nu} - h^{\alpha\nu}{}_{\alpha\nu} + h^{\nu}{}_{\nu}{}^{\alpha}{}_{\alpha}\right)$

Performing dummy swap on first term

Out[∘]= $R == \frac{1}{2}\left(h^{\alpha}{}_{\alpha}{}^{\nu}{}_{\nu} - 2\, h^{\alpha\nu}{}_{\alpha\nu} + h^{\nu}{}_{\nu}{}^{\alpha}{}_{\alpha}\right)$

Performing index change on third term and simplification

Out[∘]= $R == \frac{1}{2}\left(2\, h^{\alpha}{}_{\alpha}{}^{\nu}{}_{\nu} - 2\, h^{\alpha\nu}{}_{\alpha\nu}\right)$

Out[∘]= $R == h^{\alpha}{}_{\alpha}{}^{\nu}{}_{\nu} - h^{\alpha\nu}{}_{\alpha\nu}$

Index renaming

Out[∘]= $R == h^{\alpha}{}_{\alpha}{}^{\beta}{}_{\beta} - h^{\alpha\beta}{}_{\alpha\beta}$

Switch back to ordinary notation

Out[ ]//DisplayForm=

$$R == h^{\alpha}{}_{\alpha}{}^{,\beta}{}_{,\beta} - h^{\alpha\beta}{}_{,\alpha,\beta}$$

Compare with eq.(5.4),p.171.

Turn into a rule

Out[ ]= $R \to h^{\alpha}{}_{\alpha}{}^{,\beta}{}_{,\beta} - h^{\alpha\beta}{}_{,\alpha,\beta}$

Note: I put indices in standard order and in up-down configuration, I don't simplify $h^{\mu}{}_{\mu} == h$. This explains (hopefully) slight differences between my results and FN.

Substituting our expressions for the Ricci tensor and curvature scalar into Einstein's field equations in covariant form and using η for g we obtain:

In[ ]:=
```
eqn[3, 38, cov]
% /. eqn[5, 3] /. eqn[5, 4] /. gdd[μ, ν] → ηdd[μ, ν]
Distribute[2 #] & /@ %;
eqn[A] = (%[[1]] // FullSimplify) == %[[2]];
% // FrameBox // DisplayForm
Print["I will call this result eq.(A).\nCompare
   with the equation immediately after eq.(5.4),p.171."]
```

Out[ ]= $-\frac{1}{2} R\, g_{\mu\nu} + R_{\mu\nu} == \kappa\, T_{\mu\nu}$

Out[ ]= $\frac{1}{2}\left(h_{\mu\nu}{}^{,\alpha}{}_{,\alpha} + h^{\alpha}{}_{\alpha,\mu,\nu} - h^{\alpha}{}_{\mu,\alpha,\nu} - h^{\alpha}{}_{\nu,\alpha,\mu}\right) - \frac{1}{2}\left(h^{\alpha}{}_{\alpha}{}^{,\beta}{}_{,\beta} - h^{\alpha\beta}{}_{,\alpha,\beta}\right) \eta_{\mu\nu} == \kappa\, T_{\mu\nu}$

Out[ ]//DisplayForm=

$$h_{\mu\nu}{}^{,\alpha}{}_{,\alpha} + h^{\alpha}{}_{\alpha,\mu,\nu} - h^{\alpha}{}_{\mu,\alpha,\nu} - h^{\alpha}{}_{\nu,\alpha,\mu} + \left(-h^{\alpha}{}_{\alpha}{}^{,\beta}{}_{,\beta} + h^{\alpha\beta}{}_{,\alpha,\beta}\right) \eta_{\mu\nu} == 2\, \kappa\, T_{\mu\nu}$$

I will call this result eq.(A).
Compare with the equation immediately after eq.(5.4),p.171.

■ We can simplify eq.(A) using the following substitution eq.(5.6), p.171, which leads to eq.(5.5), p.171:

In[ ]:=
```
Print["Simplifying substitution eq.(5.6),p.171:"]
eqn[5, 6] = Hold[hbdd[μ, ν]] == hdd[μ, ν] - 1 / 2 hud[σ, σ] × ηdd[μ, ν];
% // ReleaseHold // FrameBox // DisplayForm
Print["Trace-reversed perturbation\nTurn into a rule"]
eqn[5, 6, r] = (Rule @@ eqn[5, 6] // LHSSymbolsToPatterns[{μ, ν}])
```

Simplifying substitution eq.(5.6),p.171:

Out[ ]//DisplayForm=

$$\bar{h}_{\mu\nu} == h_{\mu\nu} - \frac{1}{2} h^{\sigma}{}_{\sigma}\, \eta_{\mu\nu}$$

Trace-reversed perturbation
Turn into a rule

Out[ ]= Hold[hbdd[μ_, ν_]] → $h_{\mu\nu} - \frac{1}{2} h^{\sigma}{}_{\sigma}\, \eta_{\mu\nu}$

We show now that eq.(5.5), p.171, transforms into eq.(A) when the substitution eq.(5.6), p.171, is applied.

In[ ]:=
```
Print["We start from eq.(5.5),p.171:"]
eqn[5, 5] = Tensor[hb, {Void, Void, Void, Dif[α]}, {μ, ν, Dif[α], Void}] + (
```

```
      ηdd[μ, ν] × PartialD[hbuu[α, β], {α, β}] -
       PartialD[hbud[α, ν], {α, μ}] - PartialD[hbud[α, μ], {α, ν}]) == 2 κ Tdd[μ, ν];
% // FrameBox // DisplayForm
Print["For convenience, all indices are lowered using η"]
ηuu[a, α] × PartialD[Hold[hbdd[μ, ν]], {α, a}] +
   (ηdd[μ, ν] × ηuu[b, α] × ηuu[c, β] × PartialD[Hold[hbdd[b, c]], {α, β}] -
     ηuu[d, α] × PartialD[Hold[hbdd[d, ν]], {μ, α}] -
     ηuu[f, α] × PartialD[Hold[hbdd[f, μ]], {ν, α}]) == 2 κ Tdd[μ, ν];
ReleaseHold[%]
Print["Now we apply the substitution eq.(5.6), p.171, ",
 eqn[5, 6, r] // ReleaseHold, ":"]
%%% /. eqn[5, 6, r]
Print["η derivatives are zero"]
%% /. Tensor[η, _, List[__, Dif[_]]] → 0 // Expand
Print["Switch to plain notation and contracting"]
%% // ToNonDifTensor[h]
% // MetricSimplify[η]
Print["Symmetrizing on last two slots"]
%% // SymmetrizeSlots[h, 4, {1, {3, 4}}]
Print["Performing dummy swap on first and sixth term"]
MapAt[UpDownSwap[α], %%, {{1, 1}, {1, 6}}]
Print["Index changing in forth and seventh term"]
MapAt[IndexChange[{σ, α}], %%, {{1, 4}, {1, 7}}];
MapAt[IndexChange[{{α, β}, {σ, α}}], %, {1, 7}];
MapAt[FullSimplify, %, {1}]
Print["Switch back to ordinary notation"]
%% // ToDifTensor[h, 3]
Print["Is this result identical to eq.(A)?"]
%% == eqn[A]
```

We start from eq.(5.5),p.171:

Out[∘]//DisplayForm=

$$\boxed{\bar{h}_{\mu\nu,\alpha'}{}^{\alpha} - \bar{h}^{\alpha}{}_{\mu,\alpha,\nu} - \bar{h}^{\alpha}{}_{\nu,\alpha,\mu} + \bar{h}^{\alpha\beta}{}_{,\alpha,\beta}\,\eta_{\mu\nu} == 2\,\kappa\,T_{\mu\nu}}$$

For convenience, all indices are lowered using η

Out[∘]= $$\bar{h}_{\mu\nu,\alpha,a}\,\eta^{a\alpha} - \bar{h}_{d\nu,\mu,\alpha}\,\eta^{d\alpha} - \bar{h}_{f\mu,\nu,\alpha}\,\eta^{f\alpha} + \bar{h}_{bc,\alpha,\beta}\,\eta^{b\alpha}\,\eta^{c\beta}\,\eta_{\mu\nu} == 2\,\kappa\,T_{\mu\nu}$$

Now we apply the substitution eq.(5.6), p.171, $\bar{h}_{\mu\_\nu\_} \to h_{\mu\nu} - \frac{1}{2} h^{\sigma}{}_{\sigma}\,\eta_{\mu\nu}$:

Out[∘]= $$\eta^{b\alpha}\,\eta^{c\beta}\,\eta_{\mu\nu}\,(h_{bc,\alpha,\beta} + \frac{1}{2}\,(-h^{\sigma}{}_{\sigma,\alpha,\beta}\,\eta_{bc} - h^{\sigma}{}_{\sigma,\beta}\,\eta_{bc,\alpha} - h^{\sigma}{}_{\sigma,\alpha}\,\eta_{bc,\beta} - h^{\sigma}{}_{\sigma}\,\eta_{bc,\alpha,\beta})) -$$
$$\eta^{d\alpha}\,(h_{d\nu,\mu,\alpha} + \frac{1}{2}\,(-h^{\sigma}{}_{\sigma,\mu,\alpha}\,\eta_{d\nu} - h^{\sigma}{}_{\sigma,\mu}\,\eta_{d\nu,\alpha} - h^{\sigma}{}_{\sigma,\alpha}\,\eta_{d\nu,\mu} - h^{\sigma}{}_{\sigma}\,\eta_{d\nu,\mu,\alpha})) -$$
$$\eta^{f\alpha}\,(h_{f\mu,\nu,\alpha} + \frac{1}{2}\,(-h^{\sigma}{}_{\sigma,\nu,\alpha}\,\eta_{f\mu} - h^{\sigma}{}_{\sigma,\nu}\,\eta_{f\mu,\alpha} - h^{\sigma}{}_{\sigma,\alpha}\,\eta_{f\mu,\nu} - h^{\sigma}{}_{\sigma}\,\eta_{f\mu,\nu,\alpha})) +$$
$$\eta^{a\alpha}\,(h_{\mu\nu,\alpha,a} + \frac{1}{2}\,(-h^{\sigma}{}_{\sigma,\alpha,a}\,\eta_{\mu\nu} - h^{\sigma}{}_{\sigma,\alpha}\,\eta_{\mu\nu,a} - h^{\sigma}{}_{\sigma,a}\,\eta_{\mu\nu,\alpha} - h^{\sigma}{}_{\sigma}\,\eta_{\mu\nu,\alpha,a})) == 2\,\kappa\,T_{\mu\nu}$$

η derivatives are zero

Out[∘]= $$h_{\mu\nu,\alpha,a}\,\eta^{a\alpha} - h_{d\nu,\mu,\alpha}\,\eta^{d\alpha} - h_{f\mu,\nu,\alpha}\,\eta^{f\alpha} + \frac{1}{2}\,h^{\sigma}{}_{\sigma,\mu,\alpha}\,\eta^{d\alpha}\,\eta_{d\nu} + \frac{1}{2}\,h^{\sigma}{}_{\sigma,\nu,\alpha}\,\eta^{f\alpha}\,\eta_{f\mu} -$$

$$\frac{1}{2} h^{\sigma}{}_{\sigma,\alpha,a}\, \eta^{a\alpha}\, \eta_{\mu\nu} + h_{bc,\alpha,\beta}\, \eta^{b\alpha}\, \eta^{c\beta}\, \eta_{\mu\nu} - \frac{1}{2} h^{\sigma}{}_{\sigma,\alpha,\beta}\, \eta^{b\alpha}\, \eta^{c\beta}\, \eta_{bc}\, \eta_{\mu\nu} == 2\,\kappa\, T_{\mu\nu}$$

Switch to plain notation and contracting

Out[•]= $$h_{\mu\nu\alpha a}\, \eta^{a\alpha} - h_{d\nu\mu\alpha}\, \eta^{d\alpha} - h_{f\mu\nu\alpha}\, \eta^{f\alpha} + \frac{1}{2} h^{\sigma}{}_{\sigma\mu\alpha}\, \eta^{d\alpha}\, \eta_{d\nu} + \frac{1}{2} h^{\sigma}{}_{\sigma\nu\alpha}\, \eta^{f\alpha}\, \eta_{f\mu} - \frac{1}{2} h^{\sigma}{}_{\sigma\alpha a}\, \eta^{a\alpha}\, \eta_{\mu\nu} + h_{bc\alpha\beta}\, \eta^{b\alpha}\, \eta^{c\beta}\, \eta_{\mu\nu} - \frac{1}{2} h^{\sigma}{}_{\sigma\alpha\beta}\, \eta^{b\alpha}\, \eta^{c\beta}\, \eta_{bc}\, \eta_{\mu\nu} == 2\,\kappa\, T_{\mu\nu}$$

Out[•]= $$h_{\mu\nu\alpha}{}^{\alpha} - h^{\alpha}{}_{\mu\nu\alpha} - h^{\alpha}{}_{\nu\mu\alpha} + \frac{1}{2} h^{\sigma}{}_{\sigma\mu\nu} + \frac{1}{2} h^{\sigma}{}_{\sigma\nu\mu} + h^{\alpha\beta}{}_{\alpha\beta}\, \eta_{\mu\nu} - \frac{1}{2} h^{\sigma}{}_{\sigma\alpha}{}^{\alpha}\, \eta_{\mu\nu} - \frac{1}{2} h^{\sigma}{}_{\sigma}{}^{\beta}{}_{\beta}\, \eta_{\mu\nu} == 2\,\kappa\, T_{\mu\nu}$$

Symmetrizing on last two slots

Out[•]= $$h_{\mu\nu\alpha}{}^{\alpha} - h^{\alpha}{}_{\mu\alpha\nu} - h^{\alpha}{}_{\nu\alpha\mu} + h^{\sigma}{}_{\sigma\mu\nu} + h^{\alpha\beta}{}_{\alpha\beta}\, \eta_{\mu\nu} - \frac{1}{2} h^{\sigma}{}_{\sigma\alpha}{}^{\alpha}\, \eta_{\mu\nu} - \frac{1}{2} h^{\sigma}{}_{\sigma}{}^{\beta}{}_{\beta}\, \eta_{\mu\nu} == 2\,\kappa\, T_{\mu\nu}$$

Performing dummy swap on first and sixth term

Out[•]= $$h_{\mu\nu}{}^{\alpha}{}_{\alpha} - h^{\alpha}{}_{\mu\alpha\nu} - h^{\alpha}{}_{\nu\alpha\mu} + h^{\sigma}{}_{\sigma\mu\nu} + h^{\alpha\beta}{}_{\alpha\beta}\, \eta_{\mu\nu} - \frac{1}{2} h^{\sigma}{}_{\sigma}{}^{\alpha}{}_{\alpha}\, \eta_{\mu\nu} - \frac{1}{2} h^{\sigma}{}_{\sigma}{}^{\beta}{}_{\beta}\, \eta_{\mu\nu} == 2\,\kappa\, T_{\mu\nu}$$

Index changing in forth and seventh term

Out[•]= $$h_{\mu\nu}{}^{\alpha}{}_{\alpha} + h^{\alpha}{}_{\alpha\mu\nu} - h^{\alpha}{}_{\mu\alpha\nu} - h^{\alpha}{}_{\nu\alpha\mu} + \left(-h^{\alpha}{}_{\alpha}{}^{\beta}{}_{\beta} + h^{\alpha\beta}{}_{\alpha\beta}\right) \eta_{\mu\nu} == 2\,\kappa\, T_{\mu\nu}$$

Switch back to ordinary notation

Out[•]= $$h_{\mu\nu}{}^{\prime\alpha}{}_{,\alpha} + h^{\alpha}{}_{\alpha,\mu,\nu} - h^{\alpha}{}_{\mu,\alpha,\nu} - h^{\alpha}{}_{\nu,\alpha,\mu} + \left(-h^{\alpha}{}_{\alpha}{}^{\prime\beta}{}_{,\beta} + h^{\alpha\beta}{}_{,\alpha,\beta}\right) \eta_{\mu\nu} == 2\,\kappa\, T_{\mu\nu}$$

Is this result identical to eq.(A)?

Out[•]= True

Since we can also reverse the order of the derivation, we have proven that eq.(A) and eq.(5.5) are equivalent via the substitution eq.(5.6).

For future use, we invert eq.(5.6) to get the backward transformation (see Exercise 5.1.2, p.173).

In[•]:=
```
eq = eqn[5, 6] // ReleaseHold
Print["Multiply with η^μν and metric simplify"]
Distribute[ηuu[μ, ν] #] & /@ %%
% // MetricSimplify[η]
Print["Sum η_ν^ν factor and simplify"]
%%%[[2, 2, {3, 4}]];
% == (% // MetricSimplify[η]) == (% // EinsteinSum[]) /. TensorValueRules[η]
%%%% /. Rule @@ %[[{2, 3}]]
MapAt[SimplifyTensorSum, %, 2]
Print["Invert and turn into a rule"]
hr = -%%[[2]] → -%%[[1]] // LHSSymbolsToPatterns[{ν}];
% // FrameBox // DisplayForm
Print[
 "Substituting this result in eq.(5.6) leads finally to the backward transformation."]
eq
% // Reverse;
(# - %[[1, 2]]) & /@ %
% /. hr // FrameBox // DisplayForm
Print["See Solution to Exercise 5.1.2,p.273"]
```

Out[•]= $\bar{h}_{\mu\nu} == h_{\mu\nu} - \frac{1}{2} h^{\sigma}{}_{\sigma}\, \eta_{\mu\nu}$

Multiply with $\eta^{\mu\nu}$ and metric simplify

Out[•]= $\bar{h}_{\mu\nu}\, \eta^{\mu\nu} == h_{\mu\nu}\, \eta^{\mu\nu} - \frac{1}{2} h^{\sigma}{}_{\sigma}\, \eta_{\mu\nu}\, \eta^{\mu\nu}$

Out[•]= $\bar{h}^{\nu}{}_{\nu} == h^{\nu}{}_{\nu} - \frac{1}{2} h^{\sigma}{}_{\sigma}\, \eta_{\nu}{}^{\nu}$

Sum $\eta_{\nu}{}^{\nu}$ factor and simplify

Out[•]= $\eta_{\mu\nu}\, \eta^{\mu\nu} == \eta_{\nu}{}^{\nu} == 4$

Out[•]= $\bar{h}^{\nu}{}_{\nu} == h^{\nu}{}_{\nu} - 2\, h^{\sigma}{}_{\sigma}$

Out[•]= $\bar{h}^{\nu}{}_{\nu} == - h^{\nu}{}_{\nu}$

Invert and turn into a rule

Out[•]//DisplayForm=

$$\boxed{h^{\nu_{-}}{}_{\nu_{-}} \to -\bar{h}^{\nu}{}_{\nu}}$$

Substituting this result in eq.(5.6) leads finally to the backward transformation.

Out[•]= $\bar{h}_{\mu\nu} == h_{\mu\nu} - \frac{1}{2} h^{\sigma}{}_{\sigma}\, \eta_{\mu\nu}$

Out[•]= $h_{\mu\nu} == \bar{h}_{\mu\nu} + \frac{1}{2} h^{\sigma}{}_{\sigma}\, \eta_{\mu\nu}$

Out[•]//DisplayForm=

$$\boxed{h_{\mu\nu} == \bar{h}_{\mu\nu} - \frac{1}{2} \bar{h}^{\sigma}{}_{\sigma}\, \eta_{\mu\nu}}$$

See Solution to Exercise 5.1.2,p.273

■ We can in turn simplify eq.(5.5) by means of a *gauge transformation*. A gauge transformation is a small change of coordinates defined by

In[•]:=
```
xu[red@μ] == xu[μ] + ξu[μ];
% // FrameBox // DisplayForm
Print["Gauge transformation\nTurn into a rule"]
eqn[5, 7] = Rule @@ %%%
Print["Inverse transformation as rule"]
eqn[5, 7, inv] = Solve[%%%%%, xu[μ]][[1, 1]]
```

Out[•]//DisplayForm=

$$\boxed{x^{\mu'} == x^{\mu} + \xi^{\mu}}$$

Gauge transformation
Turn into a rule

Out[•]= $x^{\mu'} \to x^{\mu} + \xi^{\mu}$

Inverse transformation as rule

Out[•]= $x^{\mu} \to x^{\mu'} - \xi^{\mu}$

where the gauge function $\xi^{\mu} = \xi^{\mu}\,(x^{\alpha})$ is of the same order of smallness as the $h_{\mu\nu}$. The gauge transformation takes a *nearly Cartesian coordinate system* into another *nearly Cartesian coordinate system*.

The transformation matrix elements $X^{\mu'}{}_{\nu}$ in expanded form are given by eq.(5.8), p.171:

```
In[•]:= Print["Definition"]
        Xud[red@μ, ν] == PartialD[labs][xu[red@μ], xu[ν]]
        Print["Using the substitution ", eqn[5, 7]]
        %% /. eqn[5, 7]
        % /. PartialD[labs][ξu[μ], xu[ν]] → ξud[μ, Dif[ν]];
        Print["Unexpanded form:"]
        %% // FrameBox // DisplayForm
        Print["Turn into a rule"]
        eqn[5, 8] = Rule @@ %%%% // LHSSymbolsToPatterns[{μ, ν}]
```

Definition

Out[•]= $X^{\mu'}{}_{\nu} == \partial_{x^{\nu}} x^{\mu'}$

Using the substitution $x^{\mu'} \to x^{\mu} + \xi^{\mu}$

Out[•]= $X^{\mu'}{}_{\nu} == \delta^{\mu}{}_{\nu} + \partial_{x^{\nu}} \xi^{\mu}$

Unexpanded form:

Out[•]//DisplayForm=

$$X^{\mu'}{}_{\nu} == \delta^{\mu}{}_{\nu} + \xi^{\mu}{}_{,\nu}$$

Turn into a rule

Out[•]= $X^{\mu\_'}{}_{\nu\_} \to \delta^{\mu}{}_{\nu} + \xi^{\mu}{}_{,\nu}$

We show now that under a wisely chosen gauge transformation each of the last three terms on the left of eq.(5.5) is separately zero and the equation reduces in the resulting new coordinate system to the well-known **wave equation**.

For the following calculations 1) - 4) see also FN Solution to Exercise 5.1.3, p .273.

1) We will need an approximation for the inverse transformation matrix elements $X^{\mu}{}_{\nu'}$ between black&unprimed and red&primed tensors.

```
In[•]:= Print["Definition"]
        Xud[μ, red@ν] == PartialD[labs][xu[μ], xu[red@ν]]
        Print["Proposed ansatz:"]
        Xud[μ, red@ν] == δud[μ, ν] - ξud[μ, Dif[ν]];
        % // FrameBox // DisplayForm
        Print["Turn into a rule"]
        eqn[5, 8, inv] = Rule @@ %%% // LHSSymbolsToPatterns[{μ, ν}]
        Print["We start with"]
        Xud[α, red@β] × Xud[red@β, γ]
        Print["Substitiution"]
        %% /. {eqn[5, 8], eqn[5, 8, inv]}
        % // Expand
        Print["Switch to plain notation and contracting"]
        %% // ToNonDifTensor[ξ]
        % // KroneckerAbsorb[δ]
        Print["Throw out second order terms"]
        %% /. HoldPattern[Tensor[ξ, __] × Tensor[ξ, __]] → 0
```

Definition

Out[•]= $X^{\mu}{}_{\nu'} == \partial_{x^{\nu'}} x^{\mu}$

Proposed *ansatz:*

Out[•]//DisplayForm=

$$\boxed{X^{\mu}{}_{\nu'} == \delta^{\mu}{}_{\nu} - \xi^{\mu}{}_{,\nu}}$$

Turn into a rule

Out[•]= $X^{\mu}{}_{\underline{\nu}'} \to \delta^{\mu}{}_{\nu} - \xi^{\mu}{}_{,\nu}$

We start with

Out[•]= $X^{\alpha}{}_{\beta'} X^{\beta'}{}_{\gamma}$

Substitiution

Out[•]= $(\delta^{\alpha}{}_{\beta} - \xi^{\alpha}{}_{,\beta})\ (\delta^{\beta}{}_{\gamma} + \xi^{\beta}{}_{,\gamma})$

Out[•]= $\delta^{\alpha}{}_{\beta}\,\delta^{\beta}{}_{\gamma} - \delta^{\beta}{}_{\gamma}\,\xi^{\alpha}{}_{,\beta} + \delta^{\alpha}{}_{\beta}\,\xi^{\beta}{}_{,\gamma} - \xi^{\alpha}{}_{,\beta}\,\xi^{\beta}{}_{,\gamma}$

Switch to plain notation and contracting

Out[•]= $\delta^{\alpha}{}_{\beta}\,\delta^{\beta}{}_{\gamma} - \delta^{\beta}{}_{\gamma}\,\xi^{\alpha}{}_{\beta} + \delta^{\alpha}{}_{\beta}\,\xi^{\beta}{}_{\gamma} - \xi^{\alpha}{}_{\beta}\,\xi^{\beta}{}_{\gamma}$

Out[•]= $\delta^{\alpha}{}_{\gamma} - \xi^{\alpha}{}_{\beta}\,\xi^{\beta}{}_{\gamma}$

Throw out second order terms

Out[•]= $\delta^{\alpha}{}_{\gamma}$

We got the Kronecker delta: our *ansatz* for $X^{\mu}{}_{\nu'}$ works up to the first order!

2) Starting with eq.(5.9), FN actually transform $g^{\mu\nu}$ and then simplify.

In[•]:=
```
Print["Transformation equation for " guu[red@μ, red@ν]]
guu[red@μ, red@ν] == Xud[red@μ, α] × Xud[red@ν, β] × guu[α, β]
% /. guu[α_, β_] → ηuu[α, β] - huu[α, β]
Print["Making the substitution ", eqn[5, 8]]
%% /. eqn[5, 8]
Print["Expanding and applying the Kronecker deltas"]
%% // ExpandAll;
% // KroneckerAbsorb[δ]
Print["Switch to plain notation and metric simplification"]
%% // ToNonDifTensor[ξ]
% // MetricSimplify[η] // StandardForm
Print["Dropping higher order terms"]
%% /. HoldPattern[Tensor[__] × Tensor[__]] → 0
Print["Switch back to ordinary notation"]
%% // ToDifTensor[ξ, 2]
Print["η is the same in both coordinate systems"]
%% /. ηuu[red@μ, red@ν] → ηuu[μ, ν];
(eqn[5, 9] = -# + ηuu[μ, ν] & /@ %) // FrameBox // DisplayForm
Print["See eq.(5.9),p.171."]
```

Transformation equation for $g^{\mu'\nu'}$

Out[•]= $g^{\mu'\nu'} == g^{\alpha\beta} X^{\mu'}{}_{\alpha} X^{\nu'}{}_{\beta}$

Out[∘]= $-h^{\mu'\nu'}+\eta^{\mu'\nu'} == X^{\mu'}{}_{\alpha}\, X^{\nu'}{}_{\beta}\, (-h^{\alpha\beta}+\eta^{\alpha\beta})$

Making the substitution $X^{\mu\_'}{}_{\nu\_} \to \delta^{\mu}{}_{\nu}+\xi^{\mu}{}_{,\nu}$

Out[∘]= $-h^{\mu'\nu'}+\eta^{\mu'\nu'} == (-h^{\alpha\beta}+\eta^{\alpha\beta})\, (\delta^{\mu}{}_{\alpha}+\xi^{\mu}{}_{,\alpha})\, (\delta^{\nu}{}_{\beta}+\xi^{\nu}{}_{,\beta})$

Expanding and applying the Kronecker deltas

Out[∘]= $-h^{\mu'\nu'}+\eta^{\mu'\nu'} == -h^{\mu\nu}+\eta^{\mu\nu}-h^{\alpha\nu}\,\xi^{\mu}{}_{,\alpha}+\eta^{\alpha\nu}\,\xi^{\mu}{}_{,\alpha}-h^{\mu\beta}\,\xi^{\nu}{}_{,\beta}+\eta^{\mu\beta}\,\xi^{\nu}{}_{,\beta}-h^{\alpha\beta}\,\xi^{\mu}{}_{,\alpha}\,\xi^{\nu}{}_{,\beta}+\eta^{\alpha\beta}\,\xi^{\mu}{}_{,\alpha}\,\xi^{\nu}{}_{,\beta}$

Switch to plain notation and metric simplification

Out[∘]= $-h^{\mu'\nu'}+\eta^{\mu'\nu'} == -h^{\mu\nu}+\eta^{\mu\nu}-h^{\alpha\nu}\,\xi^{\mu}{}_{\alpha}+\eta^{\alpha\nu}\,\xi^{\mu}{}_{\alpha}-h^{\mu\beta}\,\xi^{\nu}{}_{\beta}+\eta^{\mu\beta}\,\xi^{\nu}{}_{\beta}-h^{\alpha\beta}\,\xi^{\mu}{}_{\alpha}\,\xi^{\nu}{}_{\beta}+\eta^{\alpha\beta}\,\xi^{\mu}{}_{\alpha}\,\xi^{\nu}{}_{\beta}$

Out[∘]//StandardForm=

$-h^{\mu'\nu'}+\eta^{\mu'\nu'} == -h^{\mu\nu}+\eta^{\mu\nu}-h^{\alpha\nu}\,\xi^{\mu}{}_{\alpha}+\xi^{\mu\nu}-h^{\mu\beta}\,\xi^{\nu}{}_{\beta}-h^{\alpha\beta}\,\xi^{\mu}{}_{\alpha}\,\xi^{\nu}{}_{\beta}+\xi^{\mu\beta}\,\xi^{\nu}{}_{\beta}+\xi^{\nu\mu}$

Dropping higher order terms

Out[∘]= $-h^{\mu'\nu'}+\eta^{\mu'\nu'} == -h^{\mu\nu}+\eta^{\mu\nu}+\xi^{\mu\nu}+\xi^{\nu\mu}$

Switch back to ordinary notation

Out[∘]= $-h^{\mu'\nu'}+\eta^{\mu'\nu'} == -h^{\mu\nu}+\eta^{\mu\nu}+\xi^{\mu,\nu}+\xi^{\nu,\mu}$

η is the same in both coordinate systems

Out[∘]//DisplayForm=

$$\boxed{h^{\mu'\nu'} == h^{\mu\nu}-\xi^{\mu,\nu}-\xi^{\nu,\mu}}$$

See eq.(5.9),p.171.

3) For eq.(5.10) we lower an index and contract on eq.(5.9).

In[∘]:=
```
eqn[5, 9]
Inner[Times, %, ηdd[red@μ, red@ν] == ηdd[μ, ν], Equal]
% // ExpandAll
Print["Switch to plain notation and metric simplification"]
%% // ToNonDifTensor[ξ]
% // MetricSimplify[η]
Print["Switch back to ordinary notation"]
%% // ToDifTensor[ξ, 2]
Print["Performing dummies swap"]
MapAt[UpDownSwap[red@ν], %%, 1];
MapAt[UpDownSwap[ν], %, {{2, 1}, {2, 2}}]
Print["Index renaming"]
(eqn[5, 10] = %% /. ν → μ) // FrameBox // DisplayForm
Print["See eq.(5.10),p.171."]
```

Out[∘]= $h^{\mu'\nu'} == h^{\mu\nu}-\xi^{\mu,\nu}-\xi^{\nu,\mu}$

Out[∘]= $h^{\mu'\nu'}\,\eta_{\mu'\nu'} == \eta_{\mu\nu}\,(h^{\mu\nu}-\xi^{\mu,\nu}-\xi^{\nu,\mu})$

Out[∘]= $h^{\mu'\nu'}\,\eta_{\mu'\nu'} == h^{\mu\nu}\,\eta_{\mu\nu}-\eta_{\mu\nu}\,\xi^{\mu,\nu}-\eta_{\mu\nu}\,\xi^{\nu,\mu}$

Switch to plain notation and metric simplification

Out[∘]= $h^{\mu'\nu'}\,\eta_{\mu'\nu'} == h^{\mu\nu}\,\eta_{\mu\nu}-\eta_{\mu\nu}\,\xi^{\mu\nu}-\eta_{\mu\nu}\,\xi^{\nu\mu}$

Out[∘]= $h_{\nu'}{}^{\nu'} == h_{\nu}{}^{\nu}-\xi_{\nu}{}^{\nu}-\xi^{\nu}{}_{\nu}$

Switch back to ordinary notation

Out[ ]= $h_{\nu'}{}^{\nu'} == h_{\nu}{}^{\nu} - \xi_{\nu}{}^{,\nu} - \xi^{\nu}{}_{,\nu}$

Performing dummies swap

Out[ ]= $h^{\nu'}{}_{\nu'} == h^{\nu}{}_{\nu} - 2\, \xi^{\nu}{}_{,\nu}$

Index renaming

Out[ ]//DisplayForm=

$$h^{\mu'}{}_{\mu'} == h^{\mu}{}_{\mu} - 2\, \xi^{\mu}{}_{,\mu}$$

See eq.(5.10),p.171.

4) For eq.(5.11), we start with $\bar{h}^{\mu\nu}$ and apply the eq.(5.9) and eq.(5.10):

```
In[ ]:= Print["Definition"]
hbuu[μ, ν] == huu[μ, ν] - 1 / 2 hud[α, α] × ηuu[μ, ν] // ToFlavor[red]
Print["Apply eq.(5.9) and eq.(5,10)"]
%% /. Rule @@ eqn[5, 9]
% /. (Rule @@ eqn[5, 10] /. μ → α)
Print["η is invariant and expand"]
%% /. ηuu[red@μ, red@ν] → ηuu[μ, ν]
% // ExpandAll
Print["The first two terms are the definition of ", hbuu[μ, ν]]
(eqn[5, 11] = %% // MapLevelParts[hbuu[μ, ν] &, {2, {1, 2}}]) // FrameBox // DisplayForm
Print["See eq.(5.11),p.171."]
```

Definition

Out[ ]= $\bar{h}^{\mu'\nu'} == h^{\mu'\nu'} - \frac{1}{2} h^{\alpha'}{}_{\alpha'}\, \eta^{\mu'\nu'}$

Apply eq.(5.9) and eq.(5,10)

Out[ ]= $\bar{h}^{\mu'\nu'} == h^{\mu\nu} - \frac{1}{2} h^{\alpha'}{}_{\alpha'}\, \eta^{\mu'\nu'} - \xi^{\mu,\nu} - \xi^{\nu,\mu}$

Out[ ]= $\bar{h}^{\mu'\nu'} == h^{\mu\nu} - \frac{1}{2} \eta^{\mu'\nu'} (h^{\alpha}{}_{\alpha} - 2\, \xi^{\alpha}{}_{,\alpha}) - \xi^{\mu,\nu} - \xi^{\nu,\mu}$

η is invariant and expand

Out[ ]= $\bar{h}^{\mu'\nu'} == h^{\mu\nu} - \frac{1}{2} \eta^{\mu\nu} (h^{\alpha}{}_{\alpha} - 2\, \xi^{\alpha}{}_{,\alpha}) - \xi^{\mu,\nu} - \xi^{\nu,\mu}$

Out[ ]= $\bar{h}^{\mu'\nu'} == h^{\mu\nu} - \frac{1}{2} h^{\alpha}{}_{\alpha}\, \eta^{\mu\nu} + \eta^{\mu\nu}\, \xi^{\alpha}{}_{,\alpha} - \xi^{\mu,\nu} - \xi^{\nu,\mu}$

The first two terms are the definition of $\bar{h}^{\mu\nu}$

Out[ ]//DisplayForm=

$$\bar{h}^{\mu'\nu'} == \bar{h}^{\mu\nu} + \eta^{\mu\nu}\, \xi^{\alpha}{}_{,\alpha} - \xi^{\mu,\nu} - \xi^{\nu,\mu}$$

See eq.(5.11),p.171.

We need the following derived relation...

```
In[ ]:= eqn[5, 11]
% /. α → β /. ν → α
PartialD[#, α] & /@ %
Print["η derivatives are zero"]
```

```
%% /. Tensor[η, _, List[__, Dif[_]]] → 0
Print["Symmetrizing on first two slots of ", hb]
eqn[5, 11, der] = %% // SymmetrizeSlots[hb, 3, {1, {1, 2}}]
```

Out[•]= $\bar{h}^{\mu'\nu'} == \bar{h}^{\mu\nu} + \eta^{\mu\nu}\,\xi^{\alpha}{}_{,\alpha} - \xi^{\mu,\nu} - \xi^{\nu,\mu}$

Out[•]= $\bar{h}^{\mu'\alpha'} == \bar{h}^{\mu\alpha} - \xi^{\alpha,\mu} + \eta^{\mu\alpha}\,\xi^{\beta}{}_{,\beta} - \xi^{\mu,\alpha}$

Out[•]= $\bar{h}^{\mu'\alpha'}{}_{,\alpha} == \bar{h}^{\mu\alpha}{}_{,\alpha} + \eta^{\mu\alpha}{}_{,\alpha}\,\xi^{\beta}{}_{,\beta} - \xi^{\alpha,\mu}{}_{,\alpha} + \eta^{\mu\alpha}\,\xi^{\beta}{}_{,\beta,\alpha} - \xi^{\mu,\alpha}{}_{,\alpha}$

η derivatives are zero

Out[•]= $\bar{h}^{\mu'\alpha'}{}_{,\alpha} == \bar{h}^{\mu\alpha}{}_{,\alpha} - \xi^{\alpha,\mu}{}_{,\alpha} + \eta^{\mu\alpha}\,\xi^{\beta}{}_{,\beta,\alpha} - \xi^{\mu,\alpha}{}_{,\alpha}$

Symmetrizing on first two slots of $\bar{h}$

Out[•]= $\bar{h}^{\alpha'\mu'}{}_{,\alpha} == \bar{h}^{\alpha\mu}{}_{,\alpha} - \xi^{\alpha,\mu}{}_{,\alpha} + \eta^{\mu\alpha}\,\xi^{\beta}{}_{,\beta,\alpha} - \xi^{\mu,\alpha}{}_{,\alpha}$

5) For eq.(5.12), p.172

```
In[•]:= PartialD[hbuu[α, μ] // ToFlavor[red], red@α] ==
         PartialD[hbuu[α, μ] // ToFlavor[red], β] × Xud[β, red@α]
        Print["Substituting for X, expanding and throwing away second order terms"]
        %% /. eqn[5, 8, inv]
        % // ExpandAll
        % /. HoldPattern[Tensor[hb, __] × Tensor[ξ, __]] → 0
        Print["Applying the Kronecker delta"]
        %% // Expand // KroneckerAbsorb[δ]
        Print["Applying the rule derived from eq.(5.11)"]
        %% /. Rule @@ eqn[5, 11, der]
        Print["Switch partially to plain notation and metric simplification"]
        %% // ToNonDifTensor[ξ]
        % // MetricSimplify[η]
        β → α;
        Print["Switch back to ordinary notation and index renaming ", %]
        %%% // ToDifTensor[ξ, 2]
        % /. %%%
        Print["Symmetrizing on last two slots of ξ and simplifying"]
        eqn[5, 12] = %% // SymmetrizeSlots[ξ, 3, {1, {2, 3}}]
        Print["Compare with eq.(5.12),p.172."]
```

Out[•]= $\bar{h}^{\alpha'\mu'}{}_{,\alpha'} == \bar{h}^{\alpha'\mu'}{}_{,\beta}\,X^{\beta}{}_{\alpha'}$

Substituting for X, expanding and throwing away second order terms

Out[•]= $\bar{h}^{\alpha'\mu'}{}_{,\alpha'} == \bar{h}^{\alpha'\mu'}{}_{,\beta}\,(\delta^{\beta}{}_{\alpha} - \xi^{\beta}{}_{,\alpha})$

Out[•]= $\bar{h}^{\alpha'\mu'}{}_{,\alpha'} == \bar{h}^{\alpha'\mu'}{}_{,\beta}\,\delta^{\beta}{}_{\alpha} - \bar{h}^{\alpha'\mu'}{}_{,\beta}\,\xi^{\beta}{}_{,\alpha}$

Out[•]= $\bar{h}^{\alpha'\mu'}{}_{,\alpha'} == \bar{h}^{\alpha'\mu'}{}_{,\beta}\,\delta^{\beta}{}_{\alpha}$

Applying the Kronecker delta

Out[•]= $\bar{h}^{\alpha'\mu'}{}_{,\alpha'} == \bar{h}^{\alpha'\mu'}{}_{,\alpha}$

Applying the rule derived from eq.(5.11)

Out[•]= $\bar{h}^{\alpha'\mu'}{}_{,\alpha'} == \bar{h}^{\alpha\mu}{}_{,\alpha} - \xi^{\alpha,\mu}{}_{,\alpha} + \eta^{\mu\alpha}\,\xi^{\beta}{}_{,\beta,\alpha} - \xi^{\mu,\alpha}{}_{,\alpha}$

Switch partially to plain notation and metric simplification

Out[•]= $\bar{h}^{\alpha'\mu'}{}_{,\alpha'} == \bar{h}^{\alpha\mu}{}_{,\alpha} - \xi^{\alpha\mu}{}_{\alpha} + \eta^{\mu\alpha}\,\xi^{\beta}{}_{\beta\alpha} - \xi^{\mu\alpha}{}_{\alpha}$

Out[•]= $\bar{h}^{\alpha'\mu'}{}_{,\alpha'} == \bar{h}^{\alpha\mu}{}_{,\alpha} - \xi^{\alpha\mu}{}_{\alpha} + \xi^{\beta}{}_{\beta}{}^{\mu} - \xi^{\mu\alpha}{}_{\alpha}$

```
Switch back to ordinary notation and index renaming β→α
```

Out[•]= $\bar{h}^{\alpha'\mu'}{}_{,\alpha'} == \bar{h}^{\alpha\mu}{}_{,\alpha} - \xi^{\alpha,\mu}{}_{,\alpha} + \xi^{\beta}{}_{,\beta}{}'^{\mu} - \xi^{\mu,\alpha}{}_{,\alpha}$

Out[•]= $\bar{h}^{\alpha'\mu'}{}_{,\alpha'} == \bar{h}^{\alpha\mu}{}_{,\alpha} + \xi^{\alpha}{}_{,\alpha}{}'^{\mu} - \xi^{\alpha,\mu}{}_{,\alpha} - \xi^{\mu,\alpha}{}_{,\alpha}$

```
Symmetrizing on last two slots of ξ and simplifying
```

Out[•]= $\bar{h}^{\alpha'\mu'}{}_{,\alpha'} == \bar{h}^{\alpha\mu}{}_{,\alpha} - \xi^{\mu,\alpha}{}_{,\alpha}$

```
Compare with eq.(5.12),p.172.
```

If we therefore choose ξ^{μ} to be the solution of...

In[•]:=
```
(eqn[5, 12][[2]] == 0) // FrameBox // DisplayForm
Print["See eq.(5.13),p.172."]
```

Out[•]//DisplayForm=

$\bar{h}^{\alpha\mu}{}_{,\alpha} - \xi^{\mu,\alpha}{}_{,\alpha} == 0$

```
See eq.(5.13),p.172.
```

...then, using the corresponding gauge transformation, the right hand side of eq.(5.12) is zero:

In[•]:=
```
(eqn[5, 15, red] = eqn[5, 12][[1]] == 0) // FrameBox // DisplayForm
Print["(Harmonic) gauge condition.\nCompare with eq.(5.15),p.172."]
```

Out[•]//DisplayForm=

$\bar{h}^{\alpha'\mu'}{}_{,\alpha'} == 0$

```
(Harmonic) gauge condition.
Compare with eq.(5.15),p.172.
```

With the *harmonic gauge condition*, any term of the following form will be zero because the derivative of a constant is zero:

In[•]:=
```
Print["Differentiating the gauge condition"]
PartialD[#, {red@β}] & /@ eqn[5, 15, red]
Print["Turn into a rule"]
r1 = Rule @@ %% /. μ | β → Blank[]
```

```
Differentiating the gauge condition
```

Out[•]= $\bar{h}^{\alpha'\mu'}{}_{,\alpha',\beta'} == 0$

```
Turn into a rule
```

Out[•]= $\bar{h}^{\alpha'\_'}{}_{,\alpha',\_'} \to 0$

The new coordinate system $x^{\mu'}$ in which the harmonic gauge condition holds is sometimes called *harmonic coordinate system* or *harmonic gauge*.

We can lower the free index in $\bar{h}^{\alpha'\mu'}{}_{,\alpha'}$ getting a similar relation:

In[•]:=
```
Print["Lower free index"]
ηdd[red@a, red@μ] (eqn[5, 15, red][[1]] /. μ → a);
ToDifTensor[hb, 3][MetricSimplify[η][ToNonDifTensor[hb][%]]];
% == %%
```

```
Print["Differentiating"]
PartialD[#, {red@β}] & /@ %%[[{1, 2}]]
Print["η derivatives are zero"]
%% /. Tensor[η, _, List[__, Dif[_]]] → 0
Print["The gauge condition implies the relation"]
%% /. r1
Print["Turn into a rule"]
r2 = Rule @@ %% /. μ | β → Blank[]
```

Lower free index

Out[•]= $\bar{h}^{\alpha'}{}_{\mu',\alpha'} == \bar{h}^{\alpha' a'}{}_{,\alpha'}\, \eta_{a'\mu'}$

Differentiating

Out[•]= $\bar{h}^{\alpha'}{}_{\mu',\alpha',\beta'} == \bar{h}^{\alpha' a'}{}_{,\alpha',\beta'}\, \eta_{a'\mu'} + \bar{h}^{\alpha' a'}{}_{,\alpha'}\, \eta_{a'\mu',\beta'}$

η derivatives are zero

Out[•]= $\bar{h}^{\alpha'}{}_{\mu',\alpha',\beta'} == \bar{h}^{\alpha' a'}{}_{,\alpha',\beta'}\, \eta_{a'\mu'}$

The gauge condition implies the relation

Out[•]= $\bar{h}^{\alpha'}{}_{\mu',\alpha',\beta'} == 0$

Turn into a rule

Out[•]= $\bar{h}^{\alpha'}{}_{\_',\alpha',\_'} \to 0$

```
In[•]:= Print["So eq.(5.5) in the red&primed coordinate system"]
eqn[5, 5] // ToFlavor[red]
{r1, r2};
Print["with ", %, " simplyfies to"]
eq = %%% /. %%
```

So eq.(5.5) in the red&primed coordinate system

Out[•]= $\bar{h}_{\mu'\nu',\alpha'}{}^{,\alpha'} - \bar{h}^{\alpha'}{}_{\mu',\alpha',\nu'} - \bar{h}^{\alpha'}{}_{\nu',\alpha',\mu'} + \bar{h}^{\alpha'\beta'}{}_{,\alpha',\beta'}\, \eta_{\mu'\nu'} == 2\,\kappa\, T_{\mu'\nu'}$

with $\{\bar{h}^{\alpha'\_'}{}_{,\alpha',\_'} \to 0,\ \bar{h}^{\alpha'}{}_{\_',\alpha',\_'} \to 0\}$ simplyfies to

Out[•]= $\bar{h}_{\mu'\nu',\alpha'}{}^{,\alpha'} == 2\,\kappa\, T_{\mu'\nu'}$

By simplifying the notation dropping primes (and color) we get finally

```
In[•]:= SequenceForm[eqn[5, 14] = eq // ToFlavor[Identity, red],
   "  ", gc = SequenceForm["if the gauge condition ",
    eqn[5, 15, red] // ToFlavor[Identity, red], " holds."]] // FrameBox // DisplayForm
Print["Compare with eq.(5.14) and eq.(5.15),p.172."]
```

Out[•]//DisplayForm=

$\bar{h}_{\mu\nu,\alpha'}{}^{\alpha} == 2\,\kappa\, T_{\mu\nu}$ if the gauge condition $\bar{h}^{\alpha\mu}{}_{,\alpha} == 0$ holds.

Compare with eq.(5.14) and eq.(5.15),p.172.

We lower the dummy index and expand to make eq.(5.14) more intelligible:

```
In[•]:= -ηuu[α, β] × LowerIndex[α, β][eqn[5, 14][[1]]];
ToDifTensor[hb, 3][MetricSimplify[η][ToNonDifTensor[hb][%]]] /. β → α //
  UpDownSwap[α];
```

```
(%% // ExpandPartialD[labs] // EinsteinSum[]) /. TensorValueRules[η] //
  UseCoordinates[{c t, x, y, z}];
GridBox[{{%% == %%% == % == -eqn[5, 14][[2]]}, {gc}}] // FrameBox // DisplayForm
Print["Compare with eq.(5.19),p.172."]
```

Out[•]//DisplayForm=

$$-\bar{h}_{\mu\nu,\alpha}{}^{,\alpha} == -\bar{h}_{\mu\nu,\alpha,\beta}\,\eta^{\alpha\beta} == -\frac{1}{c^2}\partial_{t,t}\bar{h}_{\mu\nu} + \partial_{x,x}\bar{h}_{\mu\nu} + \partial_{y,y}\bar{h}_{\mu\nu} + \partial_{z,z}\bar{h}_{\mu\nu} == -2\,\kappa\,T_{\mu\nu}$$

if the gauge condition $\bar{h}^{\alpha\mu}{}_{,\alpha} == 0$ holds.

```
Compare with eq.(5.19),p.172.
```

Finally we got our good old, well-trusted **wave equation**!

If You like to simplify the notation, at this point You can introduce the *d'Alembertian* $\square^2$ defined by

In[•]:=
```
Tensor[T];
-DAlembertian[α, β][%];
% // ExpandPartialD[labs];
% // ToArrayValues[] // UseCoordinates[{c t, x, y, z}];
("□²" %%%% == SF[%%] == %% == %%% == % == First[%] + "∇²" %%%% // TraditionalForm) /.
 SF[a_] → StandardForm[a]
Print["Compare with eq.(5.17) and eq.(5.18),p.172."]
```

Out[•]//TraditionalForm=

$$\square^2\,T = -\eta^{\alpha\beta}\,\partial_{x^\alpha,\,x^\beta}\,T = -\eta^{\alpha\beta}\,\frac{\partial^2 T}{\partial x^\alpha\,\partial x^\beta} = \eta^{\alpha\beta}\,(-T_{,\alpha,\beta}) = -\frac{1}{c^2}\,\frac{\partial^2 T}{\partial t\,\partial t} + \frac{\partial^2 T}{\partial x\,\partial x} + \frac{\partial^2 T}{\partial y\,\partial y} + \frac{\partial^2 T}{\partial z\,\partial z} = \nabla^2\,T - \frac{1}{c^2}\,\frac{\partial^2 T}{\partial t\,\partial t}$$

```
Compare with eq.(5.17) and eq.(5.18),p.172.
```

Warning: There are other conventions for the sign and the symbol! I use here those of FN.

Pons asinorum: □ is the 4-gradient operator (4 corners) and ∇ is the usual 3D gradient operator (3 corners); we have $\square^2 = \square \cdot \square$ and $\nabla^2 = \nabla \cdot \nabla$. If all this is too edgy for you or you simply prefer it curvy, then multiply everything by 0...

5.2 Two polarizations p. 173-178

In[•]:=
```
Needs["Notation`"]
Notation[h̄ ⟺ hb]

Needs["TGeneralRelativity1`GeneralRelativity`"]
$PrePrint =.
DeclareBaseIndices[{0, 1, 2, 3}]
MyRed = StyleForm[Superscript[#, "′"], FontColor → RGBColor[1, 0, 0]] &;
DeclareIndexFlavor[{red, MyRed}]
DeclareZeroTensor[zero]
labs = {x, δ, g, Γ};
DefineTensorShortcuts[
  {{x, k, zero, ξ, ϵ, u, ξ}, 1},
  {{g, η, h, hb, A, δ, ξ, X, e1, e2}, 2},
  {{Γ, h}, 3},
  {{R}, 4}];
```

```
SetTensorValues[δud[i, j], IdentityMatrix[NDim]]
SetTensorValueRules[ηdd[i, j], DiagonalMatrix[{1, -1, -1, -1}]]
SetTensorValueRules[ηuu[i, j], DiagonalMatrix[{1, -1, -1, -1}]]
```

••• Get: Cannot open Graphics`Colors`.

••• Needs: Context Graphics`Colors` was not created when Needs was evaluated.

••• SetDelayed: Tag Symmetric in Symmetric[args__ /; Length[{args}] > 1][term_] is Protected.

••• SetDelayed: Tag Symmetric in Symmetric[ind_][term_] is Protected.

▲ 2) Gravitational radiation in form of a sinusoidal plane wave

The **simplest** non-trivial solution to the wave equation of empty spacetime is that representing a *sinusoidal plane wave*. The amplitude matrix $A^{\mu\nu}$ is complex, symmetric and all elements are small, but at least one element has to be non-null.

In[•]:=
```
hbuu[μ, ν];
eqn[5, 19, empty] = -DAlembertian[labs, α, β][%] == 0
Auu[μ, ν] Exp[i kd[σ] × xu[σ]];
%%% == Re[%] // FrameBox // DisplayForm
eqn[5, 22, complex] = %%%% → %%
```

Out[•]= $-\eta^{\alpha\beta}\,\partial_{x^\alpha,x^\beta}\bar{h}^{\mu\nu} == 0$

Out[•]//DisplayForm=

$$\boxed{\bar{h}^{\mu\nu} == \mathrm{Re}[e^{i\,k_\sigma x^\sigma}A^{\mu\nu}]}$$

Out[•]= $\bar{h}^{\mu\nu} \to e^{i\,k_\sigma x^\sigma}A^{\mu\nu}$

David Park : "I am going to drop the real part function from the FN notation with the understanding that we should apply it to any final solution."

We check first the FN statement “[4-wave vector] k^μ is null”:

In[•]:=
```
eqn[5, 19, empty]
eqn[5, 22, complex]
%% /. %
% // KroneckerAbsorb[δ] // NondependentPartialD[{A, x, k}] // MetricSimplify[η]
#/%[[1, {1, 2}]] &/@ %
```

Out[•]= $-\eta^{\alpha\beta}\,\partial_{x^\alpha,x^\beta}\bar{h}^{\mu\nu} == 0$

Out[•]= $\bar{h}^{\mu\nu} \to e^{i\,k_\sigma x^\sigma}A^{\mu\nu}$

Out[•]= $-\eta^{\alpha\beta}\,(e^{i\,k_\sigma x^\sigma}\partial_{x^\alpha,x^\beta}A^{\mu\nu} + i\,e^{i\,k_\sigma x^\sigma}\partial_{x^\alpha}A^{\mu\nu}\,(k_\sigma\,\delta^\sigma{}_\beta + x^\sigma\,\partial_{x^\beta}k_\sigma) +$
$i\,(i\,e^{i\,k_\sigma x^\sigma}A^{\mu\nu}\,(k_\sigma\,\delta^\sigma{}_\alpha + x^\sigma\,\partial_{x^\alpha}k_\sigma)\,(k_\sigma\,\delta^\sigma{}_\beta + x^\sigma\,\partial_{x^\beta}k_\sigma) +$
$e^{i\,k_\sigma x^\sigma}\,(\partial_{x^\beta}A^{\mu\nu}\,(k_\sigma\,\delta^\sigma{}_\alpha + x^\sigma\,\partial_{x^\alpha}k_\sigma) + A^{\mu\nu}\,(x^\sigma\,\partial_{x^\alpha,x^\beta}k_\sigma + \delta^\sigma{}_\beta\,\partial_{x^\alpha}k_\sigma + \delta^\sigma{}_\alpha\,\partial_{x^\beta}k_\sigma)))) == 0$

Out[•]= $e^{i\,k_\sigma x^\sigma}A^{\mu\nu}\,k_\beta\,k^\beta == 0$

Out[•]= $k_\beta\,k^\beta == 0$

From the gauge condition we get:

In[•]:=
```
PartialD[labs][NestedTensor[hbuu[μ, ν]], xu[ν]] == zerou[μ]
% /. eqn[5, 22, complex]
```

```
% // UnnestTensor
% // NondependentPartialD[{A, x, k}]
% // MapLevelParts[KroneckerAbsorb[δ], {1, {4, 5}}]
# / %[[1, {1, 2}]] & /@ %
eqn[5, 23] = % /. _zerou[μ] → zerou[μ]
Print["See eq. (5.23), p.173."]
```

Out[•]= $\partial_{x^\nu} \bar{h}^{\mu\nu} = \text{zero}^\mu$

Out[•]= $\partial_{x^\nu} e^{i k_\sigma x^\sigma} A^{\mu\nu} = \text{zero}^\mu$

Out[•]= $e^{i k_\sigma x^\sigma} \partial_{x^\nu} A^{\mu\nu} + i e^{i k_\sigma x^\sigma} A^{\mu\nu} (k_\sigma \delta^\sigma{}_\nu + x^\sigma \partial_{x^\nu} k_\sigma) = \text{zero}^\mu$

Out[•]= $i e^{i k_\sigma x^\sigma} A^{\mu\nu} k_\sigma \delta^\sigma{}_\nu = \text{zero}^\mu$

Out[•]= $i e^{i k_\sigma x^\sigma} A^{\mu\nu} k_\nu = \text{zero}^\mu$

Out[•]= $A^{\mu\nu} k_\nu = -i e^{-i k_\sigma x^\sigma} \text{zero}^\mu$

Out[•]= $A^{\mu\nu} k_\nu = \text{zero}^\mu$

See eq. (5.23), p.173.

For a sinusoidal plane wave propagating in the z direction we have for $k > 0$

In[•]:=
```
$Assumptions = k > 0;
SetTensorValueRules[ku[μ], {k, 0, 0, k}]
ku[μ];
% == ToArrayValues[][%]
kd[μ];
ηdd[μ, ν] × ku[ν];
ToArrayValues[][%];
SetTensorValueRules[kd[μ], %]
%%%% == %%% == %%
```

Out[•]= $k^\mu = \{k, 0, 0, k\}$

Out[•]= $k_\mu = k^\nu \eta_{\mu\nu} = \{k, 0, 0, -k\}$

Together with the gauge condition we get

In[•]:=
```
eqn[5, 23]
% // ToArrayValues[]
% // Simplify
Print["Symmetrizing the slots"]
%% // SymmetrizeSlots[A, 2, {1, {1, 2}}]
Print["Turn into the gauge condition substitution rules"]
(gcsr = Apply[Rule, #] & /@ Reverse /@ %%)
```

Out[•]= $A^{\mu\nu} k_\nu = \text{zero}^\mu$

Out[•]= $\{k A^{00} - k A^{03} = 0,\ k A^{10} - k A^{13} = 0,\ k A^{20} - k A^{23} = 0,\ k A^{30} - k A^{33} = 0\}$

Out[•]= $\{A^{00} = A^{03},\ A^{10} = A^{13},\ A^{20} = A^{23},\ A^{30} = A^{33}\}$

Symmetrizing the slots

Out[•]= $\{A^{00} = A^{03},\ A^{01} = A^{13},\ A^{02} = A^{23},\ A^{03} = A^{33}\}$

Turn into the gauge condition substitution rules

Out[•]= {A^03 → A^00, A^13 → A^01, A^23 → A^02, A^33 → A^03}

If we apply symmetry and the gauge condition substitution rules on the amplitude matrix $A^{\mu\nu}$, we obtain:

```
In[•]:= Auu[μ, ν];
        % == ToArrayValues[][%];
        MatrixForm /@ %
        Print["1) Applying symmetry"]
        %%% /. Auu[a_, b_] /; a > b → Auu[b, a];
        MatrixForm /@ %
        Print["2) Applying (repeatedly) the gauge condition substitution rules"]
        %%% //. gcsr;
        MatrixForm /@ %
        Print["See eq.(5.25),p.174"]
        SetTensorValues[Auu[μ, ν], %%%[[2]]]
```

Out[•]= $A^{\mu\nu} == \begin{pmatrix} A^{00} & A^{01} & A^{02} & A^{03} \\ A^{10} & A^{11} & A^{12} & A^{13} \\ A^{20} & A^{21} & A^{22} & A^{23} \\ A^{30} & A^{31} & A^{32} & A^{33} \end{pmatrix}$

1) Applying symmetry

Out[•]= $A^{\mu\nu} == \begin{pmatrix} A^{00} & A^{01} & A^{02} & A^{03} \\ A^{01} & A^{11} & A^{12} & A^{13} \\ A^{02} & A^{12} & A^{22} & A^{23} \\ A^{03} & A^{13} & A^{23} & A^{33} \end{pmatrix}$

2) Applying (repeatedly) the gauge condition substitution rules

Out[•]= $A^{\mu\nu} == \begin{pmatrix} A^{00} & A^{01} & A^{02} & A^{00} \\ A^{01} & A^{11} & A^{12} & A^{01} \\ A^{02} & A^{12} & A^{22} & A^{02} \\ A^{00} & A^{01} & A^{02} & A^{00} \end{pmatrix}$

See eq.(5.25),p.174

■ We consider now a transformation to a special gauge, called *TT gauge*, which is a subtype of the harmonic gauge and is generated by the following gauge function:

```
In[•]:= ξu[μ] == -ⅈ ϵu[μ] Exp[ⅈ kd[α] × xu[α]]
        rule[5, 25, a][α_] = Rule @@ % // LHSSymbolsToPatterns[{μ}]
```

Out[•]= $\xi^\mu == -i\, e^{i k_\alpha x^\alpha} \epsilon^\mu$

Out[•]= $\xi^{\mu}\_ \to -i\, e^{i k_\alpha x^\alpha} \epsilon^\mu$

where ϵ^μ is a constant vector with all small elements.

The following calculation shows that the new ξ^μ satisfies the condition eq.(5.21), preserving the gauge condition.

```
In[•]:= Print["Wave equation for a general gauge function ξ^μ"]
        -DAlembertian[labs, β, γ][NestedTensor[ξu[μ]]] == zerou[μ]
        Print["Substituting for ξ^μ"]
        %% /. rule[5, 25, a][α]
        Print["Expanding the partial derivative and simplifying (k^μ and ϵ^μ are constants)"]
        %% // UnnestTensor;
        % /. a_ kd[α] :→ KroneckerAbsorb[δ][a kd[α]];
```

```
% // NondependentPartialD[{k, ϵ, x}]
Print["Metric simplify and use the fact that k^μ is a null vector"]
%% // MetricSimplify[η]
% // ToArrayValues[] // Union
```

Wave equation for a general gauge function ξ^μ

Out[•]= $-\eta^{\beta\gamma}\,\partial_{x^\beta,x^\gamma}\,\xi^\mu == \text{zero}^\mu$

Substituting for ξ^μ

Out[•]= $-\eta^{\beta\gamma}\,\partial_{x^\beta,x^\gamma} - i\, e^{i\,k_\alpha\,x^\alpha}\,\epsilon^\mu == \text{zero}^\mu$

Expanding the partial derivative and simplifying (k^μ and ϵ^μ are constants)

Out[•]= $-i\, e^{i\,k_\alpha\,x^\alpha}\,k_\beta\,k_\gamma\,\epsilon^\mu\,\eta^{\beta\gamma} == \text{zero}^\mu$

Metric simplify and use the fact that k^μ is a null vector

Out[•]= $-i\, e^{i\,k_\alpha\,x^\alpha}\,k_\gamma\,k^\gamma\,\epsilon^\mu == \text{zero}^\mu$

Out[•]= {True}

The derivatives $\xi^\mu{}_{,\nu}$ and $\xi^{\mu,\nu}$ of the gauge function are:

In[•]:=
```
Print["Covariant form:"]
PartialD[ξu[μ], ν] == PartialD[labs][NestedTensor[ξu[μ]], xu[ν]]
% /. rule[5, 25, a][α]
% // UnnestTensor
% // NondependentPartialD[{k, ϵ, x}]
Print["Applying the Kronecker delta"]
eqn[5, 26] = %% // MapLevelParts[KroneckerAbsorb[δ], {2, {2, 3}}]
Print["Compare with eq.(5.26),p.174."]
Print["Turn into a rule"]
rule[5, 26, cov] = Rule @@ eqn[5, 26] // LHSSymbolsToPatterns[{μ, ν}]
Print["Contravariant form:"]
ηuu[ν, β] # & /@ eqn[5, 26]
Print["Switch to plain notation and metric simplification"]
%% // ToNonDifTensor[ξ]
MetricSimplify[η] /@ %
Print["Switch back to ordinary notation, rename index and turn into a rule"]
%% // ToDifTensor[ξ, 2]
% // IndexChange[{β, ν}]
rule[5, 26, contra] = Rule @@ % // LHSSymbolsToPatterns[{μ, ν}]
```

Covariant form:

Out[•]= $\xi^\mu{}_{,\nu} == \partial_{x^\nu}\,\xi^\mu$

Out[•]= $\xi^\mu{}_{,\nu} == \partial_{x^\nu} - i\, e^{i\,k_\alpha\,x^\alpha}\,\epsilon^\mu$

Out[•]= $\xi^\mu{}_{,\nu} == -i\,(i\, e^{i\,k_\alpha\,x^\alpha}\,\epsilon^\mu\,(k_\alpha\,\delta^\alpha{}_\nu + x^\alpha\,\partial_{x^\nu}\,k_\alpha) + e^{i\,k_\alpha\,x^\alpha}\,\partial_{x^\nu}\,\epsilon^\mu)$

Out[•]= $\xi^\mu{}_{,\nu} == e^{i\,k_\alpha\,x^\alpha}\,k_\alpha\,\delta^\alpha{}_\nu\,\epsilon^\mu$

Applying the Kronecker delta

Out[•]= $\xi^\mu{}_{,\nu} == e^{i\,k_\alpha\,x^\alpha}\,k_\nu\,\epsilon^\mu$

Compare with eq.(5.26),p.174.

Turn into a rule

Out[∘]= $\xi^{\mu}{}_{-,\nu\_} \to e^{i\, k_\alpha x^\alpha}\, k_\nu\, \epsilon^\mu$

Contravariant form:

Out[∘]= $\eta^{\nu\beta}\, \xi^{\mu}{}_{,\nu} == e^{i\, k_\alpha x^\alpha}\, k_\nu\, \epsilon^\mu\, \eta^{\nu\beta}$

Switch to plain notation and metric simplification

Out[∘]= $\eta^{\nu\beta}\, \xi^{\mu}{}_{\nu} == e^{i\, k_\alpha x^\alpha}\, k_\nu\, \epsilon^\mu\, \eta^{\nu\beta}$

Out[∘]= $\xi^{\mu\beta} == e^{i\, k_\alpha x^\alpha}\, k^\beta\, \epsilon^\mu$

Switch back to ordinary notation, rename index and turn into a rule

Out[∘]= $\xi^{\mu,\beta} == e^{i\, k_\alpha x^\alpha}\, k^\beta\, \epsilon^\mu$

Out[∘]= $\xi^{\mu,\nu} == e^{i\, k_\alpha x^\alpha}\, k^\nu\, \epsilon^\mu$

Out[∘]= $\xi^{\mu\_,\nu\_} \to e^{i\, k_\alpha x^\alpha}\, k^\nu\, \epsilon^\mu$

We will need a first order approximation for $k_{\alpha'}\, x^{\alpha'}$ in the TT gauge:

```
In[∘]:= kd[red@α] × xu[red@β] == Xud[a, red@α] × Xud[red@β, b] × kd[a] × xu[b]
        Print["General gauge transformation matrix (see p.171):"]
        {Xud[red@μ_, ν_] → δud[μ, ν] + ξud[μ, Dif[ν]],
         Xud[μ_, red@ν_] → δud[μ, ν] - ξud[μ, Dif[ν]]}
        %%% /. %
        % // Expand
        % // KroneckerAbsorb[δ]
        Print["Drop second order terms"]
        %% /. HoldPattern[Tensor[ξ, __] × Tensor[ξ, __]] → 0
        % /. β → α
        rekx = Rule @@ % /. rule[5, 26, cov]
```

Out[∘]= $k_{\alpha'}\, x^{\beta'} == k_a\, x^b\, X^a{}_{\alpha'}\, X^{\beta'}{}_b$

General gauge transformation matrix (see p.171):

Out[∘]= $\{X^{\mu\_'}{}_{\nu\_} \to \delta^\mu{}_\nu + \xi^\mu{}_{,\nu},\ X^{\mu\_}{}_{\nu\_'} \to \delta^\mu{}_\nu - \xi^\mu{}_{,\nu}\}$

Out[∘]= $k_{\alpha'}\, x^{\beta'} == k_a\, x^b\, (\delta^a{}_\alpha - \xi^a{}_{,\alpha})\, (\delta^\beta{}_b + \xi^\beta{}_{,b})$

Out[∘]= $k_{\alpha'}\, x^{\beta'} == k_a\, x^b\, \delta^a{}_\alpha\, \delta^\beta{}_b - k_a\, x^b\, \delta^\beta{}_b\, \xi^a{}_{,\alpha} + k_a\, x^b\, \delta^a{}_\alpha\, \xi^\beta{}_{,b} - k_a\, x^b\, \xi^a{}_{,\alpha}\, \xi^\beta{}_{,b}$

Out[∘]= $k_{\alpha'}\, x^{\beta'} == k_\alpha\, x^\beta - k_a\, x^\beta\, \xi^a{}_{,\alpha} + k_\alpha\, x^b\, \xi^\beta{}_{,b} - k_a\, x^b\, \xi^a{}_{,\alpha}\, \xi^\beta{}_{,b}$

Drop second order terms

Out[∘]= $k_{\alpha'}\, x^{\beta'} == k_\alpha\, x^\beta - k_a\, x^\beta\, \xi^a{}_{,\alpha} + k_\alpha\, x^b\, \xi^\beta{}_{,b}$

Out[∘]= $k_{\alpha'}\, x^{\alpha'} == k_\alpha\, x^\alpha - k_a\, x^\alpha\, \xi^a{}_{,\alpha} + k_\alpha\, x^b\, \xi^\alpha{}_{,b}$

Out[∘]= $k_{\alpha'}\, x^{\alpha'} \to k_\alpha\, x^\alpha - e^{i\, k_\alpha x^\alpha}\, k_a\, k_\alpha\, x^\alpha\, \epsilon^a + e^{i\, k_\alpha x^\alpha}\, k_b\, k_\alpha\, x^b\, \epsilon^\alpha$

Note: The difference $k_{\alpha'}\, x^{\alpha'} - k_\alpha\, x^\alpha$ is a first order quantity.

We recall eq.(5.11), p.171, valid for a general gauge transformation, and substitute for $\xi^\alpha{}_{,\alpha}$ and $\xi^{\mu,\nu}$ in the TT gauge:

```
In[∘]:= hbuu[red@μ, red@ν] == hbuu[μ, ν] + ηuu[μ, ν] × ξud[α, Dif[α]] - ξuu[μ, Dif[ν]] - ξuu[ν, Dif[μ]]
        % /. {rule[5, 26, cov], rule[5, 26, contra]}
        Print["Substituting both "h̄"^μν expressions and simplifying"]
```

```
%% /. eqn[5, 22, complex] /.
  (eqn[5, 22, complex] // ToFlavor[red]) /. σ → α
Simplify[# / %[[2, 1, 1]]] & /@ %
Print["The argument of the exponential is small, so we put ⅇ^a≃1+a"]
%% /. Exp[a_] → 1 + a // Expand
Print["Substituing for k_α"," x^α"," "]
%% /. rekx // Expand
Print["Dropping second order terms on lhs"]
%% /. HoldPattern[Tensor[A, __] × Tensor[ϵ, __]] → 0
Print["In matrix form:"]
MatrixForm /@ (eqnAA = ToArrayValues[] /@ %% // Simplify)
Print["(See FN,p.174)"]
```

Out[∘]= $\bar{h}^{\mu'\nu'} == \bar{h}^{\mu\nu} + \eta^{\mu\nu}\,\xi^{\alpha}{}_{,\alpha} - \xi^{\mu,\nu} - \xi^{\nu,\mu}$

Out[∘]= $\bar{h}^{\mu'\nu'} == \bar{h}^{\mu\nu} - e^{i k_\alpha x^\alpha} k^\nu \epsilon^\mu - e^{i k_\alpha x^\alpha} k^\mu \epsilon^\nu + e^{i k_\alpha x^\alpha} k_\alpha \epsilon^\alpha \eta^{\mu\nu}$

Substituting both $\bar{h}^{\mu\nu}$ expressions and simplifying

Out[∘]= $e^{i k_{\alpha'} x^{\alpha'}} A^{\mu'\nu'} == e^{i k_\alpha x^\alpha} A^{\mu\nu} - e^{i k_\alpha x^\alpha} k^\nu \epsilon^\mu - e^{i k_\alpha x^\alpha} k^\mu \epsilon^\nu + e^{i k_\alpha x^\alpha} k_\alpha \epsilon^\alpha \eta^{\mu\nu}$

Out[∘]= $e^{-i k_\alpha x^\alpha + i k_{\alpha'} x^{\alpha'}} A^{\mu'\nu'} == A^{\mu\nu} - k^\nu \epsilon^\mu - k^\mu \epsilon^\nu + k_\alpha \epsilon^\alpha \eta^{\mu\nu}$

The argument of the exponential is small, so we put $e^a \simeq 1+a$

Out[∘]= $A^{\mu'\nu'} - i A^{\mu'\nu'} k_\alpha x^\alpha + i A^{\mu'\nu'} k_{\alpha'} x^{\alpha'} == A^{\mu\nu} - k^\nu \epsilon^\mu - k^\mu \epsilon^\nu + k_\alpha \epsilon^\alpha \eta^{\mu\nu}$

Substituing for $k_{\alpha'}$ $x^{\alpha'}$

Out[∘]= $A^{\mu'\nu'} - i e^{i k_\alpha x^\alpha} A^{\mu'\nu'} k_a k_\alpha x^\alpha \epsilon^a + i e^{i k_\alpha x^\alpha} A^{\mu'\nu'} k_b k_\alpha x^b \epsilon^\alpha == A^{\mu\nu} - k^\nu \epsilon^\mu - k^\mu \epsilon^\nu + k_\alpha \epsilon^\alpha \eta^{\mu\nu}$

Dropping second order terms on lhs

Out[∘]= $A^{\mu'\nu'} == A^{\mu\nu} - k^\nu \epsilon^\mu - k^\mu \epsilon^\nu + k_\alpha \epsilon^\alpha \eta^{\mu\nu}$

In matrix form:

Out[∘]= $$\begin{pmatrix} A^{0'0'} & A^{0'1'} & A^{0'2'} & A^{0'3'} \\ A^{1'0'} & A^{1'1'} & A^{1'2'} & A^{1'3'} \\ A^{2'0'} & A^{2'1'} & A^{2'2'} & A^{2'3'} \\ A^{3'0'} & A^{3'1'} & A^{3'2'} & A^{3'3'} \end{pmatrix} == \begin{pmatrix} A^{00} - k\,(\epsilon^0+\epsilon^3) & A^{01} - k\,\epsilon^1 & A^{02} - k\,\epsilon^2 & A^{00} - k\,(\epsilon^0+\epsilon^3) \\ A^{01} - k\,\epsilon^1 & A^{11} + k\,(-\epsilon^0+\epsilon^3) & A^{12} & A^{01} - k\,\epsilon^1 \\ A^{02} - k\,\epsilon^2 & A^{12} & A^{22} + k\,(-\epsilon^0+\epsilon^3) & A^{02} - k\,\epsilon^2 \\ A^{00} - k\,(\epsilon^0+\epsilon^3) & A^{01} - k\,\epsilon^1 & A^{02} - k\,\epsilon^2 & A^{00} - k\,(\epsilon^0+\epsilon^3) \end{pmatrix}$$

(See FN,p.174)

Conveniently choosing our constants ϵ^μ (which means specifying the special gauge transformation completely) and noting that then $A^{1'1'} = -A^{2'2'}$, we see that $A^{\mu'\nu'}$ is determined by only two independent variables α and β.

In[∘]:=
```
{ϵu[0] → (2 Auu[0, 0] + Auu[1, 1] + Auu[2, 2]) / (4 k),
 ϵu[1] → Auu[0, 1] / k,
 ϵu[2] → Auu[0, 2] / k,
 ϵu[3] → (2 Auu[0, 0] - Auu[1, 1] - Auu[2, 2]) / (4 k)}
Simplify[eqnAA[[2]] /. %];
% /. {Auu[1, 1] → 2 α + Auu[2, 2], Auu[1, 2] → β};
MatrixForm /@ (eqnAA[[1]] == %% == %)
```

Out[∘]= $\left\{\epsilon^0 \to \frac{2A^{00} + A^{11} + A^{22}}{4k},\ \epsilon^1 \to \frac{A^{01}}{k},\ \epsilon^2 \to \frac{A^{02}}{k},\ \epsilon^3 \to \frac{2A^{00} - A^{11} - A^{22}}{4k}\right\}$

Out[•]= $\begin{pmatrix} A^{0'0'} & A^{0'1'} & A^{0'2'} & A^{0'3'} \\ A^{1'0'} & A^{1'1'} & A^{1'2'} & A^{1'3'} \\ A^{2'0'} & A^{2'1'} & A^{2'2'} & A^{2'3'} \\ A^{3'0'} & A^{3'1'} & A^{3'2'} & A^{3'3'} \end{pmatrix} == \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & \frac{1}{2}(A^{11} - A^{22}) & A^{12} & 0 \\ 0 & A^{12} & \frac{1}{2}(-A^{11} + A^{22}) & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} == \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & \alpha & \beta & 0 \\ 0 & \beta & -\alpha & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$

The values ϵ^μ specifying the gauge are determined by the values of the amplitude matrix $A^{\mu\nu}$ of the sinusoidal plane wave from which we started. FN: "This new gauge, *which is determined by the wave itself*, is known as the *transverse traceless gauge*, or *TT gauge* for short." We shall work in the TT gauge for the remainder of this section and drop primes (and color) for simplicity.

We can introduce two *linear polarization matrices* $[e1^{\mu\nu}]$ and $[e2^{\mu\nu}]$ defined by...

```
In[•]:= SetTensorValueRules[e1uu[μ, ν], DiagonalMatrix[{0, 1, -1, 0}]]
        SetTensorValueRules[e2uu[μ, ν], Reverse[DiagonalMatrix[{0, 1, 1, 0}]]]
        eq[5, 28] = (# == MatrixForm[ToArrayValues[][#]] &/@ {e1uu[μ, ν], e2uu[μ, ν]})
        Print["See eq.(5.28),p.175."]
```

Out[•]= $\left\{e1^{\mu\nu} == \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, e2^{\mu\nu} == \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}\right\}$

```
See eq.(5.28),p.175.
```

The general amplitude matrix in TT gauge is then given by the following linear combination, where α and β are complex constants:

```
In[•]:= eqn[5, 29] = Auu[μ, ν] == α e1uu[μ, ν] + β e2uu[μ, ν]
        Print["See eq.(5.29),p.175."]
```

Out[•]= $A^{\mu\nu} == \alpha\, e1^{\mu\nu} + \beta\, e2^{\mu\nu}$

```
See eq.(5.29),p.175.
```

We can show that in TT gauge $\bar{h}^{\mu\nu}$ is *traceless* with the following calculation...

```
In[•]:= Print["Recall eq.(5.22)"]
        Equal @@ eqn[5, 22, complex]
        Print["Calculating the trace"]
        ηdd[μ, ν] # &/@ %%
        MetricSimplify[η] /@ %
        UpDownSwap[ν] /@ %
        Print["Substituting the with contravariant expression A^μν"]
        %% /. Aud[ν, ν] → Auu[ν, a] × ηdd[ν, a]
        Print["Expanding the rhs"]
        %% // MapLevelParts[ToArrayValues[], {2, {2, 3}}]
        Auu[1, 1] == -Auu[2, 2];
        Print["In TT gauge ", %]
        %%% /. Rule @@ %%
```

```
Recall eq.(5.22)
```

Out[•]= $\bar{h}^{\mu\nu} == e^{i k_\sigma x^\sigma} A^{\mu\nu}$

```
Calculating the trace
```

Out[•]= $\bar{h}^{\mu\nu}\, \eta_{\mu\nu} == e^{i k_\sigma x^\sigma} A^{\mu\nu}\, \eta_{\mu\nu}$

Out[•]= $\bar{h}_\nu{}^\nu == e^{i k_\sigma x^\sigma} A_\nu{}^\nu$

Out[•]= $\bar{h}^{\nu}{}_{\nu} == e^{i k_\sigma x^\sigma} A^{\nu}{}_{\nu}$

Substituting the with contravariant expression $A^{\mu\nu}$

Out[•]= $\bar{h}^{\nu}{}_{\nu} == e^{i k_\sigma x^\sigma} A^{\nu a} \eta_{\nu a}$

Expanding the rhs

Out[•]= $\bar{h}^{\nu}{}_{\nu} == e^{i k_\sigma x^\sigma} (-A^{11} - A^{22})$

In TT gauge $A^{11} == -A^{22}$

Out[•]= $\bar{h}^{\nu}{}_{\nu} == 0$

Furthermore, from Exercise 5.1.2, p.173, solution p.273, we have $h_{\mu\nu} == \bar{h}_{\mu\nu} - \frac{1}{2} \bar{h}^{\sigma}{}_{\sigma} \eta_{\mu\nu}$, hence $h_{\mu\nu} == \bar{h}_{\mu\nu} == \eta_{a\mu} \eta_{b\nu} e^{i k_\sigma x^\sigma} A^{ab}$ in TT gauge.

```
In[•]:= ηdd[a, μ] × ηdd[b, ν] × Tensor[T, {a, b}, {Void, Void}];
        % // MetricSimplify[η];
        %% // ToArrayValues[] // MatrixForm;
        Print["Note:  ", %%% == %% == %]
```

Note: $T^{ab} \eta_{a\mu} \eta_{b\nu} == T_{\mu\nu} == \begin{pmatrix} T^{00} & -T^{01} & -T^{02} & -T^{03} \\ -T^{10} & T^{11} & T^{12} & T^{13} \\ -T^{20} & T^{21} & T^{22} & T^{23} \\ -T^{30} & T^{31} & T^{32} & T^{33} \end{pmatrix}$

▲ 3) Manifestation of a gravitational sinusoidal plane wave

We derive here a faithful representation of actual *spatial separation* between free massive particles during a passing gravitational wave.

Let's show first that in the TT gauge associated with a plane wave we have $\Gamma^{\mu}{}_{00} == 0$ and $\Gamma^{\mu}{}_{0\nu} == \frac{1}{2} h^{\mu}{}_{\nu,0}$ (see Exercise 5.2.2, p.178, solution p.273). This relations will be used soon.

```
In[•]:= Print["Christoffel symbols as given by eq.(5.2),p.171:"]
        CS = Γudd[μ, ν, σ] ==
          1 / 2 ηuu[μ, β] (PartialD[hdd[σ, β], ν] + PartialD[hdd[ν, β], σ] - PartialD[hdd[ν, σ], β])
        {ν → 0, σ → 0};
        Print["Substituting the index values ", %, ":"]
        %%% /. %%
        Print["In TT gauge we have hμν = "‾"h μν and therefore its border elements are all zero."]
        %% /. hddd[0, _, _] → zerod[β]
        (eqnA = (% // MetricSimplify[η]) /. _zerou[a_] → zerou[a]) // FrameBox // DisplayForm
        %[[1]] // ToArrayValues[]
```

Christoffel symbols as given by eq.(5.2),p.171:

Out[•]= $\Gamma^{\mu}{}_{\nu\sigma} == \frac{1}{2} (h_{\nu\beta,\sigma} - h_{\nu\sigma,\beta} + h_{\sigma\beta,\nu}) \eta^{\mu\beta}$

Substituting the index values {ν → 0, σ → 0}:

Out[•]= $\Gamma^{\mu}{}_{00} == \frac{1}{2} (-h_{00,\beta} + 2 h_{0\beta,0}) \eta^{\mu\beta}$

In TT gauge we have $h_{\mu\nu} = \bar{h}_{\mu\nu}$ and therefore its border elements are all zero.

Out[•]= $\Gamma^{\mu}{}_{00} == \frac{1}{2} \text{zero}_{\beta} \eta^{\mu\beta}$

Out[•]//DisplayForm=

$$\Gamma^{\mu}{}_{00} == \text{zero}^{\mu}$$

Out[•]= $\{\Gamma^{0}{}_{00} == 0,\ \Gamma^{1}{}_{00} == 0,\ \Gamma^{2}{}_{00} == 0,\ \Gamma^{3}{}_{00} == 0\}$

In[•]:=
```
Print["Christoffel symbols again:"]
CS
{ν → 0, σ → ν};
Print["Substituting the index values ", %, ":"]
%%% /. %%
Print["In TT gauge, hμν = "‾"h μν and hence its border elements are all zero."]
%% /. hddd[0, _, _] → 0
Print["Switch to plain notation and metric simplify:"]
%% // ToNonDifTensor[h] // MetricSimplify[η]
Print["Switch back to ordinary notation and symmetrize first two slots:"]
(eqnB = %% // ToDifTensor[h, 3]) // SymmetrizeSlots[h, 3, {1, {1, 2}}] // FrameBox //
 DisplayForm
```

Christoffel symbols again:

Out[•]= $\Gamma^{\mu}{}_{\nu\sigma} == \frac{1}{2}\,(h_{\nu\beta,\sigma} - h_{\nu\sigma,\beta} + h_{\sigma\beta,\nu})\ \eta^{\mu\beta}$

Substituting the index values {ν → 0, σ → ν}:

Out[•]= $\Gamma^{\mu}{}_{0\nu} == \frac{1}{2}\,(h_{0\beta,\nu} - h_{0\nu,\beta} + h_{\nu\beta,0})\ \eta^{\mu\beta}$

In TT gauge, $h_{\mu\nu} = \bar{h}_{\mu\nu}$ and hence its border elements are all zero.

Out[•]= $\Gamma^{\mu}{}_{0\nu} == \frac{1}{2}\,h_{\nu\beta,0}\ \eta^{\mu\beta}$

Switch to plain notation and metric simplify:

Out[•]= $\Gamma^{\mu}{}_{0\nu} == \frac{1}{2}\,h_{\nu}{}^{\mu}{}_{0}$

Switch back to ordinary notation and symmetrize first two slots:

Out[•]//DisplayForm=

$$\Gamma^{\mu}{}_{0\nu} == \frac{1}{2}\,h^{\mu}{}_{\nu,0}$$

We transform this two results into a rule:

In[•]:=
```
Gammar = Rule @@@ (LHSSymbolsToPatterns[{μ, ν}] /@ {eqnA, eqnB})
```

Out[•]= $\{\Gamma^{\mu\_}{}_{00} \to \text{zero}^{\mu},\ \Gamma^{\mu\_}{}_{0\nu\_} \to \frac{1}{2}\,h_{\nu}{}^{\mu}{}_{,0}\}$

The motion of free falling particles is determined by the **geodesic equation**.

In[•]:=
```
Print["4-position of a particle: ",
 xu[μ] == xu[μ][τ], " parametrized by the proper time τ."]
Print["4-velocity: ", uu[μ] == TotalD[xu[μ], τ]]
Print["4-acceleration: ", TotalD[uu[μ], τ]]
Print[
 "Geodesic equation with proper time as affine parameter in terms of 4-velocity uμ:"]
(ge = PartialD[{u, δ, g, Γ}][uu[μ], τ] + Γudd[μ, ν, σ] × uu[σ] × uu[ν] == zerou[μ]) // FrameBox //
```

```
 DisplayForm
Print["Fully written out:"]
Print[ge // EinsteinSum[], "   with μ=0,1,2,3"]
```

4-position of a particle: $x^\mu == x^\mu[\tau]$ parametrized by the proper time τ.

4-velocity: $u^\mu == \frac{dx^\mu}{d\tau}$

4-acceleration: $\frac{du^\mu}{d\tau}$

Geodesic equation with proper time as affine parameter in terms of 4-velocity u^μ:

Out[•]//DisplayForm=

$$u^\nu u^\sigma \Gamma^\mu{}_{\nu\sigma} + \partial_\tau u^\mu == \text{zero}^\mu$$

Fully written out:

$$(u^0)^2 \Gamma^\mu{}_{00} + u^0 u^1 \Gamma^\mu{}_{01} + u^0 u^2 \Gamma^\mu{}_{02} + u^0 u^3 \Gamma^\mu{}_{03} + u^0 u^1 \Gamma^\mu{}_{10} + (u^1)^2 \Gamma^\mu{}_{11} + u^1 u^2 \Gamma^\mu{}_{12} + u^1 u^3 \Gamma^\mu{}_{13} + u^0 u^2 \Gamma^\mu{}_{20} + u^1 u^2 \Gamma^\mu{}_{21} + (u^2)^2 \Gamma^\mu{}_{22} + u^2 u^3 \Gamma^\mu{}_{23} + u^0 u^3 \Gamma^\mu{}_{30} + u^1 u^3 \Gamma^\mu{}_{31} + u^2 u^3 \Gamma^\mu{}_{32} + (u^3)^2 \Gamma^\mu{}_{33} + \partial_\tau u^\mu == \text{zero}^\mu \quad \text{with } \mu=0,1,2,3$$

The geodesic equation in the TT gauge admits a **constant spatial coordinates solution**; the 4-velocity for this solution is $u^\mu = (c, 0, 0, 0) = c\,\delta^\mu{}_0$.

In[•]:=
```
sol = uu[μ_] → c δud[μ, 0];
Print["Substitute 4-velocity of the constant spatial coordinates solution ", sol]
ge
% /. sol
Print["Absorb the Kronecker delta, put Γ^μ_00=0 and simplify"]
%% // KroneckerAbsorb[δ]
% /. Gammar /. _zerou[a_] → zerou[a]
Print["Check: 4-velocities should have square length equal c^2."]
uu[ν] × ud[ν] == c^2
gdd[μ, ν] × uu[μ] × uu[ν] == c^2
% /. gdd[μ, ν] → ηdd[μ, ν] + hdd[μ, ν]
% /. sol
% // ExpandAll
% // KroneckerAbsorb[δ]
Print["In TT gauge the border elements of h_μν are all zero and η_00=1."]
%% /. hdd[0, _] → 0 /. TensorValueRules[η]
```

Substitute 4-velocity of the constant spatial coordinates solution $u^{\mu\_} \to c\,\delta^\mu{}_0$

Out[•]= $u^\nu u^\sigma \Gamma^\mu{}_{\nu\sigma} + \partial_\tau u^\mu == \text{zero}^\mu$

Out[•]= $c^2 \Gamma^\mu{}_{\nu\sigma} \delta^\nu{}_0 \delta^\sigma{}_0 == \text{zero}^\mu$

Absorb the Kronecker delta, put $\Gamma^\mu{}_{00}$=0 and simplify

Out[•]= $c^2 \Gamma^\mu{}_{00} == \text{zero}^\mu$

Out[•]= True

Check: 4-velocities should have square length equal c^2.

Out[•]= $u_\nu u^\nu == c^2$

Out[•]= $g_{\mu\nu} u^\mu u^\nu == c^2$

Out[•]= $u^\mu u^\nu (h_{\mu\nu} + \eta_{\mu\nu}) == c^2$

Out[•]= $c^2\,\delta^\mu{}_0\,\delta^\nu{}_0\,(h_{\mu\nu}+\eta_{\mu\nu}) == c^2$

Out[•]= $c^2\,h_{\mu\nu}\,\delta^\mu{}_0\,\delta^\nu{}_0 + c^2\,\delta^\mu{}_0\,\delta^\nu{}_0\,\eta_{\mu\nu} == c^2$

Out[•]= $c^2\,h_{00} + c^2\,\eta_{00} == c^2$

In TT gauge the border elements of $h_{\mu\nu}$ are all zero and $\eta_{00}=1$.

Out[•]= True

FN : "Hence curves having constant spatial coordinates are timelike geodesics, and may be taken as the world lines of a cloud of test particles. [...] However, this does *not* mean that their spatial separation *d* is constant [...]"

We consider two test particles moving on **neighboring geodesics** separated by a small *spatial separation vector* ξ^μ.

Notation: ξ^μ no longer means the gauge function. ∂_x, d/dx, ∇_x and D/dx denote respectively the **partial derivative**, the **total derivative**, the **covariant derivative** and the **absolute derivative** with respect to x. The different types of derivatives must be distinguished carefully to avoid subtle mistakes! (From the **ExpandTotalD** help: "Tensors are functions of the coordinates and any variation of the tensor is due to the variation of the coordinates over the parameter of differentiation used. Tensors are not allowed to depend on the parameters directly but only through the coordinates!")

We show now that in the TT gauge associated with a plane wave the constant spatial coordinates solution (with $u^\mu = c\,\delta^\mu{}_0$) implies that the equation of geodesic deviation has ξ^μ = constant as a solution, which means that that the spatial separation vector between two nearby particles is constant (see Problem 5.1, p.182, solution p.274).

In[•]:=
```
Print["Equation of geodesic deviation with
   proper time τ = x⁰/c as affine parameter,eq.(3.41),p.113:"]
AbsoluteD[ξu[μ], {τ, τ}] + Ruddd[μ, σ, ν, ρ] × ξu[ν] × uu[σ] × uu[ρ] == 0
Print[
 "Substitute 4-velocity of the constant spatial coordinates solution and contract"]
%% /. sol
temp = % // KroneckerAbsorb[δ]
Print["Apply the Riemann rule"]
%% /. (RiemannRule // LHSSymbolsToPatterns[{a, b, c, d}]) // IndexChange[{e, ρ}]
Gammar;
Print["Apply the rules from Exercise 5.2.2 ", %]
%%% /. %%
% /. zerou[a_] → 0
Print["Rearrange"]
step1 = (%%[[1, 1]] == Minus /@ %%[[1, 2]]) // Expand
Print["This is the equation of geodesic deviation specialized
   for TT gauge and for the constant spatial coordinates solution."]
```

Equation of geodesic deviation with proper time $\tau = x^0/c$ as affine parameter,eq.(3.41),p.113:

Out[•]= $\frac{D^2\xi^\mu}{d\tau d\tau} + R^\mu{}_{\sigma\nu\rho}\,u^\rho\,u^\sigma\,\xi^\nu == 0$

Substitute 4-velocity of the constant spatial coordinates solution and contract

Out[•]= $\frac{D^2\xi^\mu}{d\tau d\tau} + c^2\,R^\mu{}_{\sigma\nu\rho}\,\delta^\rho{}_0\,\delta^\sigma{}_0\,\xi^\nu == 0$

Out[•]= $\frac{D^2\xi^\mu}{d\tau d\tau} + c^2\,R^\mu{}_{0\nu 0}\,\xi^\nu == 0$

Apply the Riemann rule

Out[•]= $\frac{D^2\xi^\mu}{d\tau d\tau} + c^2\,\xi^\nu\,(\Gamma^\mu{}_{\nu\rho}\,\Gamma^\rho{}_{00} - \Gamma^\mu{}_{0\rho}\,\Gamma^\rho{}_{0\nu} + \partial_{x^\nu}\Gamma^\mu{}_{00} - \partial_{x^0}\Gamma^\mu{}_{0\nu}) == 0$

Apply the rules from Exercise 5.2.2 $\{\Gamma^{\mu}{}_{00} \to \text{zero}^\mu,\ \Gamma^{\mu}{}_{0\nu\_} \to \frac{1}{2} h_\nu{}^\mu{}_{,0}\}$

Out[•]= $\frac{D^2\xi^\mu}{d\tau d\tau} + c^2\,\xi^\nu\,(-\frac{1}{4} h_\rho{}^\mu{}_{,0}\, h_\nu{}^\rho{}_{,0} + \text{zero}^\rho\,\Gamma^\mu{}_{\nu\rho} - \frac{1}{2}\partial_{x^0} h_\nu{}^\mu{}_{,0} + \partial_{x^\nu}\text{zero}^\mu) == 0$

Out[•]= $\frac{D^2\xi^\mu}{d\tau d\tau} + c^2\,\xi^\nu\,(-\frac{1}{4} h_\rho{}^\mu{}_{,0}\, h_\nu{}^\rho{}_{,0} - \frac{1}{2}\partial_{x^0} h_\nu{}^\mu{}_{,0}) == 0$

Rearrange

Out[•]= $\frac{D^2\xi^\mu}{d\tau d\tau} == \frac{1}{4} c^2\, h_\rho{}^\mu{}_{,0}\, h_\nu{}^\rho{}_{,0}\,\xi^\nu + \frac{1}{2} c^2\,\xi^\nu\,\partial_{x^0} h_\nu{}^\mu{}_{,0}$

This is the equation of geodesic deviation specialized
 for TT gauge and for the constant spatial coordinates solution.

In[•]:=
```
Print["Generic expression for a second order absolute derivative of a vector:"]
AbsoluteD[ξu[μ], {τ, τ}];
% == (% // ExpandAbsoluteD[labs, {{λ, ν}, {σ, ρ}}])
{TotalD[ξu[_], τ] → 0, TotalD[ξu[_], {τ, τ}] → 0};
Print["Assuming constant ξ^μ implies: ", %]
%%% /. %%
Print[
 "Substitute 4-velocity of the constant spatial coordinates solution and simplify"]
%% /. {TotalD[xu[μ_], τ] → uu[μ], TotalD[xu[μ_], {τ, τ}] → TotalD[uu[μ], τ]}
% /. TotalD[uu[_], τ] → 0 /. sol
% // KroneckerAbsorb[δ]
Print["Apply the rules from Exercise 5.2.2"]
%% /. Gammar
%[[2, 2, 4]] → c PartialD[labs][hdud[ν, μ, Dif[0]], xu[0]];
Print["Applying the chain rule to the total derivative: ", %]
%%% /. %%
Print["Is this equal to the result from the
   specialized equation of geodesic deviation previously obtained?"]
%% ==
 step1
```

Generic expression for a second order absolute derivative of a vector:

Out[•]= $\frac{D^2\xi^\mu}{d\tau d\tau} == \xi^\nu \frac{dx^\lambda}{d\tau}\frac{d\Gamma^\mu{}_{\lambda\nu}}{d\tau} + \frac{d^2\xi^\mu}{d\tau d\tau} + \Gamma^\mu{}_{\lambda\nu}\,(\xi^\nu\frac{d^2x^\lambda}{d\tau d\tau} + \frac{dx^\lambda}{d\tau}\frac{d\xi^\nu}{d\tau}) + \Gamma^\mu{}_{\sigma\rho}\frac{dx^\sigma}{d\tau}\,(\Gamma^\rho{}_{\lambda\nu}\,\xi^\nu\frac{dx^\lambda}{d\tau} + \frac{d\xi^\rho}{d\tau})$

Assuming constant ξ^μ implies: $\{\frac{d\xi\_}{d\tau} \to 0,\ \frac{d^2\xi\_}{d\tau d\tau} \to 0\}$

Out[•]= $\frac{D^2\xi^\mu}{d\tau d\tau} == \Gamma^\mu{}_{\lambda\nu}\,\xi^\nu\frac{d^2x^\lambda}{d\tau d\tau} + \Gamma^\mu{}_{\sigma\rho}\,\Gamma^\rho{}_{\lambda\nu}\,\xi^\nu\frac{dx^\lambda}{d\tau}\frac{dx^\sigma}{d\tau} + \xi^\nu\frac{dx^\lambda}{d\tau}\frac{d\Gamma^\mu{}_{\lambda\nu}}{d\tau}$

Substitute 4-velocity of the constant spatial coordinates solution and simplify

Out[•]= $\frac{D^2\xi^\mu}{d\tau d\tau} == u^\lambda u^\sigma\,\Gamma^\mu{}_{\sigma\rho}\,\Gamma^\rho{}_{\lambda\nu}\,\xi^\nu + \Gamma^\mu{}_{\lambda\nu}\,\xi^\nu\frac{du^\lambda}{d\tau} + u^\lambda\,\xi^\nu\frac{d\Gamma^\mu{}_{\lambda\nu}}{d\tau}$

Out[•]= $\frac{D^2\xi^\mu}{d\tau d\tau} == c^2\,\Gamma^\mu{}_{\sigma\rho}\,\Gamma^\rho{}_{\lambda\nu}\,\delta^\lambda{}_0\,\delta^\sigma{}_0\,\xi^\nu + c\,\delta^\lambda{}_0\,\xi^\nu\frac{d\Gamma^\mu{}_{\lambda\nu}}{d\tau}$

Out[•]= $\frac{D^2 \xi^\mu}{d\tau d\tau} == c^2 \Gamma^\mu{}_{0\rho} \Gamma^\rho{}_{0\nu} \xi^\nu + c \xi^\nu \frac{d\Gamma^\mu{}_{0\nu}}{d\tau}$

Apply the rules from Exercise 5.2.2

Out[•]= $\frac{D^2 \xi^\mu}{d\tau d\tau} == \frac{1}{4} c^2 h_\rho{}^\mu{}_{,0} h_\nu{}^\rho{}_{,0} \xi^\nu + \frac{1}{2} c \xi^\nu \frac{dh_\nu{}^\mu{}_{,0}}{d\tau}$

Applying the chain rule to the total derivative: $\frac{dh_\nu{}^\mu{}_{,0}}{d\tau} \to c \partial_{x^0} h_\nu{}^\mu{}_{,0}$

Out[•]= $\frac{D^2 \xi^\mu}{d\tau d\tau} == \frac{1}{4} c^2 h_\rho{}^\mu{}_{,0} h_\nu{}^\rho{}_{,0} \xi^\nu + \frac{1}{2} c^2 \xi^\nu \partial_{x^0} h_\nu{}^\mu{}_{,0}$

Is this equal to the result from the
  specialized equation of geodesic deviation previously obtained?

Out[•]= True

But since the metric is actually varying with time, a constant spatial separation vector ξ^μ doesn't imply a constant spatial separation *d*.

```
In[•]:= Print["A small, constant, spacelike spatial separation vector:"]
        ξu[μ] == (ξu[μ] // ToArrayValues[])
        % /. ξu[0] → 0
        Print[
         "Expression for small spatial separations. The sum is over the spatial coordinates."]
        d² == -gdd[i, j] × ξu[i] × ξu[j]
        eqnDD = % /. gdd[i_, j_] → ηdd[i, j] + hdd[i, j]
```

A small, constant, spacelike spatial separation vector:

Out[•]= $\xi^\mu == \{\xi^0, \xi^1, \xi^2, \xi^3\}$

Out[•]= $\xi^\mu == \{0, \xi^1, \xi^2, \xi^3\}$

Expression for small spatial separations. The sum is over the spatial coordinates.

Out[•]= $d^2 == -g_{ij} \xi^i \xi^j$

Out[•]= $d^2 == -(h_{ij} + \eta_{ij}) \xi^i \xi^j$

```
In[•]:= Print["Definition of a new and hopefully useful variable ξ̃ⁱ, see eq.(5.31),p.176:"]
        eqn[5, 31] = ξ̃u[i] == ξu[i] + 1 / 2 hud[i, k] × ξu[k];
        % // FrameBox // DisplayForm
        Print["Squared spatial length of ξ̃ⁱ. The sum is over the spatial coordinates."]
        aa = -ηdd[i, j] × ξ̃u[i] × ξ̃u[j]
        Print["Substituting for ξ̃ and expanding"]
        %% /. (Rule @@ eqn[5, 31] // LHSSymbolsToPatterns[{i}]);
        % // ExpandAll
        Print["Eliminating the second order term in h and simplifying"]
        %% /. HoldPattern[Tensor[h, __] × Tensor[h, __]] → 0
        % //
         MapLevelParts[(# // SimplifyTensorSum // SymmetrizeSlots[η, 2, {1, {1, 2}}]) &, {{2, 3}}]
        % // MapLevelParts[MetricSimplify[η], {2, {1, 2, 3}}]
        (eqn[5, 32] = % // IndexChange[{k, j}] // Simplify)
        Print["Is this equal to the previously obtained d²?"]
        %% == eqnDD[[2]]
        Print["Yessa! So we finally have:"]
```

```
ζu[i] × ζd[i] == aa == eqnDD[[2]] == eqnDD[[1]] // FrameBox // DisplayForm
Print["Compare with eq.(5.32),p.176.\n(See also Exercise 5.2.3,p.178,solution p.273.)"]
```

Definition of a new and hopefully useful variable ζ^i, see eq.(5.31),p.176:

Out[∘]//DisplayForm=

$$\zeta^i == \xi^i + \frac{1}{2} h^i{}_k \xi^k$$

Squared spatial length of ζ^i. The sum is over the spatial coordinates.

Out[∘]= $-\zeta^i \zeta^j \eta_{ij}$

Substituting for ζ and expanding

Out[∘]= $-\eta_{ij} \xi^i \xi^j - \frac{1}{2} h^j{}_k \eta_{ij} \xi^i \xi^k - \frac{1}{2} h^i{}_k \eta_{ij} \xi^j \xi^k - \frac{1}{4} h^i{}_k h^j{}_k \eta_{ij} (\xi^k)^2$

Eliminating the second order term in h and simplifying

Out[∘]= $-\eta_{ij} \xi^i \xi^j - \frac{1}{2} h^j{}_k \eta_{ij} \xi^i \xi^k - \frac{1}{2} h^i{}_k \eta_{ij} \xi^j \xi^k$

Out[∘]= $-\eta_{ij} \xi^i \xi^j - h^j{}_k \eta_{ij} \xi^i \xi^k$

Out[∘]= $-\eta_{ij} \xi^i \xi^j - h_{ik} \xi^i \xi^k$

Out[∘]= $-(h_{ij} + \eta_{ij}) \xi^i \xi^j$

Is this equal to the previously obtained d^2?

Out[∘]= True

Yessa! So we finally have:

Out[∘]//DisplayForm=

$$\zeta^i \zeta_i == -\zeta^i \zeta^j \eta_{ij} == -(h_{ij} + \eta_{ij}) \xi^i \xi^j == d^2$$

Compare with eq.(5.32),p.176.
(See also Exercise 5.2.3,p.178,solution p.273.)

FN: “So ζ^i may be regarded as a faithful position vector giving *correct* spatial separations when contracted with the Euclidean metric tensor δ_{ij} [= $-\eta_{ij}$, with i, j = 1, 2, 3].”

We can pull this all together and determine the constant spatial separation *d* by calculating ζ^i under the conditions considered here.

In[∘]:=
```
Print["We start from eq.(5.22)"]
Equal @@ eqn[5, 22, complex] /. σ → δ
Print["In TT gauge we have "h̄"^μν = h^μν, so"]
%% /. hb → h
Print["Substitute A^μν values from eq.(5.29)"]
%% /. Rule @@ eqn[5, 29]
Print["Lower an index"]
ηdd[ν, σ] # & /@ %%
Print[eqnh = MapAt[MetricSimplify[η], %, 1], "  with  ", eq[5, 28], " and α,β ∈ ℂ"]
Print["Substitute h^μ_σ in ζ^i"]
eqn[5, 31]
(eqζ = % /. (Rule @@ eqnh // LHSSymbolsToPatterns[{μ, σ}])) // FrameBox // DisplayForm
Print["General case (elliptical polarisation)"]
```

We start from eq.(5.22)

Out[•]= $\bar{h}^{\mu\nu} == e^{i k_\delta x^\delta} A^{\mu\nu}$

In TT gauge we have $\bar{h}^{\mu\nu} = h^{\mu\nu}$, so

Out[•]= $h^{\mu\nu} == e^{i k_\delta x^\delta} A^{\mu\nu}$

Substitute $A^{\mu\nu}$ values from eq.(5.29)

Out[•]= $h^{\mu\nu} == e^{i k_\delta x^\delta} (\alpha\, e1^{\mu\nu} + \beta\, e2^{\mu\nu})$

Lower an index

Out[•]= $h^{\mu\nu}\, \eta_{\nu\sigma} == e^{i k_\delta x^\delta} (\alpha\, e1^{\mu\nu} + \beta\, e2^{\mu\nu})\, \eta_{\nu\sigma}$

$h^{\mu}{}_{\sigma} == e^{i k_\delta x^\delta} (\alpha\, e1^{\mu\nu} + \beta\, e2^{\mu\nu})\, \eta_{\nu\sigma}$ with $\{e1^{\mu\nu} == \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, e2^{\mu\nu} == \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}\}$ and $\alpha, \beta \in \mathbb{C}$

Substitute $h^{\mu}{}_{\sigma}$ in ζ^i

Out[•]= $\zeta^i == \xi^i + \frac{1}{2} h^i{}_k\, \xi^k$

Out[•]//DisplayForm=

$$\boxed{\zeta^i == \xi^i + \frac{1}{2} e^{i k_\delta x^\delta} (\alpha\, e1^{i\nu} + \beta\, e2^{i\nu})\, \eta_{\nu k}\, \xi^k}$$

General case (elliptical polarisation)

A) David Park: "The following calculation shows that in the TT gauge there is no change in the z component of ζ^i (i=3). This shows that the gravitational wave is *transverse* since we picked k_α so the wave was propagating in the z direction."

In[•]:=
```
Print["Recall that we defined"]
ku[μ];
% == ToArrayValues[][%]
Print["Put i=3 (for the z direction)
   in the previous general expression for ζ^i and simplify."]
eqζ /. i → 3
(% // MapLevelParts[ToArrayValues[], {2, 2, {3, 4, 5}}]);
%[[1]] == %[[2]] == "constant" // FrameBox // DisplayForm
```

Recall that we defined

Out[•]= $k^\mu == \{k, 0, 0, k\}$

Put i=3 (for the z direction) in the previous general expression for ζ^i and simplify.

Out[•]= $\zeta^3 == \xi^3 + \frac{1}{2} e^{i k_\delta x^\delta} (\alpha\, e1^{3\nu} + \beta\, e2^{3\nu})\, \eta_{\nu k}\, \xi^k$

Out[•]//DisplayForm=

$$\boxed{\zeta^3 == \xi^3 == \text{constant}}$$

B) Let's calculate the x and y components of ζ^i (i=1,2) in the elliptical polarisation case (general case) too.

In[•]:=
```
Print["General expression for ζ^i with
   coordinates (ct,x,y,z) used in the exponential function"]
eqζ
% /. Exp[a_] :→ (Exp[a // ToArrayValues[]]);
```

```
% // UseCoordinates[{c t, x, y, z}]
Print["Expansion to array values"]
ToArrayValues[] /@ %%
Print["(We got ξ³ = ξ³ again.) Taking
   the x and y components, which are the only ones that vary"]
Take[#, {2, 3}] & /@ %%
Print["Take the real part, considering that α,β ∈ ℂ"]
(eqξ12 = MapAt[Re[Assuming[α ∈ Complexes && β ∈ Complexes, ComplexExpand[#]]] &, %%, 2] /.
     {Cos[a_] :> Cos[Collect[a, k]],
      Sin[a_] :> Sin[Collect[a, k]]}) // FrameBox // DisplayForm
Print["General case (elliptical polarisation)"]
```

General expression for ξ^{i} with coordinates (ct,x,y,z) used in the exponential function

Out[•]= $\xi^{i} == \xi^{i} + \frac{1}{2} e^{i k_\delta x^\delta} (\alpha\, e1^{i\nu} + \beta\, e2^{i\nu})\, \eta_{\nu k}\, \xi^{k}$

Out[•]= $\xi^{i} == \xi^{i} + \frac{1}{2} e^{i c k t - i k z} (\alpha\, e1^{i\nu} + \beta\, e2^{i\nu})\, \eta_{\nu k}\, \xi^{k}$

Expansion to array values

Out[•]= $\{\xi^0, \xi^1, \xi^2, \xi^3\} == \{\xi^0, \xi^1 - \frac{1}{2} e^{i c k t - i k z} \alpha\, \xi^1 - \frac{1}{2} e^{i c k t - i k z} \beta\, \xi^2, -\frac{1}{2} e^{i c k t - i k z} \beta\, \xi^1 + \xi^2 + \frac{1}{2} e^{i c k t - i k z} \alpha\, \xi^2, \xi^3\}$

(We got $\xi^3 = \xi^3$ again.) Taking the x and y components, which are the only ones that vary

Out[•]= $\{\xi^1, \xi^2\} == \{\xi^1 - \frac{1}{2} e^{i c k t - i k z} \alpha\, \xi^1 - \frac{1}{2} e^{i c k t - i k z} \beta\, \xi^2, -\frac{1}{2} e^{i c k t - i k z} \beta\, \xi^1 + \xi^2 + \frac{1}{2} e^{i c k t - i k z} \alpha\, \xi^2\}$

Take the real part, considering that α,β ∈ ℂ

Out[•]//DisplayForm=

$$\begin{aligned}\{\xi^1, \xi^2\} == \{&-\mathrm{Im}[-\tfrac{1}{2} \alpha \,\mathrm{Sin}[k (c t - z)]\, \xi^1 - \tfrac{1}{2} \beta\, \mathrm{Sin}[k (c t - z)]\, \xi^2] + \\ &\mathrm{Re}[\xi^1 - \tfrac{1}{2} \alpha\, \mathrm{Cos}[k (c t - z)]\, \xi^1 - \tfrac{1}{2} \beta\, \mathrm{Cos}[k (c t - z)]\, \xi^2], \\ &-\mathrm{Im}[-\tfrac{1}{2} \beta\, \mathrm{Sin}[k (c t - z)]\, \xi^1 + \tfrac{1}{2} \alpha\, \mathrm{Sin}[k (c t - z)]\, \xi^2] + \\ &\mathrm{Re}[-\tfrac{1}{2} \beta\, \mathrm{Cos}[k (c t - z)]\, \xi^1 + \xi^2 + \tfrac{1}{2} \alpha\, \mathrm{Cos}[k (c t - z)]\, \xi^2]\}\end{aligned}$$

General case (elliptical polarisation)

We take advantage of this general relation in the following animation showing what happens to a set of free particles initially at rest and arranged in a circle (“dust” ring) when it is hit orthogonally by an elliptically polarized gravitational wave. All the parameters used are fictitious, because we have not yet performed any calculation that can help us estimate their real values. In any case, it is easy to guess that the real parameters would not produce any notable effects in the visualization, so a “little” exaggeration is due. By varying the parameters α and β, the effects of linear and circular polarization, which are the special cases of the general case given by elliptical polarization, can be explored too.

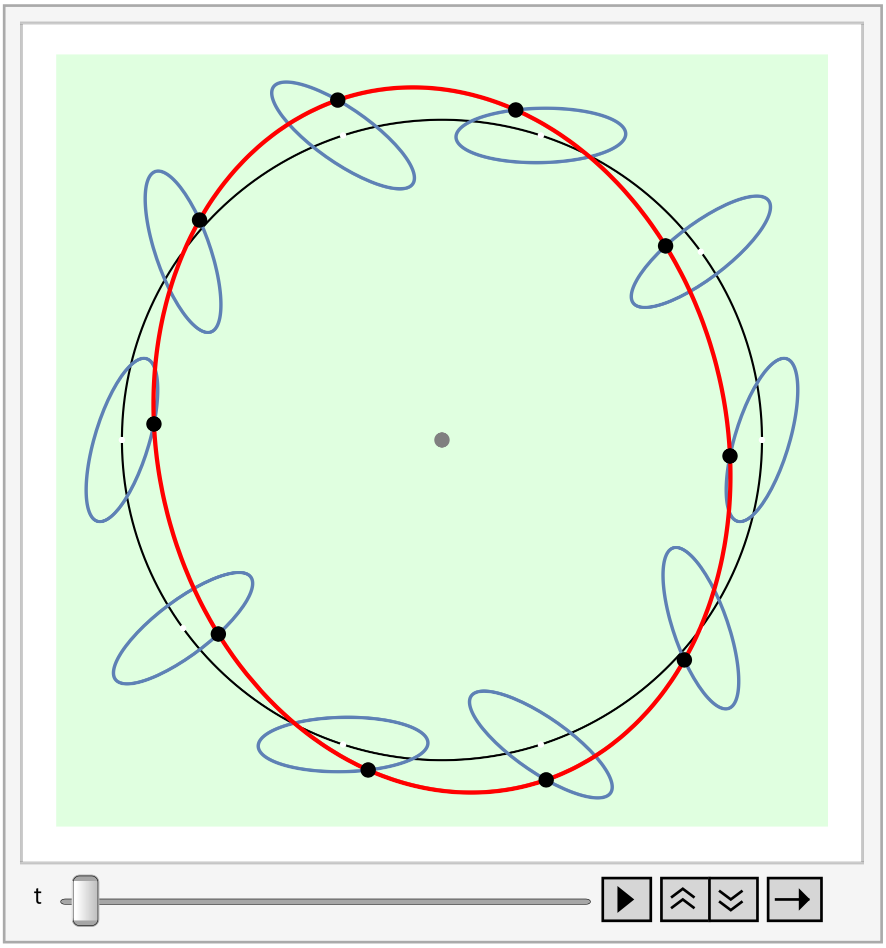
```
In[•]:= p = {α → .3, β → 0}; (* example of ⊕ linear polarization *)

In[•]:= p = {α → 0, β → .3}; (* example of ⊗ linear polarization *)

In[•]:= p = {α → .3, β → .3}; (* example of general linear polarization *)

In[•]:= p = {α → .3, β → .3 i}; (* example of circular polarization *)
```

In[ ]:=
```
p = {α → (.2 + .1 i), β → (.1 + .5 i)}; (* example of elliptical polarization *)
```

In[ ]:=
```
n = 10; (* number of particles *)
```

```
particle[α_, β_, θ_][t_] =
  eqξ12[[2]] /. {c → 1, z → 0, k → 1, ξu[1] → Cos[θ], ξu[2] → Sin[θ]} // FullSimplify;
iniPosition = Graphics[
   {White, Table[Point[particle[0, 0, θ][0]], {θ, 0, 2 π - 2 π / n, 2 π / n}]}];
particlePoints[t_] := Graphics[{PointSize[.02],
   Table[Point[particle[α, β, θ][t] /. p], {θ, 0, 2 π - 2 π / n, 2 π / n}]}]
particleOrbits = Table[ParametricPlot[particle[α, β, θ][t] /. p, {t, 0, 2 π}],
   {θ, 0, 2 π - 2 π / n, 2 π / n}];
polarisationEllipse[t_] := ParametricPlot[particle[α, β, θ][t] /. p,
   {θ, 0, 2 π}, PlotStyle → Directive[Red, Thick]];

Animate[Show[
  Graphics[{Circle[{0, 0}]}],
  particleOrbits,
  iniPosition,
  polarisationEllipse[t],
  Graphics[{PointSize[.02], Gray, Point[{0, 0}]}],
  particlePoints[t],
  ImageSize → Medium, PlotRange → 1.2 {{-1, 1}, {-1, 1}}, AspectRatio → Automatic,
  Background → LightGreen], {t, 0, 2 π}, ControlPlacement → Bottom, AnimationRunning → False]
```

Out[ ]=

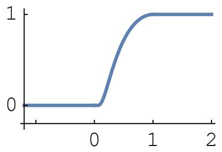


The free particles (black dots) are initially arranged in a circle (black line) at positions marked with small white dots; their motion are referred to the particle at the center (gray dot) which is the origin of the coordinate system $\{\zeta^1, \zeta^2\}$ in a plane orthogonal to the propagation direction of the gravitational wave. Under the influence of an elliptical polarized gravitational wave the free particles move on fixed little ellipses (blue) centered at their initial positions and synchronized to form the polarisation ellipse (red) which varies over time. For linear or circular polarized gravitational waves the little ellipses degenerate to short straight segments or little circles respectively.

If we switch on the gravitational wave slowly (applying for example an adapted standard mollifier to the parameters α and β), we can see how the particle swings gradually from the initial rest position into the final elliptical orbit.

```
In[•]:= Mollifier[t_] := ⎧ 0                          t ≤ 0
                         ⎨ Exp[1 + -1/(1-(t-1)^2)]    0 < t < 1
                         ⎩ 1                          t ≥ 1

        Plot[Mollifier[t], {t, -1.2, 2}, ImageSize → Tiny,
         AxesOrigin → {-1.2, -.2}, Ticks → {{-1, 0, 1, 2}, {0, 1}}]
        ParametricPlot[particle[Mollifier[t] α, Mollifier[t] β, π / 4][t] /. p,
         {t, 0 , 2.2 π}, ImageSize → Small]
```

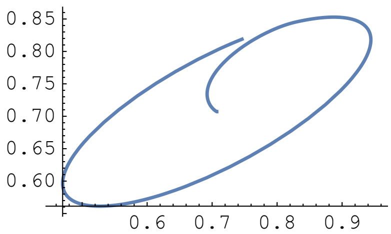
Out[•]=



••• General: Exp[-3544.06] is too small to represent as a normalized machine number; precision may be lost.

••• General: Exp[-3544.06] is too small to represent as a normalized machine number; precision may be lost.

Out[•]=



5.3 Simple generation and detection p. 178-182

```
In[1]:= Needs["Notation`"]
        Notation[h̄ ⟺ hb]

        Needs["TGeneralRelativity1`GeneralRelativity`"]
        $PrePrint =.
        DeclareBaseIndices[{0, 1, 2, 3}]
        DeclareZeroTensor[zero]
        labs = {x, δ, g, Γ};
        DefineTensorShortcuts[
         {{x, zero, k}, 1},
         {{g, η, δ, hb, T, A}, 2},
         {{T, hb}, 3}]
        SetTensorValues[δud[i, j], IdentityMatrix[NDim]]
        SetTensorValueRules[ηdd[i, j], DiagonalMatrix[{1, -1, -1, -1}]]
        SetTensorValueRules[ηuu[i, j], DiagonalMatrix[{1, -1, -1, -1}]]
```

```
(* Another little adjustment...*)
(
 Unprotect[PartialD];
 PartialD[l_][tensor_, c t] := c^-1 PartialD[l][tensor, t];
 Protect[PartialD];
)

(* Notation for formal integrals when
 we don't want Mathematica to evaluate anything *)
integral[intvar_][intregion_][integrand_] :=
 RowBox[{UnderscriptBox["∫", intregion], integrand, ⅆ intvar}]

 (* linear polarization matrices, TT gauge in x3-direction *)
e1 = (1 0 0; 0 -1 0; 0 0 0); e2 = (0 1 0; 1 0 0; 0 0 0);
```

Get: Cannot open Graphics`Colors`.

Needs: Context Graphics`Colors` was not created when Needs was evaluated.

SetDelayed: Tag Symmetric in Symmetric[args__ /; Length[{args}] > 1][term_] is Protected.

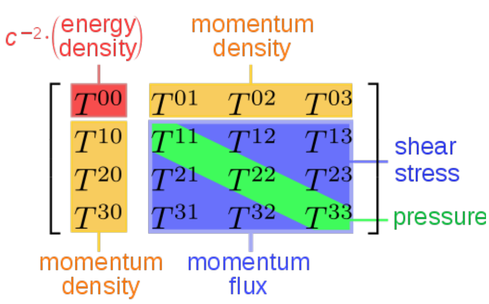
SetDelayed: Tag Symmetric in Symmetric[ind_][term_] is Protected.

1) Derivation of the far zone gravitational radiation field of a mass system

We want to study the gravitational waves generated by a mass system. We have to start from eq.(5.19), p.172, again, but this time we must also consider the *energy-momentum-stress tensor* (or *stress tensor* for short) $T^{\mu\nu}$ which describes the source of the gravitation field.

Wikipedia: “The stress–energy tensor, sometimes called the stress–energy–momentum tensor or the energy–momentum tensor, is a tensor quantity in physics that describes the density and flux of energy and momentum in spacetime, generalizing the stress tensor of Newtonian physics. It is an attribute of matter, radiation, and non-gravitational force fields. This density and flux of energy and momentum are the sources of the gravitational field in the Einstein field equations of general relativity, just as mass density is the source of such a field in Newtonian gravity.”

In[∘]:=



In[15]:=

```
Print["Recall eq.(5.19),p.172."]
eqn[5, 19] = -DAlembertian[α, β][hbuu[μ, ν]] == -2 κ Tuu[μ, ν];
{%[[1]] == ((%[[1]] // ExpandPartialD[labs] // EinsteinSum[]) /. TensorValueRules[η]) ==
    %[[2]], hbuu[μ, ν] == hbuu[μ, ν][Sequence @@ (xu[μ] // ToArrayValues[])],
   Tuu[μ, ν] == Tuu[μ, ν][Sequence @@ (xu[μ] // ToArrayValues[])],
   "\nThe gauge condition holds."} // TableForm // FrameBox // DisplayForm
```

Recall eq.(5.19),p.172.

Out[17]//DisplayForm=

$$-\overline{h}^{\mu\nu}{}_{,\alpha,\beta}\,\eta^{\alpha\beta} == -\partial_{x^0,x^0}\overline{h}^{\mu\nu} + \partial_{x^1,x^1}\overline{h}^{\mu\nu} + \partial_{x^2,x^2}\overline{h}^{\mu\nu} + \partial_{x^3,x^3}\overline{h}^{\mu\nu} == -2\,\kappa\,T^{\mu\nu}$$
$$\overline{h}^{\mu\nu} == \overline{h}^{\mu\nu}[x^0, x^1, x^2, x^3]$$
$$T^{\mu\nu} == T^{\mu\nu}[x^0, x^1, x^2, x^3]$$

The gauge condition holds.

This is a *second order nonhomogeneous linear PDE with constant coefficients* (in four independent variables). It is convenient to solve it with the *Green's function method;* as integral kernel we choose the *retarded Green's function* which assures causality and finite propagation speed.

In[18]:=

```
Print[
 "Enter the appropriate quantities in the Green's function method solution formula:"]
hbuu[μ, ν][Sequence @@ (xu[μ] // ToArrayValues[])] == (Inactivate[
   ∫_{-∞}^{∞}∫_{-∞}^{∞}∫_{-∞}^{∞}∫_{-∞}^{∞} HoldForm[HoldForm[-
     DiracDelta[x⁰ - τ - Sqrt[(x¹ - X)² + (x² - Y)² + (x³ - Z)²]] / (4 π Sqrt[(x¹ - X)² + (x² - Y)² + (x³ - Z)²])] ×
       HoldForm[-2 κ T^μν[τ, X, Y, Z]]] dτ dX dY dZ, Integrate] // TraditionalForm)
Print["Carry out the innermost integration:"]
((ReleaseHold //@ %%) // Activate) /. κ/(2 π) → HoldForm[κ/(2 π)] /. TraditionalForm → StandardForm
```

Enter the appropriate quantities in the Green's function method solution formula:

Out[19]=

$$\overline{h}^{\mu\nu}[x^0, x^1, x^2, x^3] == \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} -\frac{\delta\left(x^0-\tau-\sqrt{(x^1-X)^2+(x^2-Y)^2+(x^3-Z)^2}\right)}{4\,\pi\,\sqrt{(x^1-X)^2+(x^2-Y)^2+(x^3-Z)^2}}\,(-2\,\kappa\,T^{\mu\nu}(\tau, X, Y, Z))\,d\tau\,dX\,dY\,dZ$$

Carry out the innermost integration:

Out[21]=

$$\overline{h}^{\mu\nu}[x^0, x^1, x^2, x^3] == \frac{\kappa}{2\,\pi}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\frac{T^{\mu\nu}\left[x^0-\sqrt{(X-x^1)^2+(Y-x^2)^2+(Z-x^3)^2}, X, Y, Z\right]}{\sqrt{(X-x^1)^2+(Y-x^2)^2+(Z-x^3)^2}}\,dX\,dY\,dZ$$

Let's make some notation changes to match the FN notation.

In[22]:=

```
hbuu[μ, ν][xu[0], x⃗] == HoldForm[κ/(2 π)] × integral[V′][ℝ³][
   Tuu[μ, ν][xu[0] - Norm[x⃗ - x⃗′], x⃗′] / Norm[x⃗ - x⃗′]] // FrameBox // DisplayForm
Print["Compare with eq.(5.35),p.178."]
{Norm[x⃗ - x⃗′] → r, xu[0] → c t, HoldForm[κ/(2 π)] → HoldForm[-4 G/c⁴]};
Print["With the substitutions ", %, " we get"]
%%%%[[1]] /. %% // DisplayForm
```

Out[22]//DisplayForm=

$$\overline{h}^{\mu\nu}[x^0, \vec{x}] == \frac{\kappa}{2\,\pi}\int_{\mathbb{R}^3}\frac{T^{\mu\nu}[-\mathrm{Norm}[\vec{x}-\vec{x}'] + x^0, \vec{x}']}{\mathrm{Norm}[\vec{x}-\vec{x}']}\,dV'$$

Compare with eq.(5.35),p.178.

With the substitutions {Norm[$\vec{x}-\vec{x}'$] → r, x^0 → c t, $\frac{\kappa}{2\pi}$ → $-\frac{4G}{c^4}$} we get

Out[26]//DisplayForm=

$$\bar{h}^{\mu\nu}[c\,t,\vec{x}] == -\frac{4G}{c^4}\int_{\mathbb{R}^3}\frac{T^{\mu\nu}[-r+c\,t,\vec{x}']}{r}\,dV'$$

FN: “Suppose now that the source is some sort of matter distribution localized near the origin O, and that the source particles have speeds which are small compared with *c*. If we take our field point at a distance *r* from O that is large compared to the maximum displacements of the source particles from O, then equation (5.35) may be approximated by”

In[27]:=

```
Print["r = |x⃗-x⃗′| ≈ constant = r₀ =
   approximated distance of the field point from the source = |x⃗|"]
hbuu[μ, ν][c t, x⃗] == HoldForm[-4 G / (c⁴ r₀)] × integral[V′]["source\nregion"][
     Tuu[μ, ν][c t - r₀, x⃗′]] // FrameBox // DisplayForm
Print["Compare with eq.(5.36),p.178."]
```

r = |$\vec{x}-\vec{x}'$| ≈ constant = r_0 = approximated distance of the field point from the source = |$\vec{x}$|

Out[28]//DisplayForm=

$$\bar{h}^{\mu\nu}[c\,t,\vec{x}] == -\frac{4G}{c^4 r_0}\int_{\text{source region}} T^{\mu\nu}[c\,t-r_0,\vec{x}']\,dV'$$

Compare with eq.(5.36),p.178.

FN: “This approximation is appropriate for looking at the gravitational radiation in the *far zone* or *wave zone* [...] we expect that (over not too large regions of space) it looks like a plane wave”

David Park: “If, in the far zone, the solution looks like a plane wave then Exercise 5.2.1, p.178, showed that the radiative part of $\bar{h}^{\mu\nu}$ is completely determined by its spatial part $\bar{h}^{ij}$.”

Brute-force *Mathematica* solution:

In[30]:=

```
Print["Plane wave definition eq.(5.22),p.172:"]
hbuu[μ, ν] == Re[Auu[μ, ν] Exp[ⅈ kd[σ] × xu[σ]]]

Print["\nNull 4-wave vector k^μ condition:"]
kd[ν] × kd[μ] × ηuu[μ, ν] == 0
% // ToArrayValues[]
Print["If k^μ is not a zero vector, then k⁰≠0."]

Print["\nEq.(5.23),p.173:"]
Auu[μ, ν] × kd[ν] == zerou[μ]
Print["Expand equations and symmetrize slots on A"]
ToArrayValues[] /@ %% // Flatten
% // SymmetrizeSlots[A, 2, {1, {1, 2}}]
Print["Solve equations for required quantities"]
Solve[%%, {Auu[0, 0], Auu[0, 1], Auu[0, 2], Auu[0, 3]}][[1]] // Distribute // ColumnForm //
  FrameBox // DisplayForm
```

Plane wave definition eq.(5.22),p.172:

Out[31]=

$\bar{h}^{\mu\nu} == \mathrm{Re}[e^{i k_\sigma x^\sigma} A^{\mu\nu}]$

Null 4-wave vector k^μ condition:

Out[33]=

$k_\mu k_\nu \eta^{\mu\nu} == 0$

Out[34]=

$(k_0)^2 - (k_1)^2 - (k_2)^2 - (k_3)^2 == 0$

If k^μ is not a zero vector, then $k^0 \neq 0$.

Eq.(5.23),p.173:

Out[37]=

$A^{\mu\nu} k_\nu == \mathrm{zero}^\mu$

Expand equations and symmetrize slots on A

Out[39]=

$\{A^{00} k_0 + A^{01} k_1 + A^{02} k_2 + A^{03} k_3,\ A^{10} k_0 + A^{11} k_1 + A^{12} k_2 + A^{13} k_3,\ A^{20} k_0 + A^{21} k_1 + A^{22} k_2 + A^{23} k_3,\ A^{30} k_0 + A^{31} k_1 + A^{32} k_2 + A^{33} k_3\} == \{0, 0, 0, 0\}$

Out[40]=

$\{A^{00} k_0 + A^{01} k_1 + A^{02} k_2 + A^{03} k_3,\ A^{01} k_0 + A^{11} k_1 + A^{12} k_2 + A^{13} k_3,\ A^{02} k_0 + A^{12} k_1 + A^{22} k_2 + A^{23} k_3,\ A^{03} k_0 + A^{13} k_1 + A^{23} k_2 + A^{33} k_3\} == \{0, 0, 0, 0\}$

Solve equations for required quantities

Out[42]//DisplayForm=

$$A^{00} \to -\frac{-A^{11} (k_1)^2 - 2 A^{12} k_1 k_2 - A^{22} (k_2)^2 - 2 A^{13} k_1 k_3 - 2 A^{23} k_2 k_3 - A^{33} (k_3)^2}{(k_0)^2}$$

$$A^{01} \to -\frac{A^{11} k_1 + A^{12} k_2 + A^{13} k_3}{k_0}$$

$$A^{02} \to -\frac{A^{12} k_1 + A^{22} k_2 + A^{23} k_3}{k_0}$$

$$A^{03} \to -\frac{A^{13} k_1 + A^{23} k_2 + A^{33} k_3}{k_0}$$

All time components $A^{0\nu}$ are expressed in terms of the spatial components A^{ij}, hence all time components $\bar{h}^{0\nu}$ are expressed in terms of the spatial components $\bar{h}^{ij}$. QED.

FN: "It follows that we need only consider $\int T^{ij}\, dV$ (at the retarded time), a neat expression for which may obtained as follows."

1) FN: "The stress tensor of the source satisfies the [special-relativistic] conservation equation $T^{\mu\nu}{}_{,\nu} == 0$ (see Exercise 5.1.4, p.173)"

In[43]:=

```
Print["Eq.(5.19) rearranged"]
eqn[5, 19]
(#/(-2 κ)) &/@ % // Reverse
Print["Taking the divergence"]
PartialD[#, ν] &/@ %%
Print["η derivatives are zero"]
%% /. Tensor[η, _, List[__, Dif[_]]] → 0
Print["Reindexing and symmetrising"]
MapAt[(# // IndexChange[{{ν, α}, {α, σ}}]) &, %%, 2]
% // SymmetrizeSlots[hb, 5, {1, {3, 4, 5}}]
```

```
Print["But ", hbuud[μ, α, Dif[α]] == 0,
 " (guage condition, see eq.(5.15),p.172) and so are higher derivatives"]
%% /. Tensor[hb, _, _] _ → zerou[μ];
Print["Finally we get the conservation equation: ", % // FrameBox // DisplayForm]
Print["Partial sum on ν=0"]
%%% // PartialSum[0, {k}]
Print["Partial array on μ=0"]
({eqn[5, 37], eqn[5, 38]} = %% // PartialArray[0, {i}] // Thread) // TableForm // FrameBox //
 DisplayForm
Print["- See eq.(5.37) and (5.38),p.179. -"]
Print["Substituting x^0 = c t and turn into rules for use below :"]
eqn[5, 37];
# - Part[%, 1, 1] & /@ %;
(% // ExpandPartialD[labs]) /. xu[0] → c t ;
rule[5, 37] = Rule @@ % // LHSSymbolsToPatterns[{k}]
eqn[5, 38];
# - Part[%, 1, 1] & /@ %;
% /. zerou[i_] + a_ → a;
(% // ExpandPartialD[labs]) /. xu[0] → c t;
rule[5, 38] = Rule @@ % // LHSSymbolsToPatterns[{i, k}]
```

Eq.(5.19) rearranged

Out[44]=

$$-\bar{h}^{\mu\nu}{}_{,\alpha,\beta}\,\eta^{\alpha\beta} == -2\,\kappa\,T^{\mu\nu}$$

Out[45]=

$$T^{\mu\nu} == \frac{\bar{h}^{\mu\nu}{}_{,\alpha,\beta}\,\eta^{\alpha\beta}}{2\,\kappa}$$

Taking the divergence

Out[47]=

$$T^{\mu\nu}{}_{,\nu} == \frac{\bar{h}^{\mu\nu}{}_{,\alpha,\beta,\nu}\,\eta^{\alpha\beta} + \bar{h}^{\mu\nu}{}_{,\alpha,\beta}\,\eta^{\alpha\beta}{}_{,\nu}}{2\,\kappa}$$

η derivatives are zero

Out[49]=

$$T^{\mu\nu}{}_{,\nu} == \frac{\bar{h}^{\mu\nu}{}_{,\alpha,\beta,\nu}\,\eta^{\alpha\beta}}{2\,\kappa}$$

Reindexing and symmetrising

Out[51]=

$$T^{\mu\nu}{}_{,\nu} == \frac{\bar{h}^{\mu\alpha}{}_{,\sigma,\beta,\alpha}\,\eta^{\sigma\beta}}{2\,\kappa}$$

Out[52]=

$$T^{\mu\nu}{}_{,\nu} == \frac{\bar{h}^{\mu\alpha}{}_{,\alpha,\beta,\sigma}\,\eta^{\sigma\beta}}{2\,\kappa}$$

But $\bar{h}^{\mu\alpha}{}_{,\alpha} == 0$ (guage condition, see eq.(5.15),p.172) and so are higher derivatives

Finally we get the conservation equation: $\boxed{T^{\mu\nu}{}_{,\nu} == \text{zero}^{\mu}}$

Partial sum on ν=0

Out[57]=

$T^{\mu 0}{}_{,0} + T^{\mu k}{}_{,k} == \text{zero}^{\mu}$

Partial array on μ=0

Out[59]//DisplayForm=

$T^{00}{}_{,0} + T^{0k}{}_{,k} == 0$
$T^{i0}{}_{,0} + T^{ik}{}_{,k} == \text{zero}^{i}$

- See eq.(5.37) and (5.38),p.179. -

Substituting $x^0 = c\,t$ and turn into rules for use below :

Out[65]=

$$\partial_{x^{k}\_}\, T^{0k}\_ \to -\frac{\partial_t T^{00}}{c}$$

Out[70]=

$$\partial_{x^{k}\_}\, T^{i\_k}\_ \to -\frac{\partial_t T^{i0}}{c}$$

Hans Stephani. *Relativity*, 2004, p.217: "Since no covariant derivatives occur in [$T^{\mu\nu}{}_{,\nu} == 0$], in the linearized theory the gravitational field has no influence upon the motion of the matter producing the field. One can specify the energy-momentum tensor arbitrarily, provided only that [$T^{\mu\nu}{}_{,\nu} == 0$] is satisfied, and calculate the gravitational field associated with it. This apparently advantageous property of the linearized theory has, however, the consequence that the gravitational field corresponding to the exact solution can deviate considerably from that of the linearized theory if the sources of the field (under the influence of their own gravitational field) move in a manner rather different from that supposed. It is therefore quite possible that there is no exact solution whose essential features agree with those of a particular solution of the linearized theory."

(This is an example of the many and often well hidden subtleties that must be taken into account when working with General Relativity...)

2) FN: "Consider the integral identity" ...

Note: David Park: "the [volume integral of the] divergence of a quantity can be replaced by a surface integral of the quantity dotted into a normal vector [Gauss's theorem or divergence theorem]. By making the surface large enough to include all the source particles, we will have $T^{ik} == 0$ and so the integral will be zero."

In[71]:=

```
Print["Expanding the partial derivative and simplifying"]
PartialD[labs][NestedTensor[Tuu[i, k] × xu[j]], xu[k]];
Distribute /@ (Inactivate[∫# dV, Integrate] & /@ (% == (% // UnnestTensor)))
% // KroneckerAbsorb[δ]
Print["Gauss's theorem on lhs and rearrangement"]
%% /. %%[[1]] → 0
# - %[[2, 2]] & /@ % // Reverse
Print["Apply eq.(5.38)"]
Tint =
 %% /. rule[5, 38] /. Inactive[Integrate][(-1 / c) a_, d_] → (-1 / c) Inactive[Integrate][a, d]
```

Expanding the partial derivative and simplifying

Out[73]=

$$\int \partial_{x^k} T^{ik}\, x^j\, dV == \int T^{ik}\, \delta^j{}_k\, dV + \int x^j\, \partial_{x^k} T^{ik}\, dV$$

Out[74]=

$$\int \partial_{x_k} T^{ik}\, x^j\, dV == \int T^{ij}\, dV + \int x^j\, \partial_{x_k} T^{ik}\, dV$$

Gauss's theorem on lhs and rearrangement

Out[76]=

$$0 == \int T^{ij}\, dV + \int x^j\, \partial_{x_k} T^{ik}\, dV$$

Out[77]=

$$\int T^{ij}\, dV == -\int x^j\, \partial_{x_k} T^{ik}\, dV$$

Apply eq.(5.38)

Out[79]=

$$\int T^{ij}\, dV == \frac{\int x^j\, \partial_t T^{i0}\, dV}{c}$$

3) FN: "Interchanging *i* and *j* and adding gives"

In[80]:=

```
Inner[Plus, Tint, IndexChange[{{i, j}, {j, i}}] /@ Tint, Equal]
Print["Symmetrize T^μν"]
MapAt[SymmetrizeSlots[T, 2, {1, {1, 2}}], %%, {{1}, {2}}];
% /. IgnoringInactive[ Integrate[α_, d_] / c + Integrate[β_, d_] / c] →
  Inactive[Integrate][α + β, d] / c
Divide[#, 2] & /@ %
Print["Leibniz integral rule on rhs "]
eqn[5, 39] = MapAt[Composition[Inactivate[∂t #, D] &,
   ReplaceAll[#, HoldPattern[PartialD[_][T_, t]] → T] &], %%, 2]
Print["Compare with eq.(5.39),p.179."]
```

Out[80]=

$$\int T^{ij}\, dV + \int T^{ji}\, dV == \frac{\int x^j\, \partial_t T^{i0}\, dV}{c} + \frac{\int x^i\, \partial_t T^{j0}\, dV}{c}$$

Symmetrize $T^{\mu\nu}$

Out[83]=

$$2 \int T^{ij}\, dV == \frac{\int (x^j\, \partial_t T^{0i} + x^i\, \partial_t T^{0j})\, dV}{c}$$

Out[84]=

$$\int T^{ij}\, dV == \frac{\int (x^j\, \partial_t T^{0i} + x^i\, \partial_t T^{0j})\, dV}{2\, c}$$

Leibniz integral rule on rhs

Out[86]=

$$\int T^{ij}\, dV == \partial_t \frac{\int (T^{0j}\, x^i + T^{0i}\, x^j)\, dV}{2\, c}$$

Compare with eq.(5.39),p.179.

4) FN: "But"

In[88]:=

```
Print["Expanding the partial derivative and simplifying"]
PartialD[labs][NestedTensor[Tuu[0, k] × xu[i] × xu[j]], xu[k]];
Distribute /@ (Inactivate[∫# dV, Integrate] & /@ (% == (% // UnnestTensor)))
Expand //@ % // KroneckerAbsorb[δ]
```

```
Print["Gauss's theorem on lhs and rearrangement"]
%% /. %%[[1]] → 0
# - %[[2, 2]] & /@ % // Reverse
Print["Apply eq.(5.37)"]
%% /. rule[5, 37] /. Inactive[Integrate][(-1 / c) α_, d_] → (-1 / c) Inactive[Integrate][α, d]
Print["Leibniz integral rule on rhs "]
eqn[5, 40] = MapAt[Composition[Inactivate[∂t #, D] &,
   ReplaceAll[#, HoldPattern[PartialD[_][T_, t]] → T] &], %%, 2]
Print["Compare with eq.(5.40),p.179."]
```

Expanding the partial derivative and simplifying

Out[90]=

$$\int \partial_{x^k} T^{0k} x^i x^j \, dV == \int T^{0k} (x^j \delta^i{}_k + x^i \delta^j{}_k) \, dV + \int x^i x^j \partial_{x^k} T^{0k} \, dV$$

Out[91]=

$$\int \partial_{x^k} T^{0k} x^i x^j \, dV == \int (T^{0j} x^i + T^{0i} x^j) \, dV + \int x^i x^j \partial_{x^k} T^{0k} \, dV$$

Gauss's theorem on lhs and rearrangement

Out[93]=

$$0 == \int (T^{0j} x^i + T^{0i} x^j) \, dV + \int x^i x^j \partial_{x^k} T^{0k} \, dV$$

Out[94]=

$$\int (T^{0j} x^i + T^{0i} x^j) \, dV == -\int x^i x^j \partial_{x^k} T^{0k} \, dV$$

Apply eq.(5.37)

Out[96]=

$$\int (T^{0j} x^i + T^{0i} x^j) \, dV == \frac{\int x^i x^j \partial_t T^{00} \, dV}{c}$$

Leibniz integral rule on rhs

Out[98]=

$$\int (T^{0j} x^i + T^{0i} x^j) \, dV == \partial_t \frac{\int T^{00} x^i x^j \, dV}{c}$$

Compare with eq.(5.40),p.179.

5) FN: “Combining equations (5.39) and (5.40) gives”

In[100]:=

```
eqn[5, 39] /. Rule @@ eqn[5, 40]
Print["Igittigitt! Beautify..."]
%% //. {Inactive[D][α_ / c, t] → Inactive[D][α, t] / c,
   Inactive[D][α_ / 2, t] → Inactive[D][α, t] / 2};
% /. Inactive[D][Inactive[D][α_, t], t] → Inactive[D][α, t, t];
% /. (1 / (2 c²)) α_ → HoldForm[1 / (2 c²)] α
Print["Compare with the equation immediately following eq.(5.40),p.179."]
```

Out[100]=

$$\int T^{ij} \, dV == \partial_t \frac{\partial_t \frac{\int T^{00} x^i x^j \, dV}{c}}{2 c}$$

Igittigitt! Beautify...

Out[104]=

$$\int T^{ij}\, dV == \frac{1}{2\,c^2}\,\partial_{t,t}\int T^{00}\, x^i\, x^j\, dV$$

Compare with the equation immediately following eq.(5.40),p.179.

6) FN: “For slowly moving source particles $T^{00} \approx \rho\, c^2$, where ρ is the proper density, and equation (5.36) yields the approximate expression”

In[106]:=

```
hbuu[i, j][c t, x⃗] == HoldForm[-2 G / (c^4 r0)] ×
    Inactivate[D[integral[V′]["source\nregion"][ρ [c t - r0, x⃗′] xu[i]′ xu[j]′], t, t],
     D] // FrameBox // DisplayForm
Print["for i,j=1,2,3 and with ", {xu[1], xu[2], xu[3]} == x⃗, " and ",
  {xu[1]′, xu[2]′, xu[3]′} == x⃗′]
Print["The gauge condition holds."]
Print["Compare with eq.(5.41),p.180."]
```

Out[106]//DisplayForm=

$$\bar{h}^{ij}[c\,t, \vec{x}] == -\frac{2\,G}{c^4\, r_0}\,\partial_{t,t}\left(\int_{\text{source region}} \rho[c\,t - r_0, \vec{x}'] \,(x^i)'\,(x^j)'\, dV'\right)$$

for i,j=1,2,3 and with $\{x^1, x^2, x^3\} == \vec{x}$ and $\{(x^1)', (x^2)', (x^3)'\} == \vec{x}'$

The gauge condition holds.

Compare with eq.(5.41),p.180.

■ 2) Gravitational radiation from a binary system

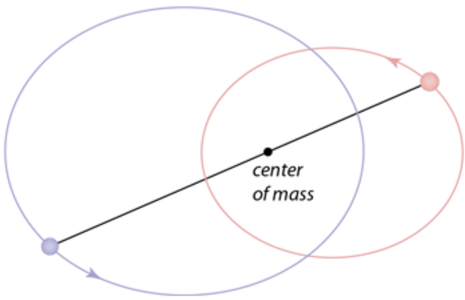
Thomas Moore: “The most *common* cosmic sources of gravitational waves are binary systems.”

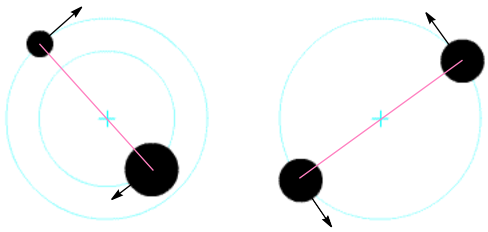
2.1) Introduction

Let us consider here the case of two bodies of mass *m1* and *m2* bound by gravitation and moving on closed orbits (seen by an observer at a distance r_0). **Newton’s theory of gravitation** (Kepler problem), here the appropriate theory for source modelling, tells us that the two bodies (seen as point masses) in general will move periodically on two coplanar, aligned, similar (same eccentricity) ellipses with a common focus and so that the line joining their centers of gravity always passes through the overall center of gravity of the system (coincident with the common focus). (In any case, the theory can be extended to more general sources without too much difficulty.)

With suitable initial conditions, the orbits can be two concentric circles with orbital radii *r1* and *r2*, or even two coincident circles if the masses are identical. In the case of circular orbits, the distance *d* between the two bodies remains constant and their orbital angular frequency *ω* is also constant and identical.

The following figures summarize the situation (closed Kepler orbits):





In[110]:=

```
Print["Some usefull relations:"]
$Assumptions = {c > 0, G > 0, m1 > 0, m2 > 0, r0 > 0, r > 0, d > 0, {ω, t, ϕ, ψ} ∈ Reals}
r1r = r1 → m2/(m1 + m2) d
r2r = r2 → m1/(m1 + m2) d
d == r1 + r2 /. {r1r, r2r} // Simplify
dr = Solve[d^3 ω^2 == G (m1 + m2), d][[1, 1]] (* for circular orbits *)
```

Some usefull relations:

Out[111]=

$\{c > 0, G > 0, m1 > 0, m2 > 0, r_0 > 0, r > 0, d > 0, (\omega \mid t \mid \phi \mid \psi) \in \mathbb{R}\}$

Out[112]=

$r1 \to \frac{d\, m2}{m1 + m2}$

Out[113]=

$r2 \to \frac{d\, m1}{m1 + m2}$

Out[114]=

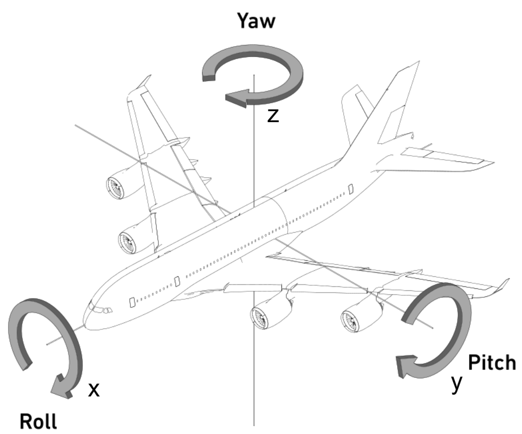
True

Out[115]=

$d \to \frac{G^{1/3}\,(m1 + m2)^{1/3}}{\omega^{2/3}}$

In[116]:=

(* A *basic rotation* R_{x_i} is a rotation about the x_i axis of a Cartesian coordinate system. A *general rotation* R_{zyx} is obtained applying the three basic rotations *roll* (R_x), *pitch* (R_y) and *yaw* (R_z) in this exact order. See *Wikipedia* sub voce "Rotation matrix". *)



;

```
              1      0          0
Rx[γ_] := ( 0  Cos[γ]  -Sin[γ] )     (* ROLL *)
              0  Sin[γ]   Cos[γ]

            Cos[β]   0  Sin[β]
Ry[β_] := (    0       1     0    )     (* PITCH *)
           -Sin[β]   0  Cos[β]

           Cos[α]  -Sin[α]  0
Rz[α_] := ( Sin[α]   Cos[α]   0 )     (* YAW *)
              0         0       1

Rzyx[α_, β_, γ_] := Rz[α].Ry[β].Rx[γ] (* GENERAL ROTATION *)
```

In[121]:=

```
Print["Positions of the two point masses at time t (elliptical orbit with generic
   parameters). The common orbital plane is rotated by a general rotation."]
Rzyx[α, β, γ];
({A[1], A[2], A[3]} = %.{-c1 + a1 Cos[ω t], b1 Sin[ω t], 0} /. c1 → √(a1² - b1²)) // MatrixForm
({B[1], B[2], B[3]} = %%.{+c2 - a2 Cos[ω t], -b2 Sin[ω t], 0} /. c2 → √(a2² - b2²)) // MatrixForm
```

```
Positions of the two point masses at time t (elliptical orbit with
  generic parameters). The common orbital plane is rotated by a general rotation.
```

Out[123]//MatrixForm=

```
  Cos[α] Cos[β] (-√(a1² - b1²) + a1 Cos[t ω]) + b1 (-Cos[γ] Sin[α] + Cos[α] Sin[β] Sin[γ]) Sin[t ω]
(  Cos[β] (-√(a1² - b1²) + a1 Cos[t ω]) Sin[α] + b1 (Cos[α] Cos[γ] + Sin[α] Sin[β] Sin[γ]) Sin[t ω]  )
                        -(-√(a1² - b1²) + a1 Cos[t ω]) Sin[β] + b1 Cos[β] Sin[γ] Sin[t ω]
```

Out[124]//MatrixForm=

```
  Cos[α] Cos[β] (√(a2² - b2²) - a2 Cos[t ω]) - b2 (-Cos[γ] Sin[α] + Cos[α] Sin[β] Sin[γ]) Sin[t ω]
(  Cos[β] (√(a2² - b2²) - a2 Cos[t ω]) Sin[α] - b2 (Cos[α] Cos[γ] + Sin[α] Sin[β] Sin[γ]) Sin[t ω]  )
                        -(√(a2² - b2²) - a2 Cos[t ω]) Sin[β] - b2 Cos[β] Sin[γ] Sin[t ω]
```

In[125]:=

```
Print["An example of elliptical orbits in a binary system A and
   B (similar to binary star system Alpha Centauri AB). Parameters:"]
{a1 → 10.57, b1 → a1 √(1 - e²), a2 → 12.83, b2 → a2 √(1 - e²), e → 0.5179,
 ω → 1, α → 30 Degree, β → 1.382, γ → 0}
Print["Position of A and B as a function of time:"]
{A[1], A[2], A[3]} //. %%
{B[1], B[2], B[3]} //. %%%
Show[
 ParametricPlot3D[{%%, %}, {t, 0, 2 π},
  PlotStyle → {Yellow, Orange}, PlotLegends → {"orbit of A", "orbit of B"}],
 Graphics3D[{Green, Opacity[.1], InfinitePlane[Table[%% /. t → i, {i, {0, 2, 4}}]]}],
 AxesLabel → {"x1", "x2", "x3"}, AxesOrigin → {0, 0, 0},
 ViewPoint → Top, AspectRatio → Automatic, ImageSize → Medium
]
```

```
An example of elliptical orbits in a binary system A
  and B (similar to binary star system Alpha Centauri AB). Parameters:
```

Out[126]=

```
{a1 → 10.57, b1 → a1 √(1 - e²), a2 → 12.83,
 b2 → a2 √(1 - e²), e → 0.5179, ω → 1, α → 30 °, β → 1.382, γ → 0}
```

```
Position of A and B as a function of time:
```

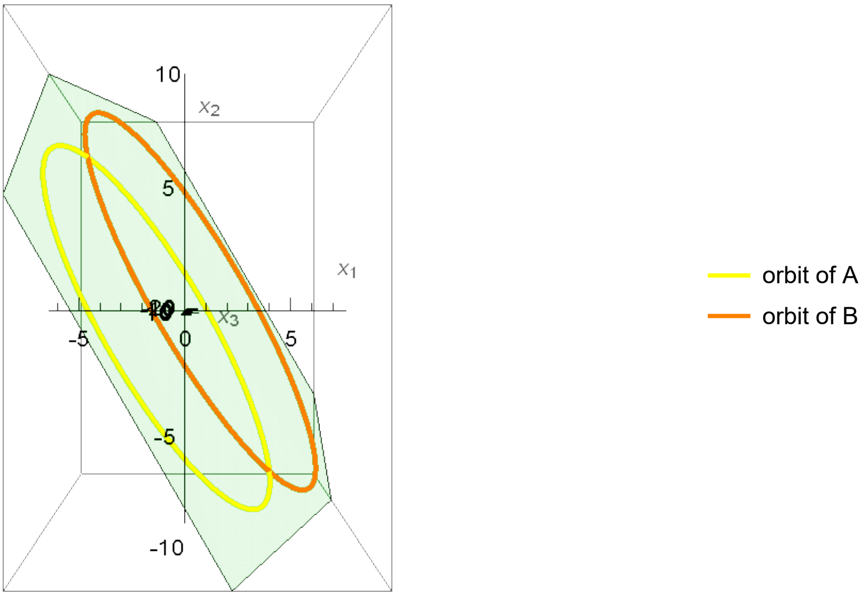
Out[128]=

```
{0.162533 (-5.4742 + 10.57 Cos[t]) - 4.52101 Sin[t],
 0.0938384 (-5.4742 + 10.57 Cos[t]) + 7.83061 Sin[t], 0. - 0.982231 (-5.4742 + 10.57 Cos[t])}
```

Out[129]=

```
{0.162533 (6.64466 - 12.83 Cos[t]) + 5.48766 Sin[t],
 0.0938384 (6.64466 - 12.83 Cos[t]) - 9.5049 Sin[t], 0. - 0.982231 (6.64466 - 12.83 Cos[t])}
```

Out[130]=



In[131]:=

```
Print["We evaluate now the integral in eq.(5.41), p.180, for elliptic motion. The
   two point masses are modeled with a mass density containing Dirac delta
   functions δ, something that looks like ρ = m1 δ(position₁)+m2 δ(position₂)."]
(m1 Table[A[i] × A[j], {i, 1, 3}, {j, 1, 3}] + m2 Table[B[i] × B[j], {i, 1, 3}, {j, 1, 3}]) //
  Simplify;
DisplayForm[integral[V′]["source\nregion"][ρ [c t - r0, x⃗′] xu[i]′ xu[j]′]] == MatrixForm[%]
Print[
 "Evaluating also the derivative in eq.(5.41) we get finally (in harmonic gauge):"]
hbarijMatrix = - (2 G)/(c^4 r0) D[%%%, t, t] /. t → t - r0 / c;
f = HoldForm[- (4 d^2 G m1 m2 ω^2)/(c^4 (m1 + m2) r0)];
hbuu[i, j][c t, x⃗] == f MatrixForm[%% / ReleaseHold[f]] // FrameBox // DisplayForm
Print["ELLIPTIC MOTION VERSION. r0 =
   approximated distance of the field point from the source = |x⃗|"]
```

```
We evaluate now the integral in eq.(5.41), p.180, for elliptic motion.
  The two point masses are modeled with a mass density containing Dirac delta
  functions δ, something that looks like ρ = m1 δ(position₁)+m2 δ(position₂).
```

Out[133]=

$$\int_{\text{source region}} \rho[c\,t - r_0, \vec{x}'] \,(x^i)'\,(x^j)'\, dV' == (\,m1\,(\text{Cos}[\alpha]\,\text{Cos}[\beta]\,(-\sqrt{a1^2 - b1^2} + a1\,\text{Cos}[t\,\omega]) + b1\,(-\text{Cos}[\gamma]\,\text{Sin}$$
$$m2\,((-\sqrt{a2^2 - b2^2} + a2\,\text{Cos}[t\,\omega])$$

Evaluating also the derivative in eq.(5.41) we get finally (in *harmonic gauge*):

Out[137]//DisplayForm=

$$\bar{h}^{ij}[c\,t, \vec{x}] == -\frac{4\,d^2\,G\,m1\,m2\,\omega^2}{c^4\,(m1 + m2)\,r_0}\;(\,\frac{(m1+m2)\,(2\,m1\,(b1\,\omega\,\text{Cos}[\omega\,(t-\frac{r_0}{c})]\,(-\text{Cos}[\gamma]\,\text{Sin}[\alpha]+\text{Cos}[\alpha]\,\text{Sin}[\beta]\,\text{Sin}[\gamma])-a1\,\omega\,\text{Cos}[\alpha]\,\text{Cos}[\beta]\,\text{Si}}{}$$

ELLIPTIC MOTION VERSION. r_0 = approximated distance of the field point from the source = $|\vec{x}|$

The formula for the general case (elliptic motion) is very long, but it does not present significant complexities; once entered the actual values for the parameters it becomes more manageable.

For circular motion in a coordinate plane we obtain a drastic simplification:

In[139]:=

```
Print["Bodies moving in circular orbits
   at a distance d in the x1 x2-plane (in harmonic gauge):"]
hbarijMatrix //. {a1 → r1, b1 → r1, a2 → r2, b2 → r2, r1r, r2r, γ → 0, β → 0, α → 0} // Simplify;
hbuu[i, j][c t, x⃗] == f MatrixForm[% / ReleaseHold[f]] // FrameBox // DisplayForm
Print["CIRCULAR MOTION VERSION. r0 =
   approximated distance of the field point from the source = |x⃗|"]
```

Bodies moving in circular orbits at a distance d in the $x_1\,x_2$-plane (in *harmonic gauge*):

Out[141]//DisplayForm=

$$\bar{h}^{ij}[c\,t, \vec{x}] == -\frac{4\,d^2\,G\,m1\,m2\,\omega^2}{c^4\,(m1 + m2)\,r_0} \begin{pmatrix} -\text{Cos}[2\,\omega\,(t - \frac{r_0}{c})] & -\text{Sin}[2\,\omega\,(t - \frac{r_0}{c})] & 0 \\ -\text{Sin}[2\,\omega\,(t - \frac{r_0}{c})] & \text{Cos}[2\,\omega\,(t - \frac{r_0}{c})] & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

CIRCULAR MOTION VERSION. r_0 = approximated distance of the field point from the source = $|\vec{x}|$

Once $\bar{h}^{ij}[c\,t, \vec{x}]$ has been transformed from harmonic gauge to its TT gauge in the appropriate direction, the **intensity and polarization analysis** can be carried out by determining the complex coefficients α and β of the specific *linear polarization matrices* e1 and e2; we have also previously shown that in TT gauge $\bar{h}^{ij} = h^{ij}$.

2.2) The TT gauge projection operator

To find the **polarization mode** of this plane-wave approximation at a field point in an arbitrary direction we have to transform $\bar{h}^{ij}[c\,t, \vec{x}]$ from the generic *harmonic gauge* to its *TT gauge* in the appropriate direction first. This can be done elegantly by a projection operator P as shown in C.W.Misner, K.S.Thorne, J.A.Wheeler, *Gravitation*, 1973, p.948-949. See the following excerpt for their recipe:

Box 35.1 METHODS TO CALCULATE "TRANSVERSE-TRACELESS PART" OF A WAVE

Problem: Let a gravitational wave $h_{\mu\nu}(t, x^j)$ in an arbitrary gauge of linearized theory be known. How can one calculate the metric perturbation $h^{TT}_{\mu\nu}(t, x^j)$ for this wave in the transverse-traceless gauge?

Solution 1 (valid only for waves; i.e., when $\Box\bar{h}_{\mu\nu} = 0$). Calculate the components R_{j0k0} of **Riemann** in the initial gauge; then integrate equation (35.10)

$$h^{TT}_{jk,00} = -2R_{j0k0} \qquad (1)$$

to obtain h^{TT}_{jk}. When the wave is monochromatic, $h_{\mu\nu} = h_{\mu\nu}(x^l)e^{-i\omega t}$; then the solution of (1) has the simple form

$$h^{TT}_{jk} = 2\omega^{-2}R_{j0k0}. \qquad (2)$$

Solution 2 (valid only for plane waves). "Project out" the *TT* components in an algebraic manner using the operator

$$P_{jk} = \delta_{jk} - n_j n_k. \qquad (3)$$

Here

$$n_k = k_k/|\boldsymbol{k}|$$

is the unit vector in the direction of propagation. Verify that P_{jk} is a projection operator onto the transverse plane:

$$P_{jl}P_{lk} = P_{jk}, \qquad P_{jk}n_k = 0, \qquad P_{kk} = 2.$$

Then the transverse part of h_{jk} is $P_{jl}h_{lm}P_{mk}$ (or in matrix notation, PhP); and the *TT* part is this quantity diminished by its trace:

$$h^{TT}_{jk} = P_{jl}P_{mk}h_{lm} - \frac{1}{2}P_{jk}(P_{ml}h_{lm}) \qquad (4)$$

(index notation),

$$h^{TT} = PhP - \frac{1}{2}P\,\mathrm{Tr}\,(Ph)$$

(matrix notation). (4′)

The sequence of operations that gives h^{TT}_{ij} cuts two parts out of h_{ij}. The first part cut out is

$$h^{T}_{jk} = \frac{1}{2}P_{jk}(P_{lm}h_{lm}), \qquad (5)$$

which is transverse but is built from its own trace,

$$h^T \equiv \mathrm{Tr}\,(PhP) = \mathrm{Tr}\,(Ph) = P_{lm}h_{ml}.$$

In[143]:=

```
(* Implementation of the MTW definitions (3) and (4') in matrix form: *)
P[n_] := Simplify[IdentityMatrix[3] - (Transpose[{n}].{n} / Norm[n]^2),
  Assumptions → {{n[[1]], n[[2]], n[[3]]} ∈ Reals}]
TTguageificator[n_, h_] := Simplify[P[n].h.P[n] - 1/2 P[n] × Tr[P[n].h]]
Print["Let's take a look at the projection operator P at
   work and verify the algebraic properties as suggested by MTW:"]
v = {a, b, c};
{Inactivate[P[v] , P] == (P[v] // MatrixForm),
 {P[v].P[v] == P[v], P[v] .v == {0, 0, 0}, Tr[P[v]] == 2} // Simplify}
```

```
Let's take a look at the projection operator P
  at work and verify the algebraic properties as suggested by MTW:
```

Out[147]=

$$\{P[\{a, b, c\}] == \begin{pmatrix} \frac{b^2+c^2}{a^2+b^2+c^2} & -\frac{a\,b}{a^2+b^2+c^2} & -\frac{a\,c}{a^2+b^2+c^2} \\ -\frac{a\,b}{a^2+b^2+c^2} & \frac{a^2+c^2}{a^2+b^2+c^2} & -\frac{b\,c}{a^2+b^2+c^2} \\ -\frac{a\,c}{a^2+b^2+c^2} & -\frac{b\,c}{a^2+b^2+c^2} & \frac{a^2+b^2}{a^2+b^2+c^2} \end{pmatrix}, \{True, True, True\}\}$$

2.3) Test

Of course, before proceeding we make some comparisons to verify that we are not completely off course.

In[148]:=

```
Print["- 1.Check -\nLes Houches summer school 2018. Thomas Moore, Session 5:
   Gravitational Waves\n(http://pages.pomona.edu/~tmoore/LesHouches/)."]
hbarijMatrix //. {a1 → r1, b1 → r1, a2 → r2, b2 → r2, r1r, r2r, γ → 0, β → 0, α → 0} //
  Simplify;
{TTguageificator[{0, 0, 1}, %] // MatrixForm, TTguageificator[{1, 0, 0}, %] // MatrixForm}
Print["Compare with equations 5.81 and 5.83,
   p.14.\nWarning: Other sign convention for d'Alembertian and c=1!"]
```

```
- 1.Check -
Les Houches summer school 2018. Thomas Moore, Session 5: Gravitational Waves
(http://pages.pomona.edu/~tmoore/LesHouches/).
```

Out[150]=

$$\left\{\begin{pmatrix} \frac{4 d^2 G\, m1\, m2\, \omega^2 \operatorname{Cos}[2\omega(t-\frac{r_0}{c})]}{c^4 (m1+m2) r_0} & \frac{4 d^2 G\, m1\, m2\, \omega^2 \operatorname{Sin}[2\omega(t-\frac{r_0}{c})]}{c^4 (m1+m2) r_0} & 0 \\ \frac{4 d^2 G\, m1\, m2\, \omega^2 \operatorname{Sin}[2\omega(t-\frac{r_0}{c})]}{c^4 (m1+m2) r_0} & -\frac{4 d^2 G\, m1\, m2\, \omega^2 \operatorname{Cos}[2\omega(t-\frac{r_0}{c})]}{c^4 (m1+m2) r_0} & 0 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 \\ 0 & -\frac{2 d^2 G\, m1\, m2\, \omega^2 \operatorname{Cos}[2\omega(t-\frac{r_0}{c})]}{c^4 (m1+m2) r_0} & 0 \\ 0 & 0 & \frac{2 d^2 G\, m1\, m2\, \omega^2 \operatorname{Cos}[2\omega(t-\frac{r_0}{c})]}{c^4 (m1+m2) r_0} \end{pmatrix}\right\}$$

```
Compare with equations 5.81 and 5.83, p.14.
Warning: Other sign convention for d'Alembertian and c=1!
```

In[152]:=

```
ar = {a1 → r1, b1 → r1, a2 → r2, b2 → r2,
   r1r, r2r, γ → 0, β → 0, α → 0, m1 → M, m2 → M, d → 2 a, r0 → r};
Print["- 2.Check -\na) With the substitutions ", ar,
 ", we get exactly the result of the rotating dumbbell example in FN."]
Print["Positions of particles A and B of the dumbbell at time t: "]
{{A[1], A[2], A[3]}, {B[1], B[2], B[3]}} //. ar
hh = hbarijMatrix //. ar // Simplify;
f = HoldForm[(8 a^2 G M ω^2)/(c^4 r)];
hbuu[i, j][c t, x⃗] == f MatrixForm[hh / ReleaseHold[f]] // FrameBox // DisplayForm
Print["See FN eq.(5.42),p.180."]
Print["Projection into x3-direction TT gauge leaves "h̄"^ij invariant (r ≈ x3):"]
TTguageificator[{0, 0, 1}, hh] // MatrixForm
% == hh
Print["FN,p.181: \"Note that this plane-wave approximation
   [observer on x3-axis] automatically has "h̄"^ij in its TT gauge.\""]
ar = {a1 → r1, b1 → r1, a2 → r2, b2 → r2, r1r, r2r, γ → 0, β → π / 2,
   α → 0, m1 → M, m2 → M, d → 2 a, r0 → r};
br = t → (t + π/(2 ω));
```

```
Print["b) With the substitutions ", ar, " and the phase shift ",
  br, ", we get exactly the solution to FN Exercice 5.3.1,p.182."]
hh = (hbarijMatrix //. ar /. br) // Simplify;
f = HoldForm[8 a^2 G M ω^2 / (c^4 r)];
Print["Positions of particles A and B of the dumbbell at time t: "]
{{A[1], A[2], A[3]}, {B[1], B[2], B[3]}} //. ar /. br // Simplify
hbuu[i, j][c t, x⃗] == f MatrixForm[hh / ReleaseHold[f]] // FrameBox // DisplayForm
Print["Projection into x3-direction TT gauge:"]
TTguageificator[{0, 0, 1}, hh] // MatrixForm
Print["Compare with solution to Exercice 5.3.1,p.274.
   The FN solution up to the multiplicative constant is:"]
{ Re[({{0, 0, 0}, {0, 1, -i}, {0, -i, -1}}) e^(2 i ω (t - r/c)) // ExpToTrig] // Simplify // MatrixForm,
 Re[(- (1 / 2) e1) e^(2 i ω (t - r/c)) // ExpToTrig] // Simplify // MatrixForm}
```

- 2.Check -

a) With the substitutions $\{a1 \to r1,\ b1 \to r1,\ a2 \to r2,\ b2 \to r2,$
$r1 \to \frac{d\,m2}{m1 + m2},\ r2 \to \frac{d\,m1}{m1 + m2},\ \gamma \to 0,\ \beta \to 0,\ \alpha \to 0,\ m1 \to M,\ m2 \to M,\ d \to 2\,a,\ r_0 \to r\}$
, we get exactly the result of the *rotating dumbbell example* in FN.

Positions of particles A and B of the dumbbell at time t:

Out[155]=

{{a Cos[t ω], a Sin[t ω], 0}, {-a Cos[t ω], -a Sin[t ω], 0}}

Out[158]//DisplayForm=

$$\bar{h}^{ij}[c\,t, \vec{x}] == \frac{8\,a^2\,G\,M\,\omega^2}{c^4\,r} \begin{pmatrix} \text{Cos}[2\,(-\frac{r}{c} + t)\,\omega] & \text{Sin}[2\,(-\frac{r}{c} + t)\,\omega] & 0 \\ \text{Sin}[2\,(-\frac{r}{c} + t)\,\omega] & -\text{Cos}[2\,(-\frac{r}{c} + t)\,\omega] & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

See FN eq.(5.42),p.180.

Projection into x_3-direction TT gauge leaves $\bar{h}^{ij}$ invariant ($r \approx x_3$):

Out[161]//MatrixForm=

$$\begin{pmatrix} \frac{8\,a^2\,G\,M\,\omega^2\,\text{Cos}[2\,(-\frac{r}{c}+t)\,\omega]}{c^4\,r} & \frac{8\,a^2\,G\,M\,\omega^2\,\text{Sin}[2\,(-\frac{r}{c}+t)\,\omega]}{c^4\,r} & 0 \\ \frac{8\,a^2\,G\,M\,\omega^2\,\text{Sin}[2\,(-\frac{r}{c}+t)\,\omega]}{c^4\,r} & -\frac{8\,a^2\,G\,M\,\omega^2\,\text{Cos}[2\,(-\frac{r}{c}+t)\,\omega]}{c^4\,r} & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Out[162]=

True

FN,p.181: "Note that this plane-wave approximation
  [observer on x_3-axis] automatically has $\bar{h}^{ij}$ in its TT gauge."

b) With the substitutions $\{a1 \to r1,\ b1 \to r1,\ a2 \to r2,\ b2 \to r2,\ r1 \to \frac{d\, m2}{m1+m2},$ $r2 \to \frac{d\, m1}{m1+m2},\ \gamma \to 0,\ \beta \to \frac{\pi}{2},\ \alpha \to 0,\ m1 \to M,\ m2 \to M,\ d \to 2\, a,\ r_0 \to r\}$ and the phase shift $t \to t + \frac{\pi}{2\,\omega}$, we get exactly the solution to FN Exercice 5.3.1,p.182.

Positions of particles A and B of the dumbbell at time t:

Out[170]=

$$\{\{0,\ a\, \text{Cos}[t\,\omega],\ a\, \text{Sin}[t\,\omega]\},\ \{0,\ -a\, \text{Cos}[t\,\omega],\ -a\, \text{Sin}[t\,\omega]\}\}$$

Out[171]//DisplayForm=

$$\bar{h}^{ij}[c\, t,\ \vec{x}] == \frac{8\, a^2\, G\, M\, \omega^2}{c^4\, r} \begin{pmatrix} 0 & 0 & 0 \\ 0 & \text{Cos}[\frac{2\, r\, \omega}{c} - 2\, t\, \omega] & -\text{Sin}[\frac{2\, r\, \omega}{c} - 2\, t\, \omega] \\ 0 & -\text{Sin}[\frac{2\, r\, \omega}{c} - 2\, t\, \omega] & -\text{Cos}[\frac{2\, r\, \omega}{c} - 2\, t\, \omega] \end{pmatrix}$$

Projection into x_3-direction TT gauge:

Out[173]//MatrixForm=

$$\begin{pmatrix} -\frac{4\, a^2\, G\, M\, \omega^2\, \text{Cos}[\frac{2\, r\, \omega}{c} - 2\, t\, \omega]}{c^4\, r} & 0 & 0 \\ 0 & \frac{4\, a^2\, G\, M\, \omega^2\, \text{Cos}[\frac{2\, r\, \omega}{c} - 2\, t\, \omega]}{c^4\, r} & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Compare with solution to Exercice 5.3.1,p.274. The FN solution up to the multiplicative constant is:

Out[175]=

$$\left\{\begin{pmatrix} 0 & 0 & 0 \\ 0 & \text{Cos}[\frac{2\, r\, \omega}{c} - 2\, t\, \omega] & -\text{Sin}[\frac{2\, r\, \omega}{c} - 2\, t\, \omega] \\ 0 & -\text{Sin}[\frac{2\, r\, \omega}{c} - 2\, t\, \omega] & -\text{Cos}[\frac{2\, r\, \omega}{c} - 2\, t\, \omega] \end{pmatrix},\ \begin{pmatrix} -\frac{1}{2}\, \text{Cos}[\frac{2\, r\, \omega}{c} - 2\, t\, \omega] & 0 & 0 \\ 0 & \frac{1}{2}\, \text{Cos}[\frac{2\, r\, \omega}{c} - 2\, t\, \omega] & 0 \\ 0 & 0 & 0 \end{pmatrix}\right\}$$

Conclusion: In the case of the circular motion the code seems to work fine, but for the case of the elliptic motion I have not found any references in the literature with which to compare my formulas, so there remains some uncertainty about their actual correctness. Therefore, use them with caution!

2.4) Example: Gravitational radiation from a binary system on circular orbits

We make here the intensity and polarization analysis in the case of a binary system on circular orbits in the $x_1\, x_2$-plane. For convenience, I choose to keep the view point fixed on the x_3 axis and to vary the orientation of the orbital plane instead. So, for example, an orbital plane pitch angle of ϕ is equivalent to an observation of the binary system in the $x_1\, x_2$-plane from an angle ϕ with respect to the x_3 axis. The following code implements the analysis in this particular case:

In[176]:=

```
ar = {a1 → r1, b1 → r1, a2 → r2, b2 → r2, r1r, r2r, γ → ψ, β → ϕ, α → 0};
```

```
hbTT = TTguageificator[{0, 0, 1}, hbarijMatrix //. ar] // FullSimplify;
f = HoldForm[(d^2 G m1 m2 ω^2)/(c^4 (m1 + m2) r0)];
HoldForm[Re[(α e1 + β e2) Exp[ⅈ 2 ω (t - r0/c)]]];
Print["h̄"^ij_"TT gauge in\nx3-direction", " ==\n== ", %%% // MatrixForm, "\n== ", %,
 " ,\nwith e1 = ", MatrixForm[e1], " and e2 = ", MatrixForm[e2], " and α,β ∈ ℂ."]
EE = ReleaseHold[%%] // ExpToTrig;
Print["Extracting constants on lhs and expanding rhs we get"]
heqn = (hbTT == Simplify[EE]);
Print[f MatrixForm[heqn[[1]] / ReleaseHold[f]], " ==\n== ", MatrixForm[heqn[[2]]]]
Coefficient[#, Cos[2 ω (t - r0/c)]] - ⅈ Coefficient[#, Sin[2 ω (t - r0/c)]] &;
αβr = {α → %@hbTT[[1, 1]], β → %@hbTT[[1, 2]]};
Print["By simple inspection we see that the (redundant) system is solved by ",
 %, ".\nCheck that the proposed solution is actually valid:"]
Simplify[heqn /. αβr]
```

$\bar{h}^{ij}_{\text{TT gauge in } x_3\text{-direction}}$ ==

$$== \begin{pmatrix} \frac{d^2 G\, m1\, m2\, \omega^2\, ((6 \cos[\phi]^2-(-3+\cos[2\phi])\cos[2\psi])\cos[2\omega(t-\frac{r_0}{c})]+4\sin[2\phi]\sin[\psi]\sin[2\omega(t-\frac{r_0}{c})])}{2 c^4 (m1+m2) r_0} & -\frac{2 d^2 G\, m1\, m2\, \omega^2 (\cos[2\omega(t-\frac{r_0}{c})]\sin \ldots}{c^{\cdot}} \\ -\frac{2 d^2 G\, m1\, m2\, \omega^2 (\cos[2\omega(t-\frac{r_0}{c})]\sin[\phi]\sin[2\psi]-2\cos[\phi]\cos[\psi]\sin[2\omega(t-\frac{r_0}{c})])}{c^4 (m1+m2) r_0} & \frac{d^2 G\, m1\, m2\, \omega^2\, ((-6\cos[\phi]^2+(-3+\cos[2\phi])\cos[2 \ldots}{2 c^{\cdot}} \\ 0 & \end{pmatrix}$$

$$== \text{Re}[(\alpha\, e1 + \beta\, e2) \text{Exp}[i\, 2\, \omega\, (t - \frac{r_0}{c})]] \ ,$$

with $e1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$ and $e2 = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$ and $\alpha,\beta \in \mathbb{C}$.

Extracting constants on lhs and expanding rhs we get

$$\frac{d^2 G\, m1\, m2\, \omega^2}{c^4 (m1+m2) r_0} \begin{pmatrix} \frac{1}{2}((6\cos[\phi]^2-(-3+\cos[2\phi])\cos[2\psi])\cos[2\omega(t-\frac{r_0}{c})]+4\sin[2\phi]\sin[\psi]\sin[2\omega(t-\frac{r_0}{c})]) \\ -2(\cos[2\omega(t-\frac{r_0}{c})]\sin[\phi]\sin[2\psi]-2\cos[\phi]\cos[\psi]\sin[2\omega(t-\frac{r_0}{c})]) \\ 0 \end{pmatrix}$$

$$== \begin{pmatrix} \cos[2\omega(t-\frac{r_0}{c})]\text{Re}[\alpha]-\text{Im}[\alpha]\sin[2\omega(t-\frac{r_0}{c})] & \cos[2\omega(t-\frac{r_0}{c})]\text{Re}[\beta]-\text{Im}[\beta]\sin[2\omega(t-\frac{r_0}{c})] & 0 \\ \cos[2\omega(t-\frac{r_0}{c})]\text{Re}[\beta]-\text{Im}[\beta]\sin[2\omega(t-\frac{r_0}{c})] & -\cos[2\omega(t-\frac{r_0}{c})]\text{Re}[\alpha]+\text{Im}[\alpha]\sin[2\omega(t-\frac{r_0}{c})] & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

By simple inspection we see that the (redundant) system is solved by

$$\{\alpha \to \frac{d^2 G\, m1\, m2\, \omega^2\, (6\cos[\phi]^2-(-3+\cos[2\phi])\cos[2\psi])}{2 c^4 (m1+m2) r_0} - \frac{2\, i\, d^2 G\, m1\, m2\, \omega^2 \sin[2\phi]\sin[\psi]}{c^4 (m1+m2) r_0},$$
$$\beta \to -\frac{4\, i\, d^2 G\, m1\, m2\, \omega^2 \cos[\phi]\cos[\psi]}{c^4 (m1+m2) r_0} - \frac{2 d^2 G\, m1\, m2\, \omega^2 \sin[\phi]\sin[2\psi]}{c^4 (m1+m2) r_0}\}.$$

Check that the proposed solution is actually valid:

Out[188]=

True

To analyze the formula $\bar{h}^{ij}_{\text{TT gauge in } x_3\text{-direction}}$ numerically, we must choose some suitable concrete parameters.

In[189]:=

```
(* free choices *)
{c → 1, G → 1, m1 → 3, m2 → 5, ω → 1, r0 → 3, γ → ψ, β → ϕ, α → 0};
(* dependent parameters *)
dr /. %;
```

```
{r1r, r2r} /. % /. %%;
Print["Example of a complete set of explicit
   and consistent parameters for the system under consideration:"]
params = Append[%%%%, {%%%, %%}] // Flatten
```

Example of a complete set of explicit and consistent parameters for the system under consideration:

Out[193]=

$$\{c \to 1,\ G \to 1,\ m1 \to 3,\ m2 \to 5,\ \omega \to 1,\ r_0 \to 3,\ \gamma \to \psi,\ \beta \to \phi,\ \alpha \to 0,\ d \to 2,\ r1 \to \frac{5}{4},\ r2 \to \frac{3}{4}\}$$

With this choice of parameters we obtain for α and β as function of the observation angle ϕ and ψ:

In[194]:=

```
"h̄"^ij_"TT gauge in\nx3-direction" == (hbTT /. params // MatrixForm)
{α, β};
% == (% /. αβr /. params)
Print["Example:"]
Grid[Prepend[
  Transpose[{ϕ, α, 1. α, β, 1. β} /. αβr /. ϕ → Table[n π / 8, {n, 0, 16}]] /. ψ → 0 /. params,
  {"observation angle ϕ, ψ=0", "α", "numerical", "β", "numerical"}], Frame → All]
Grid[Prepend[Transpose[{ψ, α, 1. α, β, 1. β} /. αβr /. ϕ → 0 /. ψ → Table[n π / 8, {n, 0, 16}]] /.
   params, {"observation angle ψ, ϕ=0", "α", "numerical", "β", "numerical"}], Frame → All]
```

Out[194]=

$$\bar{h}^{ij}_{\text{TT gauge in } x_3\text{-direction}} == \left(\begin{array}{c} \frac{5}{4}\left(\mathrm{Cos}[2(-3+t)]\left(6\,\mathrm{Cos}[\phi]^2-(-3+\mathrm{Cos}[2\phi])\,\mathrm{Cos}[2\psi]\right)+4\,\mathrm{Sin}[2(-3+t)]\,\mathrm{Sin}[2\phi]\right. \\ -5\left(-2\,\mathrm{Cos}[\phi]\,\mathrm{Cos}[\psi]\,\mathrm{Sin}[2(-3+t)]+\mathrm{Cos}[2(-3+t)]\,\mathrm{Sin}[\phi]\,\mathrm{Sin}[2\psi]\right) \\ 0 \end{array}\right.$$

Out[196]=

$$\{\alpha,\ \beta\} == \{\frac{5}{4}\left(6\,\mathrm{Cos}[\phi]^2-(-3+\mathrm{Cos}[2\phi])\,\mathrm{Cos}[2\psi]\right)-5\,i\,\mathrm{Sin}[2\phi]\,\mathrm{Sin}[\psi],$$
$$-10\,i\,\mathrm{Cos}[\phi]\,\mathrm{Cos}[\psi]-5\,\mathrm{Sin}[\phi]\,\mathrm{Sin}[2\psi]\}$$

Example:

Out[198]=

observation angle ϕ, ψ=0	α	numerical	β	numerical
0	10	10.	-10 i	0. - 10. i
$\frac{\pi}{8}$	$\frac{5}{4}\left(3 - \frac{1}{\sqrt{2}} + 6\,\mathrm{Cos}[\frac{\pi}{8}]^2\right)$	9.26777	$-10\,i\,\mathrm{Cos}[\frac{\pi}{8}]$	0. - 9.2388 i
$\frac{\pi}{4}$	$\frac{15}{2}$	7.5	$-5\,i\,\sqrt{2}$	0. - 7.07107 i
$\frac{3\pi}{8}$	$\frac{5}{4}\left(3 + \frac{1}{\sqrt{2}} + 6\,\mathrm{Sin}[\frac{\pi}{8}]^2\right)$	5.73223	$-10\,i\,\mathrm{Sin}[\frac{\pi}{8}]$	0. - 3.82683 i
$\frac{\pi}{2}$	5	5.	0	0.
$\frac{5\pi}{8}$	$\frac{5}{4}\left(3 + \frac{1}{\sqrt{2}} + 6\,\mathrm{Sin}[\frac{\pi}{8}]^2\right)$	5.73223	$10\,i\,\mathrm{Sin}[\frac{\pi}{8}]$	0. + 3.82683 i
$\frac{3\pi}{4}$	$\frac{15}{2}$	7.5	$5\,i\,\sqrt{2}$	0. + 7.07107 i
$\frac{7\pi}{8}$	$\frac{5}{4}\left(3 - \frac{1}{\sqrt{2}} + 6\,\mathrm{Cos}[\frac{\pi}{8}]^2\right)$	9.26777	$10\,i\,\mathrm{Cos}[\frac{\pi}{8}]$	0. + 9.2388 i
π	10	10.	10 i	0. + 10. i
$\frac{9\pi}{8}$	$\frac{5}{4}\left(3 - \frac{1}{\sqrt{2}} + 6\,\mathrm{Cos}[\frac{\pi}{8}]^2\right)$	9.26777	$10\,i\,\mathrm{Cos}[\frac{\pi}{8}]$	0. + 9.2388 i
$\frac{5\pi}{4}$	$\frac{15}{2}$	7.5	$5\,i\,\sqrt{2}$	0. + 7.07107 i
$\frac{11\pi}{8}$	$\frac{5}{4}\left(3 + \frac{1}{\sqrt{2}} + 6\,\mathrm{Sin}[\frac{\pi}{8}]^2\right)$	5.73223	$10\,i\,\mathrm{Sin}[\frac{\pi}{8}]$	0. + 3.82683 i
$\frac{3\pi}{2}$	5	5.	0	0.
$\frac{13\pi}{8}$	$\frac{5}{4}\left(3 + \frac{1}{\sqrt{2}} + 6\,\mathrm{Sin}[\frac{\pi}{8}]^2\right)$	5.73223	$-10\,i\,\mathrm{Sin}[\frac{\pi}{8}]$	0. - 3.82683 i
$\frac{7\pi}{4}$	$\frac{15}{2}$	7.5	$-5\,i\,\sqrt{2}$	0. - 7.07107 i
$\frac{15\pi}{8}$	$\frac{5}{4}\left(3 - \frac{1}{\sqrt{2}} + 6\,\mathrm{Cos}[\frac{\pi}{8}]^2\right)$	9.26777	$-10\,i\,\mathrm{Cos}[\frac{\pi}{8}]$	0. - 9.2388 i
2π	10	10.	-10 i	0. - 10. i

Out[199]=

observation angle ψ, ϕ=0	α	numerical	β	numerical
0	10	10.	-10 i	0. - 10. i
$\frac{\pi}{8}$	$\frac{5}{4}(6 + \sqrt{2})$	9.26777	$-10\,i\,\mathrm{Cos}[\frac{\pi}{8}]$	0. - 9.2388 i
$\frac{\pi}{4}$	$\frac{15}{2}$	7.5	$-5\,i\,\sqrt{2}$	0. - 7.07107 i
$\frac{3\pi}{8}$	$\frac{5}{4}(6 - \sqrt{2})$	5.73223	$-10\,i\,\mathrm{Sin}[\frac{\pi}{8}]$	0. - 3.82683 i
$\frac{\pi}{2}$	5	5.	0	0. + 0. i
$\frac{5\pi}{8}$	$\frac{5}{4}(6 - \sqrt{2})$	5.73223	$10\,i\,\mathrm{Sin}[\frac{\pi}{8}]$	0. + 3.82683 i
$\frac{3\pi}{4}$	$\frac{15}{2}$	7.5	$5\,i\,\sqrt{2}$	0. + 7.07107 i
$\frac{7\pi}{8}$	$\frac{5}{4}(6 + \sqrt{2})$	9.26777	$10\,i\,\mathrm{Cos}[\frac{\pi}{8}]$	0. + 9.2388 i
π	10	10.	10 i	0. + 10. i
$\frac{9\pi}{8}$	$\frac{5}{4}(6 + \sqrt{2})$	9.26777	$10\,i\,\mathrm{Cos}[\frac{\pi}{8}]$	0. + 9.2388 i
$\frac{5\pi}{4}$	$\frac{15}{2}$	7.5	$5\,i\,\sqrt{2}$	0. + 7.07107 i
$\frac{11\pi}{8}$	$\frac{5}{4}(6 - \sqrt{2})$	5.73223	$10\,i\,\mathrm{Sin}[\frac{\pi}{8}]$	0. + 3.82683 i
$\frac{3\pi}{2}$	5	5.	0	0. + 0. i
$\frac{13\pi}{8}$	$\frac{5}{4}(6 - \sqrt{2})$	5.73223	$-10\,i\,\mathrm{Sin}[\frac{\pi}{8}]$	0. - 3.82683 i
$\frac{7\pi}{4}$	$\frac{15}{2}$	7.5	$-5\,i\,\sqrt{2}$	0. - 7.07107 i
$\frac{15\pi}{8}$	$\frac{5}{4}(6 + \sqrt{2})$	9.26777	$-10\,i\,\mathrm{Cos}[\frac{\pi}{8}]$	0. - 9.2388 i
2π	10	10.	-10 i	0. - 10. i

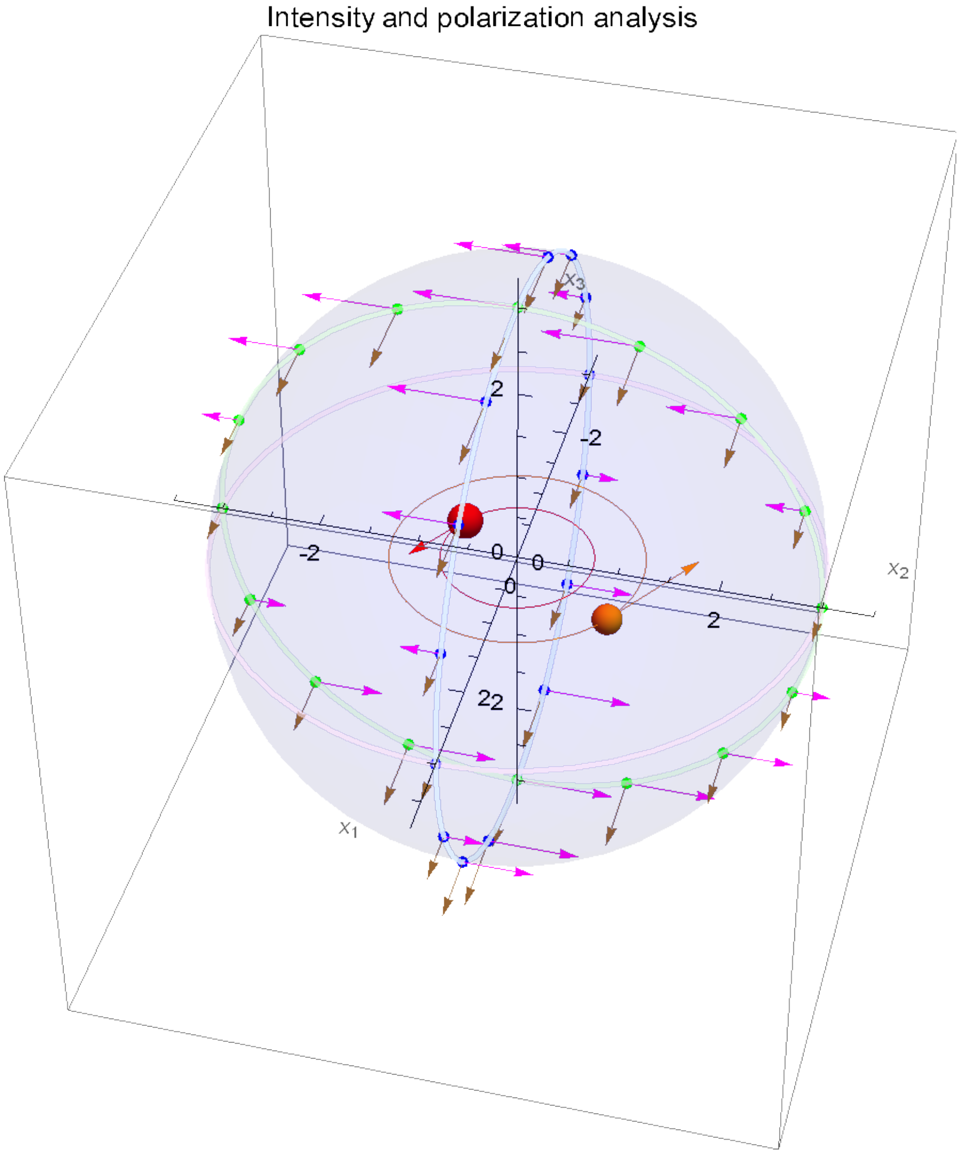
The following graph may shows the result of the intensity and polarization analysis more clearly.

In[200]:=

```
orbitA = {A[1], A[2], A[3]} //. ar /. params /. ϕ → 0 /. ψ → 0;
orbitB = {B[1], B[2], B[3]} //. ar /. params /. ϕ → 0 /. ψ → 0;
orbits = ParametricPlot3D[{orbitA, orbitB},
   {t, 0, 2 π}, PlotStyle → {{Thin, Orange}, {Thin, Red}}];
```

```
R[m_] := ∛(3 m / (4 π)) / 6
balls =
  Graphics3D[{Orange, Ball[orbitA, R[m1]], Red, Ball[orbitB, R[m2]]}] /. t → 1 /. params;
speedA = Graphics3D[{Arrowheads[Small], Orange, Arrow[{orbitA, orbitA + D[orbitA, t]}]}] /.
    t → 1;
speedB = Graphics3D[{Arrowheads[Small], Red, Arrow[{orbitB, orbitB + D[orbitB, t]}]}] /.
    t → 1;

obsball = Graphics3D[{Opacity[0.05], Blue, Ball[{0, 0, 0}, r0]}] /. params;
equator = ParametricPlot3D[r0 {Sin[u], Cos[u], 0} /. params,
    {u, 0, 2 π}, PlotStyle → {Opacity[0.6], LightMagenta}];

αβscale = 10;
(* Pitch orbital plane by ϕ *)obspoints1 = Graphics3D[{PointSize[Medium], Blue,
      Point[obpt = Table[r0 {Sin[n π / 8], 0, Cos[n π / 8]}, {n, 0, 15}]]}] /. params;
merid1 = ParametricPlot3D[r0 {Sin[u], 0, Cos[u]} /. params, {u, 0, 2 π},
    PlotStyle → {Opacity[0.9], LightBlue}];
αt = Table[{Re[α], Im[α], 0} /. αβr /. params /. ϕ → n π / 8 /. ψ → 0, {n, 0, 15}] / αβscale;
αArrows1 = Graphics3D[{Arrowheads[Small], Brown,
      Table[Arrow[{obpt[[n]], obpt[[n]] + αt[[n]]}], {n, 1, 16}]}] /. params;
βt = Table[{Re[β], Im[β], 0} /. αβr /. params /. ϕ → n π / 8 /. ψ → 0, {n, 0, 15}] / αβscale;
βArrows1 = Graphics3D[{Arrowheads[Small], Magenta,
      Table[Arrow[{obpt[[n]], obpt[[n]] + βt[[n]]}], {n, 1, 16}]}] /. params;
(* Roll orbital plane by ψ *)
obspoints2 = Graphics3D[{PointSize[Medium], Green,
      Point[obpt2 = Table[r0 {0, Sin[n π / 8], Cos[n π / 8]}, {n, 0, 15}]]}] /. params;
merid2 = ParametricPlot3D[r0 {0, Sin[u], Cos[u]} /. params, {u, 0, 2 π},
    PlotStyle → {Opacity[0.6], LightGreen}];
αt2 = Table[{Re[α], Im[α], 0} /. αβr /. params /. ϕ → 0 /. ψ → n π / 8, {n, 0, 15}] / αβscale;
αArrows2 = Graphics3D[{Arrowheads[Small], Brown,
      Table[Arrow[{obpt2[[n]], obpt2[[n]] + αt[[n]]}], {n, 1, 16}]}] /. params;
βt2 = Table[{Re[β], Im[β], 0} /. αβr /. params /. ϕ → 0 /. ψ → n π / 8, {n, 0, 15}] / αβscale;
βArrows2 = Graphics3D[{Arrowheads[Small], Magenta,
      Table[Arrow[{obpt2[[n]], obpt2[[n]] + βt[[n]]}], {n, 1, 16}]}] /. params;

Show[
 orbits, balls, speedA, speedB,
 obsball, equator,
 obspoints1, merid1, αArrows1, βArrows1,
 obspoints2, merid2, αArrows2, βArrows2,
 AxesOrigin → {0, 0, 0}, AxesLabel → {"x1", "x2", "x3"},
 PlotLabel → "Intensity and polarization analysis",
 PlotRange → All, ViewPoint → {3.5, .9, 3.1}]
```

Out[222]=



The two bodies m1 (orange) and m2 (red) of the binary system are shown together with their orbits and with their instantaneous velocity vector. The complex numbers α (brown) and β (magenta) are shown as arrows all referring to the same complex plane and are therefore directly comparable (see the previous tables for numerical values). Remember that in TT gauge $\bar{h}^{ij} = h^{ij}$.

To get an idea of the order of magnitude of h^{ij} that we can expect in astronomical reality, we insert now data into the final formula that roughly models the binary star system Alpha Centauri AB as a system on circular orbits. See also the following *2.5) Example: Alpha Centauri AB* for a more accurate model.

In[223]:=

```
Print["With the fixed distance for circular orbits ", dr, " we get:"]
hbTT /. {ϕ → 0, ψ → 0, dr};
f = HoldForm[(4 G^(5/3) m1 m2 ω^(2/3))/(c^4 (m1 + m2)^(1/3) r0)];
Subscript["h̄"^ij, "TT gauge in\nx3-direction"] == f (%% / ReleaseHold[f] // MatrixForm)
{c → Quantity["SpeedOfLight"], G → Quantity["GravitationalConstant"],
  m1 → Quantity[1.100, "SolarMass"], m2 → Quantity[0.907, "SolarMass"],
  ω → 2 π / Quantity[79.91, "Year"], r0 → Quantity[4.37, "LightYear"]};
Print["With the substitutions ", %, " the multiplicative constant becomes:"]
f == (ReleaseHold[f] //. %%);
% // FrameBox // DisplayForm
```

```
Print[
 "which determines the expected order of magnitude of the spacetime perturbation."]
{a → Quantity[1200., "KiloMeters"], b → Quantity[1064., "NanoMeters"], c → %%%[[2]]};
Print["In this case an interferometer with an effective arm length of ",
 a /. %, " operating with a laser of wavelength ", b /. %,
 " should reveal phase shifts equal to a fraction ", a c / b /. %,
 " of the wavelength used, which corresponds to a path variation
   due to the gravitational wave equal to ", UnitConvert[a c] /. %, "."]
```

With the fixed distance for circular orbits $d \to \frac{G^{1/3} (m1 + m2)^{1/3}}{\omega^{2/3}}$ we get:

Out[226]=

$$\bar{h}^{ij}_{\text{TT gauge in } x_3\text{-direction}} == \frac{4\, G^{5/3}\, m1\, m2\, \omega^{2/3}}{c^4 (m1 + m2)^{1/3}\, r_0} \begin{pmatrix} \mathrm{Cos}[2\,\omega\,(t - \frac{r_0}{c})] & \mathrm{Sin}[2\,\omega\,(t - \frac{r_0}{c})] & 0 \\ \mathrm{Sin}[2\,\omega\,(t - \frac{r_0}{c})] & -\mathrm{Cos}[2\,\omega\,(t - \frac{r_0}{c})] & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

With the substitutions

$\{c \to 1\, c,\ G \to 1\, G,\ m1 \to 1.1\, M_\odot,\ m2 \to 0.907\, M_\odot,\ \omega \to 0.0786283 \text{ per year},\ r_0 \to 4.37 \text{ ly}\}$

the multiplicative constant becomes:

Out[230]//DisplayForm=

$$\boxed{\frac{4\, G^{5/3}\, m1\, m2\, \omega^{2/3}}{c^4 (m1 + m2)^{1/3}\, r_0} == 6.01477 \times 10^{-23}}$$

which determines the *expected order of magnitude of the spacetime perturbation*.

In this case an interferometer with an effective arm length of 1200. km operating with a laser of wavelength 1064. nm should reveal phase shifts equal to a fraction 6.78358×10^{-11} of the wavelength used, which corresponds to a path variation due to the gravitational wave equal to 7.21773×10^{-17} m .

2.5) Example: Gravitational radiation from the binary star system Alpha Centauri AB

Alpha Centauri is the *closest star system* and closest planetary system to Earth’s Solar System. It is a triple star system, consisting of three stars: α Centauri A (Rigil Kentaurus), α Centauri B (Toliman) and α Centauri C (Proxima Centauri). α Centauri A and B are Sun-like stars (Class G and K) and together they form the **binary star system Alpha Centauri AB** which is weakly bound to the much more distant, small red dwarf α Centauri C that we can ignore here. (The exoplanets Proxima Centauri b, Earth-sized and in the habitable zone, and Proxima Centauri c orbit α Centauri C.)

For more information see for example Wikipedia sub voce “Alpha Centauri” or evaluate the commands
Entity[“Star”,”ProximaCentauri”][“Dataset”],
Entity[“Star”,”RigelKentaurusA”][“Dataset”],
Entity[“Star”,”RigelKentaurusB”][“Dataset”].

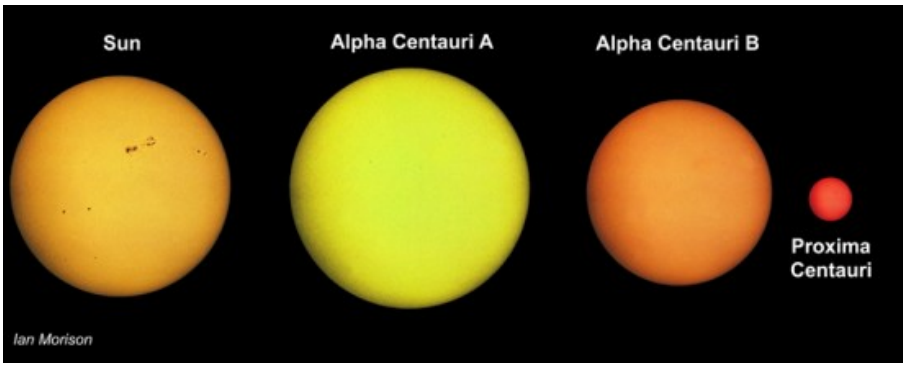
Warning: There is no exact concordance between the astronomical data of Mathematica and those of other sources.

In[234]:=

```
EntityList[EntityClass["Star", "AlphaCentauri"]]
```

Out[234]=

{ Proxima Centauri , Rigel Kentaurus A , Rigel Kentaurus B }



In[235]:=

```
(* Data *)
cr = {c → Quantity["SpeedOfLight"], G → Quantity["GravitationalConstant"],
  m1 → Quantity[1.100, "SolarMass"], m2 → Quantity[0.907, "SolarMass"],
  rperiastron → Quantity[11.2, "AstronomicalUnit"],
  rapastron → Quantity[35.6, "AstronomicalUnit"],
  a → (rperiastron + rapastron) / 2, e → 0.5179, i → 79.205 Degree,
  ω → 2 π / Quantity[79.91, "Year"], r0 → Quantity[4.37, "LightYear"]}
```

$$ar = \left\{a1 \to a\frac{m2}{m1+m2},\ b1 \to a\frac{m2}{m1+m2}\sqrt{1-e^2},\ a2 \to a\frac{m1}{m1+m2},\ b2 \to a\frac{m1}{m1+m2}\sqrt{1-e^2}\right\} \text{ //. cr}$$

Out[235]=

$$\{c \to 1\,c,\ G \to 1\,G,\ m1 \to 1.1\,M_\odot,\ m2 \to 0.907\,M_\odot,\ rperiastron \to 11.2\text{ au},\ rapastron \to 35.6\text{ au},\ a \to \frac{rapastron + rperiastron}{2},\ e \to 0.5179,\ i \to 1.38239,\ \omega \to 0.0786283\text{ per year},\ r_0 \to 4.37\text{ ly}\}$$

Out[236]=

```
{a1 → 10.5749 au, b1 → 9.04619 au, a2 → 12.8251 au, b2 → 10.9711 au}
```

Plot of the orbits. See *2.1) Introduction* for a 3D plot with the same parameters but including the inclination *i* (“pitch”).

In[237]:=

```
1.3385;
Most[{A[1], A[2], A[3]}] / % /. ar /. {ω → 1, γ → 0, β → 0, α → 0}
Most[{B[1], B[2], B[3]}] / %% /. ar /. {ω → 1, γ → 0, β → 0, α → 0}
Simplify[%% - %]
ParametricPlot[Evaluate[{%%%, %%, %} // QuantityMagnitude], {t, 0, 2 π}, PlotLegends →
  {"α Centauri A orbit", "α Centauri B orbit", "orbit of the relative motion"},
 PlotLabel → "axes units: 1 arcsec = 1.3385 au",
 PlotStyle → {Yellow, Orange, Green}, ImageSize → Medium]
```

Out[238]=

```
{0.747105 (-5.47673 au + Cos[t] (10.5749 au)), (6.75846 au) Sin[t]}
```

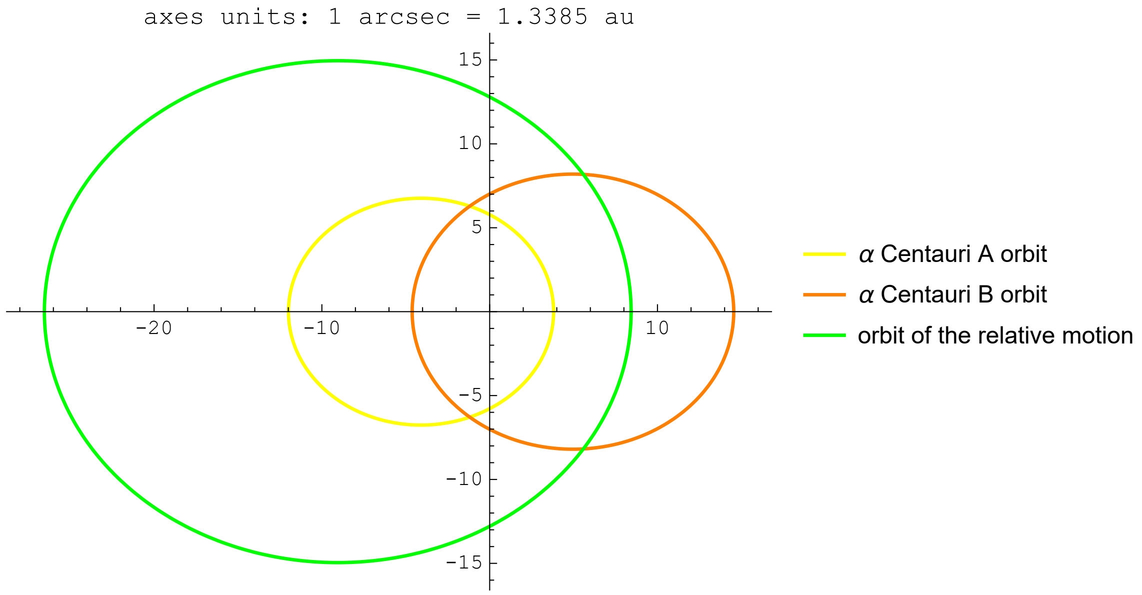
Out[239]=

```
{0.747105 (Cos[t] (-12.8251 au) + 6.64213 au), (-8.19658 au) Sin[t]}
```

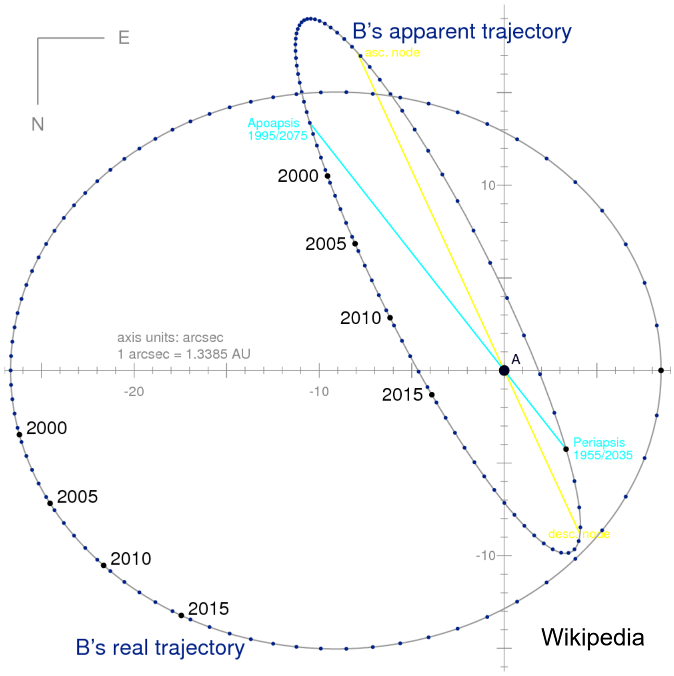
Out[240]=

```
{-9.05406 au + Cos[t] (17.4823 au), (14.955 au) Sin[t]}
```

Out[241]=



The following figure shows a plot of the positions of α Centauri B relative to α Centauri A deduced from experimental measurements. The orbit is nearly edge-on viewed from the Earth.



In our derivation of the far zone gravitational radiation field of a mass system we made a few assumptions: in fact it's a *"small-far-slow-weak" approximation* (see T. Moore, loc. cit.). We verify in the following that in the example under study these four conditions are actually satisfied, thus allowing us to use the formula deduced.

1. The source is *small* compared to the wave' s wavelength.

In[242]:=

```
rapastron /. cr
UnitConvert[c / (2 ω / 2 π) /. cr, "LightYears"]
%% / %
```

Out[242]=

35.6 au

Out[243]=

4.04552 ly

Out[244]=

0.000139148

2. The observer is so *far* away that he sees the source as tiny.

In[245]:=

```
rapastron /. cr
r0 /. cr
%% / %
```

Out[245]=

35.6 au

Out[246]=

4.37 ly

Out[247]=

0.000128816

3. The source is *slow* in that all parts of the source move with speeds v ≪ c.

In[248]:=

```
Print["a) Linear velocities of the stars wrt the observer: "]
Print["  Radial velocities (along line of sight): ",
 {#, UnitConvert[Entity["Star", #][{"RadialVelocity"}], "SpeedOfLight"]} & /@
  {"RigelKentaurusA", "RigelKentaurusB"}]

Print["  Transverse velocities in the plane of sky (orthogonal to line of sight):",
 {#, UnitConvert[Tan[√(Entity["Star", #1]["ProperMotionDeclination"]^2 +
          Entity["Star", #1]["ProperMotionRightAscension"]^2) * Quantity[1, "years"]] *
     r0 / Quantity[1, "years"] /. cr, "SpeedOfLight"]} & /@
  {"RigelKentaurusA", "RigelKentaurusB"}]

Print[
 "From the position as a function of time of the two stars we quickly obtain an upper
   limit of the norm of their velocities wrt the center of mass of the system:"]
sr = √a_ → Quantity[1, "AstronomicalUnit" "Years"^-1]
    √(a Quantity[1, "AstronomicalUnit"^-2 "Years"^2]) ;
Norm[D[Most[{A[1], A[2], A[3]}] /. t → Quantity[t, "Years"] /. ar /. cr /.
      {γ → 0, β → 0 , α → 0 }, Quantity[t, "Years"]]] /. sr // Simplify;
% /. {Cos[_] → 1, Sin[_] → 1};
Print["  α Centauri A orbital speed: ", %%, " < ", UnitConvert[%, "SpeedOfLight"]]
Norm[D[Most[{B[1], B[2], B[3]}] /. t → Quantity[t, "Years"] /. ar /. cr /.
      {γ → 0, β → 0 , α → 0 }, Quantity[t, "Years"]]] /. sr // Simplify;
% /. {Cos[_] → 1, Sin[_] → 1};
Print["  α Centauri B orbital speed: ", %%, " < ", UnitConvert[%, "SpeedOfLight"]]

Print["b) Rotational velocities at the equator: ",
 {{#, UnitConvert[2 π Entity["Star", #][{"Radius"}] / Entity["Star", #][{"RotationPeriod"}],
      "SpeedOfLight"]} &@"RigelKentaurusA",
  {#, UnitConvert[2 π Entity["Star", #][{"Radius"}] / Quantity[36, "Days"],
      "SpeedOfLight"]} &@"RigelKentaurusB"}]
```

```
Print["c) Furthermore, for Sun-like stars
   we don't expect internal mass flows at relativistic speeds."]
```

a) Linear velocities of the stars wrt the observer:

Radial velocities (along line of sight):
{{RigelKentaurusA, {-0.0000874 c}}, {RigelKentaurusB, {-0.0000604 c}}}

Transverse velocities in the plane of sky (orthogonal to line of sight):
{{RigelKentaurusA, 0.0000786471 c}, {RigelKentaurusB, 0.0000789545 c}}

From the position as a function of time of the two stars we quickly obtain an upper limit of the norm of their velocities wrt the center of mass of the system:

α Centauri A orbital speed: (1 au/yr)

$\sqrt{0.505929\,\mathrm{Cos}[0.0786283\,t]^2 + 0.691368\,\mathrm{Sin}[0.0786283\,t]^2} < 0.0000173141\,c$

α Centauri B orbital speed: (1 au/yr)

$\sqrt{0.744149\,\mathrm{Cos}[0.0786283\,t]^2 + 1.0169\,\mathrm{Sin}[0.0786283\,t]^2} < 0.0000209983\,c$

b) Rotational velocities at the equator:
{{RigelKentaurusA, {$9.\times10^{-6}$ c}}, {RigelKentaurusB, {3.9×10^{-6} c}}}

c) Furthermore, for Sun-like stars we don't expect internal mass flows at relativistic speeds.

4. The source is *weak* in that $h^{ij} \ll 1$ even very near the source.

In[261]:=

```
{#, UnitConvert[Entity["Star", #][{"Mass"}], "SolarMass"],
   UnitConvert[Entity["Star", #][{"Radius"}], "SolarRadius"]} & /@
 {"RigelKentaurusA", "RigelKentaurusB"}
Print["Similar to the Sun. We don't expect major deviations
   from flat spacetime, not even on the surface of the stars."]
```

Out[261]=

{{RigelKentaurusA, {1. $M_\odot$}, {1.0 $R_\odot$}}, {RigelKentaurusB, {0.77 $M_\odot$}, {0.82 $R_\odot$}}}

Similar to the Sun. We don't expect major deviations from flat spacetime, not even on the surface of the stars.

Conclusion: Everything seems to indicate that the *"small-far-slow-weak" approximation* is **applicable** to our case. So let's apply it!

In[263]:=

```
tmax = Ceiling[QuantityMagnitude[2 π / ω /. cr]];
{ar, cr, γ → 0, β → i , α → 0 } // Flatten;
Print["With the substitutions\n", %, "\nand after transformation in TT gauge we get"]
TTguageificator[{0, 0, 1}, hbarijMatrix //. %% /. t → Quantity[t, "years"] // Simplify];
"h̄^ij_"TT gauge in\nx3-direction"" == h^ij == (% // MatrixForm)
GraphicsGrid[Table[Plot[%%[[i, j]], {t, 0, tmax}, AxesOrigin → {0, 0}, ImageSize → Small,
   PlotLabel → h_ToString[i]<>ToString[j][t], ImagePadding → {{55, 10}, {0, 0}}],
  {i, 3}, {j, 3}], Frame → All, Background → LightCyan]
Print["Note: TT gauge in x3-direction = line of sight
   from α Cen AB to Earth, x-axis units: years."]
```

With the substitutions

{a1 → 10.5749 au , b1 → 9.04619 au , a2 → 12.8251 au , b2 → 10.9711 au , c → 1 c , G → 1 G , m1 → 1.1 $M_\odot$,

m2 → 0.907 $M_\odot$, rperiastron → 11.2 au , rapastron → 35.6 au , a → $\frac{\text{rapastron} + \text{rperiastron}}{2}$,

e → 0.5179, i → 1.38239, ω → 0.0786283 per year , r_0 → 4.37 ly , γ → 0, β → i, α → 0}

and after transformation in TT gauge we get

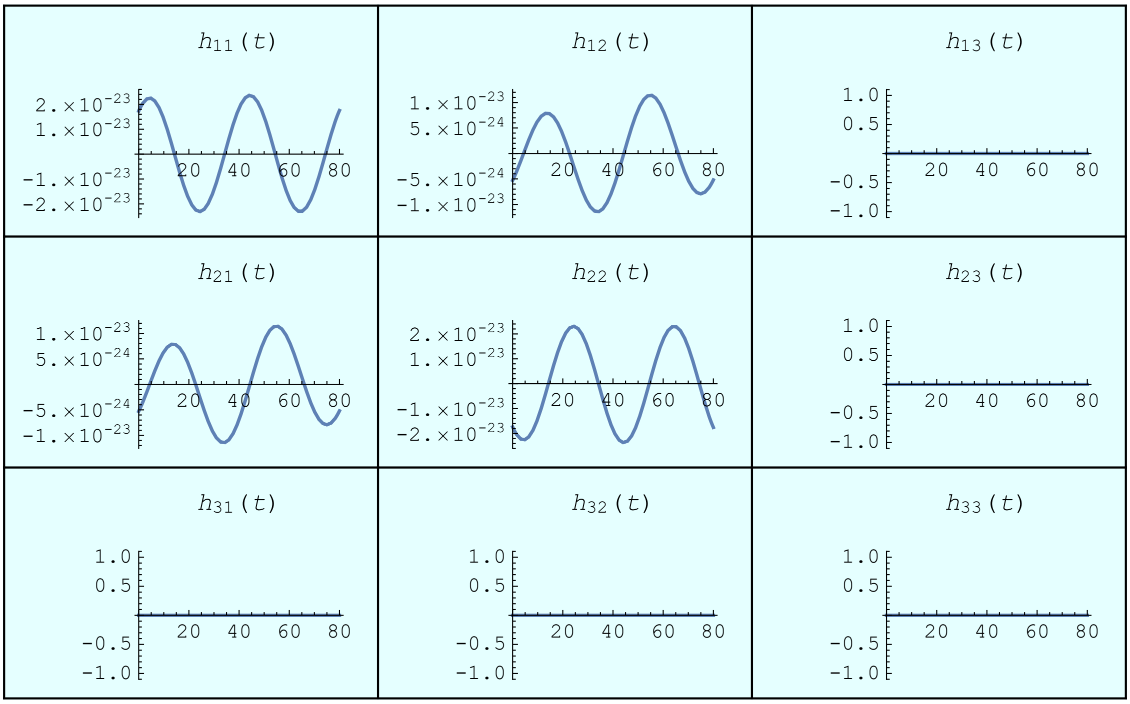
Out[267]=

$\bar{h}^{ij}_{\text{TT gauge in } x_3\text{-direction}} == h^{ij} == ($ -5.46803×10^{-25} Cos[0.343841 - 0.0786283 t] + 2.30805×10^{-23} Cos[0.343841 - 0

($-2.49743 \times 10^{-24} + 1.92889 \times 10^{-23}$ Cos[0.343841 - 0.0786

0

Out[268]=



Note: TT gauge in x_3-direction = line of sight from α Cen AB to Earth, x-axis units: years.

If I have done all the calculations right, this is finally the (weak and very slowly varying) *gravitational wave signal* expected from Alpha Centauri AB. The gravitational-wave luminosity of the system is 2.4×10^7 W (see below).

■ 3) Gravitational wave energy

FN: "This expression [energy loss rate by gravitational radiation] is obtained by looking at the energy-momentum carried by the gravitational field itself, which is quadratic in $h_{\mu\nu}$ and its derivatives, and consequently neglected in the linearized theory. As a consequence of the energy loss, [the angular speed] ω must decrease, but in the linearized theory it remains constant."

The energetic considerations in General Relativity are indeed decidedly complicated and involve the accurate analysis of already mentioned subtleties; a discussion of this topic can be found in the more advanced textbooks.

Here I will limit myself to mentioning two results. The one is the rate of loss of energy by gravitational radiation (*gravitational-wave luminosity*) from **gravitationally bound binary systems on closed Kepler orbits** such as those discussed above. The semi-major axis *a* and the eccentricity *e* are those of the orbit of relative motion. See L.D. Landau, E.M. Lifshitz; *Course of Theoretical Physics 2. The Classical Theory of Fields*, 6.ed. rus. 1973, 4.ed. eng. 1975, PROBLEMS after § 110. Radiation of gravitational waves. The original references are: P.C. Peters, J. Mathews; "Gravitational Radiation from Point Masses in a Keplerian Orbit", Phys. Rev. **131**, 435 (published 1 July 1963) and P.C. Peters, "Gravitational Radiation and the Motion of Two Point Masses", Phys. Rev. **136**, B1224 (published 23 November 1964). The other is the gravitational-wave luminosity of a **rotor** as given by FN, p.181.

In[270]:=

```
GravitationalWaveLuminosityKepler = (32 G^4 m1^2 m2^2 (m1 + m2))/(5 c^5 a^5) 1/(1 - e^2)^(7/2) (1 + 73/74 e^2 + 37/96 e^4);
```

$$\text{GravitationalWaveLuminosityKepler} = \frac{32\, G^4\, m1^2\, m2^2\, (m1 + m2)}{5\, c^5\, a^5}\, \frac{1}{(1 - e^2)^{7/2}}\, \left(1 + \frac{73}{74} e^2 + \frac{37}{96} e^4\right);$$

In[271]:=

```
Print["1) Compact circular binary system of Sun-like stars"]
{c → Quantity["SpeedOfLight"], G → Quantity["GravitationalConstant"],
 m1 → Quantity[1, "SolarMass"], m2 → Quantity[1, "SolarMass"],
 a → d /. dr, e → 0, ω → 2 π / Quantity[1, "Months"]}
GravitationalWaveLuminosityKepler /. {a → d /. dr, e → 0 };
Print["Gravitational wave luminosity:"]
%% == UnitSimplify[GravitationalWaveLuminosityKepler //. %%%]
Print["Comparison with the EM luminosity of the Sun:"]
%%[[2]] / (2 Entity["Star", "Sun"]["Luminosity"])
Print["Distance of the stars: ", UnitConvert[a //. %%%%%%, "au"]]
Print["See Exercise 5.3.2(b),p.182,solution p.274."]
```

1) Compact circular binary system of Sun-like stars

Out[272]=

$$\{c \to 1\, c,\ G \to 1\, G,\ m1 \to 1\, M_\odot,\ m2 \to 1\, M_\odot,\ a \to \frac{G^{1/3}\,(m1 + m2)^{1/3}}{\omega^{2/3}},\ e \to 0,\ \omega \to 2\pi \text{ per month}\}$$

Gravitational wave luminosity:

Out[275]=

$$\frac{32\, G^{7/3}\, m1^2\, m2^2\, \omega^{10/3}}{5\, c^5\, (m1 + m2)^{2/3}} == 5.43 \times 10^{16}\ \text{W}$$

Comparison with the EM luminosity of the Sun:

Out[277]=

7.10×10^{-11}

Distance of the stars: 0.2403 au

See Exercise 5.3.2(b),p.182,solution p.274.

In[280]:=

```
Print["2) Earth-Sun system"]
{c → Quantity["SpeedOfLight"], G → Quantity["GravitationalConstant"],
 m1 → Quantity[1, "SolarMass"], m2 → Quantity[1, "EarthMass"],
 a → Plus @@ Entity["Planet", "Earth"][{"Perihelion", "Aphelion"}] / 2,
 e → Entity["Planet", "Earth"]["Eccentricity"],
 ω → 2 π / Entity["Planet", "Earth"]["OrbitPeriod"]}
Print["Gravitational wave luminosity:"]
UnitSimplify[GravitationalWaveLuminosityKepler //. %%]
Print["Comparison with the EM luminosity of the Sun:"]
Entity["Star", "Sun"]["Luminosity"]
%%% / %
```

2) Earth-Sun system

Out[281]=

$$\{c \to 1\, c,\ G \to 1\, G,\ m1 \to 1\, M_\odot,\ m2 \to 1\, M_\oplus,$$
$$a \to 1.49597887 \times 10^8\ \text{km},\ e \to 0.016710220,\ \omega \to 0.017202125 \text{ per day}\}$$

Gravitational wave luminosity:

Out[283]=

197. W

Comparison with the EM luminosity of the Sun:

Out[285]=

3.83×10^{26} W

Out[286]=

5.1×10^{-25}

In[287]:=

```
Print["3) Binary star system Alpha Centauri AB"]
Drop[cr, {9, 11}]
Print["Gravitational wave luminosity:"]
UnitSimplify[GravitationalWaveLuminosityKepler //. %%]
Print["Comparison with the combined EM luminosity of the two stars:"]
Plus @@
 Flatten[Entity["Star", #][{"Luminosity"}] & /@ {"RigelKentaurusA", "RigelKentaurusB"}]
%%% /
 %
```

3) Binary star system Alpha Centauri AB

Out[288]=

$\{c \to 1\, c,\ G \to 1\, G,\ m1 \to 1.1\, M_\odot,\ m2 \to 0.907\, M_\odot,\ rperiastron \to 11.2\, au,\ rapastron \to 35.6\, au,\ a \to \frac{rapastron + rperiastron}{2},\ e \to 0.5179\}$

Gravitational wave luminosity:

Out[290]=

2.38836×10^{7} W

Comparison with the combined EM luminosity of the two stars:

Out[292]=

$6. \times 10^{26}$ W

Out[293]=

4.30743×10^{-20}

In[294]:=

$$\text{GravitationalWaveLuminosityRotor} = \frac{32\, G\, I^2\, \omega^6}{5\, c^5};$$

In[295]:=

```
Print["4) Rotating steel bar as gravitational wave lab experiment"]
Print["Energy loss rate by gravitational radiation of a rotor: ",
 "dE"/"dt" == GravitationalWaveLuminosityRotor // FrameBox // DisplayForm]
Print["I is the moment of inertia about the rotation
   axis, ω is the angular speed. See eq.(5.44),p.181."]
I → m l^2 / 12 ;
Print["Moment of inertia for a thin bar with rotation axis orthogonal to length: ", %]
GravitationalWaveLuminosityRotor /. %%
{c → Quantity["SpeedOfLight"], G → Quantity["GravitationalConstant"],
```

```
 m → Quantity[200, "MetricTon"], l → Quantity[10, "Meters"],
 ω → Quantity[50, "Radians" / "Seconds"], ρ → Quantity[8000., "Kilograms" / "Meters"^3]}
%% / Quantity[1, "Radians"^6] /. % // UnitSimplify
Print["Velocity of the ends of the bar: ",
 UnitConvert[l / 2 ω / Quantity[1, "Radians"] /. %%, "Kilometers" / "Hours"]]
Print["Diameter of the (cylindric) bar: ",
 Solve[d > 0 && (m == π (d / 2)^2 l ρ) /. %%%, d][[1, 1, 2]]]
Print["(Decorated in a Victorian style, this would make an impressive scientific
   device in a steampunk tale.)\nSee Exercise 5.3.2(a),p.182,solution p.274."]
```

4) Rotating steel bar as gravitational wave lab experiment

Energy loss rate by gravitational radiation of a rotor: $\frac{dE}{dt} == \frac{32\, G\, I^2\, \omega^6}{5\, c^5}$

I is the moment of inertia about the rotation axis, *ω* is the angular speed. See eq.(5.44),p.181.

Moment of inertia for a thin bar with rotation axis orthogonal to length: $I \to \frac{l^2\, m}{12}$

Out[300]=

$$\frac{2\, G\, l^4\, m^2\, \omega^6}{45\, c^5}$$

Out[301]=

$\{c \to 1\, c,\ G \to 1\, G,\ m \to 200\, t,\ l \to 10\, m,\ \omega \to 50\, rad/s,\ \rho \to 8000.\, kg/m^3\}$

Out[302]=

7.656×10^{-30} W

Velocity of the ends of the bar: 900 km/h

Diameter of the (cylindric) bar: 1.78412 m

(Decorated in a Victorian style, this would
  make an impressive scientific device in a steampunk tale.)
See Exercise 5.3.2(a),p.182,solution p.274.

The Royal Swedish Academy of Sciences has decided to award the Nobel Prize in Physics 2017 with one half to **Rainer Weiss** (LIGO/VIRGO Collaboration) and the other half jointly to **Barry C. Barish** (LIGO/VIRGO Collaboration) and **Kip S. Thorne** (LIGO/VIRGO Collaboration) *"for decisive contributions to the LIGO detector and the observation of gravitational waves"*

Press release: The Nobel Prize in Physics 2017, www.nobelprize.org

Chapter 6: Elements of cosmology

"Big Bang Theory"

(a song by the Canadian rock band *Barenaked Ladies*)

* * *

Our whole universe was in a hot dense state,
Then nearly fourteen billion years ago expansion started. Wait...
The Earth began to cool,
The autotrophs began to drool,
Neanderthals developed tools,
We built a wall (we built the pyramids),
Math, science, history, unraveling the mystery,
That all started with the big bang! (Bang!)

"Since the dawn of man" is really not that long,
As every galaxy was formed in less time than it takes to sing this song.
A fraction of a second and the elements were made.
The bipeds stood up straight,
The dinosaurs all met their fate,
They tried to leap but they were late
And they all died (they froze their asses off)
The oceans and Pangea
See ya wouldn't wanna be ya
Set in motion by the same big bang!

It all started with the big BANG!

It's expanding ever outward but one day
It will pause and start to go the other way,
Collapsing ever inward, we won't be here, it wont be heard
Our best and brightest figure that it'll make an even bigger bang!

Australopithecus would really have been sick of us
Debating out while here they're catching deer (we're catching viruses)
Religion or astronomy, Descartes or Deuteronomy
It all started with the big bang!

Music and mythology, Einstein and astrology
It all started with the big bang!
It all started with the big bang!

* * *

More details on **general relativistic cosmology** can be found in the relevant literature.

Personally, I make a distinction between the two words “universe” and “cosmos”. I call the whole of everything perceptible through the senses with the word “universe”; I call the whole of each and every thing instead with the word “cosmos”. If we admit the existence of things that are not perceptible to the senses, then the two words do not exactly coincide with their meaning, otherwise they are synonymous.

> “This argument is to my mind quite a strong one. One can say in reply that many scientific theories seem to remain workable in practice, in spite of clashing with E.S.P. [Extra-Sensory Perception]; that in fact one can get along very nicely if one forgets about it. This is rather cold comfort, and one fears that thinking is just the kind of phenomenon where E.S.P. may be especially relevant.”
>
> Alan Mathison Turing. “Computing Machinery And Intelligence”,
> *Mind*, New Series, Oxford, Volume 59, Issue 236 (Oct., 1950), p.433-460.

Appendices

coming soon...

"Nur wer nicht sucht, ist vor Irrtum sicher."

Albert Einstein (1879-1955)