Exploring Math $\boxed{\substack{\Sigma \\ \textit{math}}}$ with Eigenmath

Elementary Differential Geometry of Surfaces

using CAS Eigenmath and CalcPlot3D

Dr. Wolfgang Lindner
dr.w.g.Lindner@gmail.com
Leichlingen, Germany
2023

Contents

Preface

This booklet would like to whet your appetite to immerse yourself into the world of classical Differential Geometry. Here we do not reproduce the mathematical theory, but want instead focus on typical examples and their computations with the help of the small CAS EIGENMATH. The mathematical concepts and operations in Differential Geometry are often clumsy and uncomfortable and accompanied by heavy calculations, so EIGENMATH as your computer algebraic companion should unburden you of calculations in this theme. We only cite some of the necessary underlying mathematical definitions and facts to be able to show the corresponding implementation of the concepts into the language of the computer algebra system EIGENMATH.

For the mathematical description of Differential Geometry parallel with this script, I recommend the treatments by GALLOWAY, HECKMANN, HERTRICH-JEROMIN, HITCHIN, SHIFRIN and WHEELER to be found on the internet. These free online texts presents proofs of proposition and facts, that are only cited here. Recommended books are e.g. BANCHOFF/LEVITT, PRESSLEY or THORPE, mentioned in the bibliography.

Looking back at my first contact with EIGENMATH I was very impressed by the ingenious implementation of the EINSTEIN tensor in the calculation of the SCHWARZSCHILD metric by George WEIGT in [42]. His script is an eye opener and a convincing powerful example of the amazing efficiency of EIGENMATH.

The collection of the EIGENMATH examples and exercise scripts in this booklet not only want to help the reader to dive into the classical Differential Geometry of surfaces, but also to become comfortable with the use of the CAS EIGENMATH in this field.
It should be mentioned that one do not need to load any software packets into EIGENMATH to do your calculations as it it necessary in most of the big CAS as Maple, Mathematica, Maxima, Reduce or Sage etc. All you need is immediately at you fingertips and bulid-in in round about 100 EIGENMATH functions - the rest is coded ad hoc with no pain. So EIGENMATH is a small but well designed and powerful CAS, that may be used to solve problems in mathematics and the natural and engineering sciences. In particular, the clever built-in functions `dot`, `contract`, `d`, `sum` allow compact, easy and condensed tensorial constructs of classical Differential Geometry.

For the inspection or running an EIGENMATH script of this booklet no installation is necessary, *everything runs directly online*: a click on a link like this one ● ▷ Click here in this text is enough to invoke EIGENMATH with the corresponding script being imported - and by a click on the RUN button the calculation is made, allowing further ad hoc inputs form the user. If you own a Mac or Linux PC, there is the option to install the app EIGENMATH free of charge and run the scripts by *mark–copy–paste* into the EIGENMATH window.

There is a solution booklet with cross-checked calculations using Maple Vr4 and the packet `diffgeo.m` by RECKZIEGEL [24] to be found at LINDNER [17].

The author studied Differential Geometry from his dear academic teacher Prof. Dr. Peter DOMBROWSKI at the university of Cologne, Germany, in the 70ies of the last century in a suite of wonderful precise, crystal clear and inspiring lectures. At that time, illustrations in the lectures were hardly possible, mostly given with chalk sketches on the blackboard or hand sketches in the script. Calculations in the lectures exercises were tedious. This has now all changed for the better, this booklet try to demonstrate this with EIGENMATH for the calculations and CALCPLOT3D for the drawings.

I dedicate this little book as a friendly reference to P. DOMBROWSKI. I also took some nice lectures by Prof. Dr. Helmut RECKZIEGEL, who later wrote a fine book on Differential Geometry with Maple.

Peter DOMBROWSKI

I want to thank George WEIGT for his friendly support with tips and hints while writing these notes.

Wolfgang Lindner
Leichlingen, Germany
January 2023

1 Surfaces in Three Dimensions

Differential Geometry make heavy use of *(partial) differentiating* functions. We start by quoting a definition what a *surface in space* is, cf. e.g. SHIFRIN [32, p.35]. Then we visit and explore some specimens and individuals in the zoo of surfaces and show how to express and inspect surfaces and their characteristics in EIGENMATH.

1.1 Definition - Surface
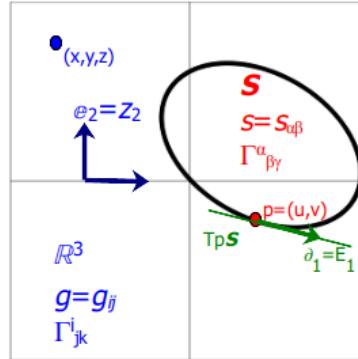
Let U be an open[1] set in $\mathbb{R}^2$. Let $\phi\colon U \to \mathbb{R}^3$ be a function on U.
We will use u, v as coordinates in our parameter region U, and x, y, z as coordinates in $\mathbb{R}^3$.
a. ϕ is called C^1 iff ϕ and its partial derivatives $\frac{\partial\phi}{\partial u}$ and $\frac{\partial\phi}{\partial v}$ exist and are continuous.
b. ϕ is called *smooth* iff ϕ is C^3. (We will assume all our functions ϕ are C^3.)
c. $\phi\colon U \to S \subset \mathbb{R}^3$ is called *parametrization of S* iff ϕ is C^3 and one-to-one and $\phi_u \times \phi_v \neq 0$, i.e. (ϕ_u, ϕ_v) are linear independent.
d. $S \subset \mathbb{R}^3$ is called a *surface* iff S is connected and each point has a neighborhood that is (regularly) parametrized.
• We will use subscripts a la $..._u$ to indicate partial derivatives, as follows:
$$f_u \equiv \frac{\partial f}{\partial u} \qquad f_v \equiv \frac{\partial f}{\partial v} \qquad f_{uu} \equiv \frac{\partial^2 f}{\partial u^2} \qquad f_{uv} \equiv \frac{\partial^2 f}{\partial u \partial v}$$
• As a first preview on what follows in the forthcoming chapters and to give an 'advanced organizer' we present in one scene most of the objects and concepts we will discuss later in detail:

Figure 1:
Basic perception/mental image of space vs surface:
$\mathbb{R}^3$: the ambient **space** □.
S: a **surface** ◯ inside the ambient space.
T_pS: tangential space at p on the surface S.
$e_2 = Z_2$ ..: basis vectors in ambient space, denoted $e_.$ or $Z_.$.
$\partial_1 = E_1$ ..: basis in tangential space T_pS, denoted $\partial_.$ or $E_.$.

---

[1]If in the following chapters U is not open, we could easily find an open set U' with $U \subset U'$.

●: a point with coordinates $(x, y, z) \in \mathbb{R}^3$ in ambient space.
●: a point with coordinates $(u, v) \in U$ on surface S.

Remark. We agree on the following conventions and notations in order to distinguish between objects in space respectively on the surface if necessary. In doubt we use latin letters like i, j, k (running through 1,2,3) to index mathematical entities in ambient space $V = \mathbb{R}^3$ and we use greek letters like α, β (running through 1,2) to address mathematical objects in a surface S. Per example, the covariant metric tensor in ambient space g (in short: the space metric g) is noted as g_{ij}, whereas the covariant metric tensor in surface (in short: the surface metric s) may be noted as $g_{\alpha\beta}$ or as $s_{\alpha\beta}$. If the context is clear and there is no doubt, we also use g_{ij} as notation for the surface metric.

LEXICON	*Math*	EIGENMATH
space coordinates $X = (x, y, z)$ or x_i	x_i	`X=(x,y,z)`
surface coordinates $U = (u, v)$ or u_α	u_α	`U=(u,v)`
space basis e_1, e_2, e_3	$e_i, i = 1, 2, 3$	`e1,e2,e3`
surface basis ∂_1, ∂_2 or E_1, E_2	∂_i or $E_i, i = 1, 2$	`E1,E2`
space metric $g = g_{ij}$	g_{ij}	`g=gij`
surface metric $s = s_{\alpha\beta}$	$s_{\alpha\beta}$	`s=salphabeta`

We start with some memories about surfaces of multivariate functions. We remember of graphs alias function surfaces, level sets and parameterizations. Function surfaces ('graphs') are preferred in advanced calculus, when it comes to partial derivatives and surface integrals, cf. [21] or [33, p.111, p.207 ff]. Parameterizations of surfaces are the main tool in Differential Geometry and we will focus later on that.

Remark. (A hint for the use of EIGENMATH)
If you see a link like
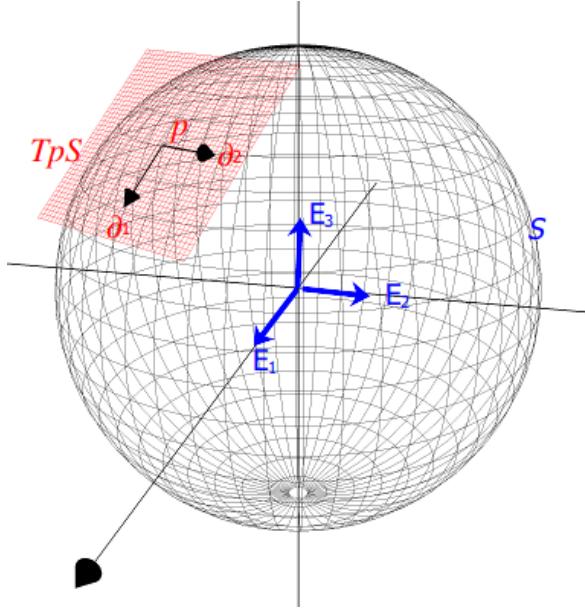
● Mark & Copy the blue code lines. ▷ Then Click here and Paste it into the input box ☐RUN. Press RUN.

do not forget to *click into the Online form* to give it the focus. You have the focus, if the EIGENMATH *Online frame* change to blue. Please check, wether all input lines are pasted with the right NEW LINE ending! Otherwise correct the pasting inside the online frame so that the code is formatted exactly as in the blue template here in this booklet.

If you see a link to open or run a script in CAS EIGENMATH like

● ▷ **Click** here to RUN the code.

you will directly be online linked to EIGENMATH to run, work or alter this code script.

Figure 2: Visualization of the tangential plane T_pS at p on surface S with tangential basis vectors $\partial_{\mathbf{1}}$ and $\partial_{\mathbf{2}}$ and ambient space basis E_1, E_2, E_3 in $\mathbb{R}^3$, cf. REJBRAND [25].

1.2 Function Surfaces

The graph of a function of two variables is a surface in space. The graph of function $f(x, y)$ consists of all points (x, y, z) in space such that (x, y) is in the definition domain of the function and $f(x, y) = z$, i.e.

$$graph(f) := \{(x, y, z) \in \mathbb{R}^3 | z = f(x, y)\}$$

Function surfaces alias function *graphs* are used primarily in multivariate calculus books, cf. DINEEN [7] or MARSDEN [21].

Example 1. We plot $graph(f)$ of the function $f\colon \mathbb{R}^2 \to \mathbb{R}$ defined by $f(x, y) = xy$ with CALCPLOT3D. Start CALCPLOT3D by clicking this link: $\triangleright$ Open CALCPLOT3D .

Then we only have to override the given function term $z = \boxed{7xy/exp(x^2 + y^2)}$ with $\boxed{xy}$ in the Graph menu:
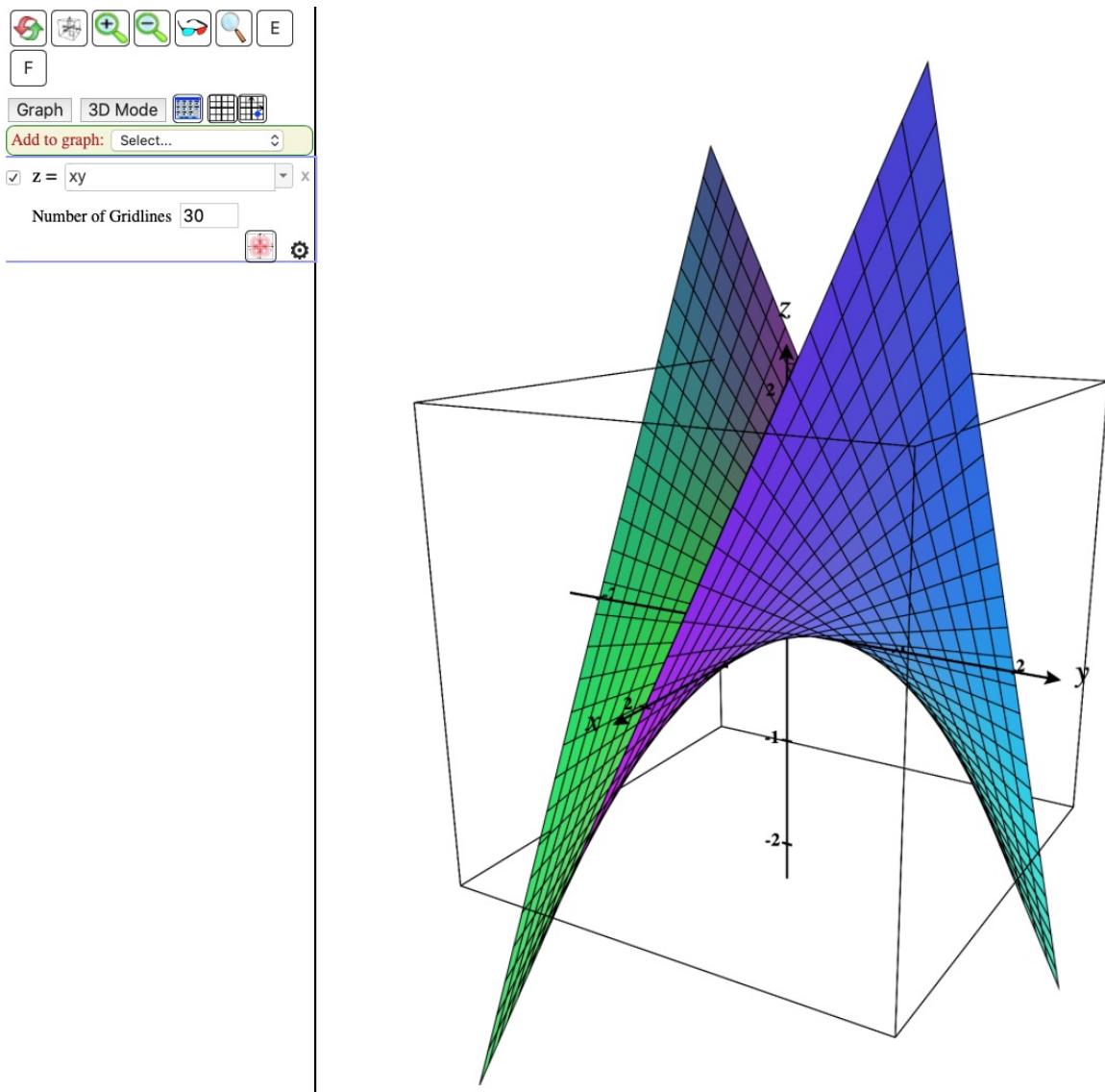
Figure 3:   surface plot of $graph(f)$ in $\mathbb{R}^3$ of function $f(x,y) = xy$ over rectangle $(x,y) \in [-2,+2] \times [-2,+2]$ in $\mathbb{R}^2$.
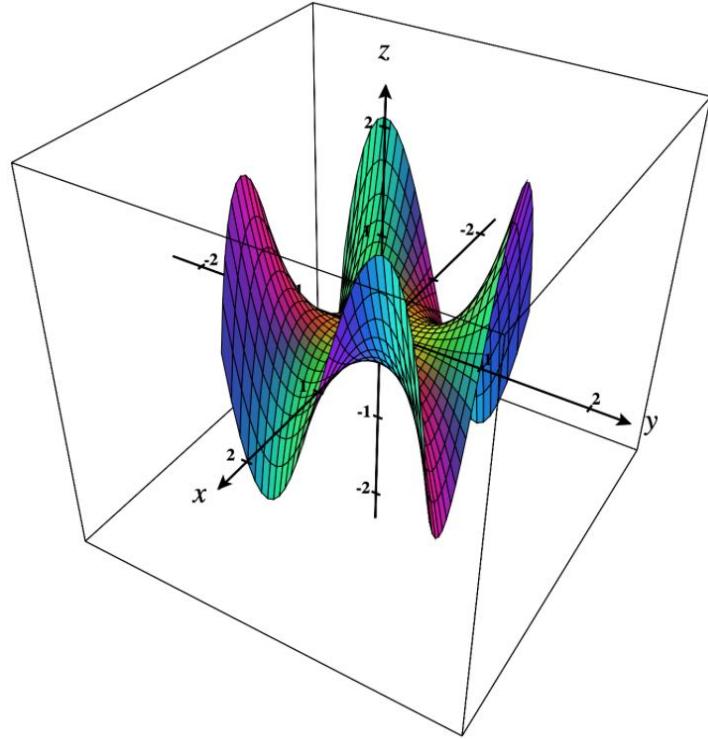
To use function f e.g. to calculate some points on the f-surface we invoke EIGENMATH.

```
# EIGENMATH
f(x,y) = x*y
f(3,2)
f(a,b)
"the point at the right peek"2D
f(2,2)
```

▷ **click** here and **copy & paste** the previous blue code lines into the input box ☐RUN. Then press RUN.

Exercise 1. Sketch the graph of $f(x, y) = x - y + 2$. For which coordinates x, y, z do you reach point $p = (-2, 0, 0)$? C.f. [21, p.711]

Exercise 2. Reproduce the plot of the so-called *dog saddle* i.e. the function surface of $f(x, y) = 4x^3y - 4xy^3$, c.f. [21, p.722].

Exercise 3. Plot the function surface of $f(x, y) = \frac{x^3 - 3x}{1 + y^2}$, c.f. [21, p.821].

Exercise 4. Plot the function surface of $f(x, y) = \frac{\sin(\pi x)}{1 + y^2}$, c.f. [21, p.822].

Exercise 5. Plot the function surface of $f(x, y) = \frac{x^2 - y^2}{x^2 + y^2}$, c.f. [21, p.837].

Exercise 6. Plot the function surface of $f(x, y) = \frac{y^3}{x^2 + y^2}$, c.f. [21, p.837].

Exercise 7. Plot the function surface of $f(x, y) = \frac{x^3 - 3xy^2}{x^2 + y^2}$, c.f. [21, p.837].

1.3 Level Surfaces

Every graph is a level surface, so level sets generalize the concept of a graph.

Let f be a function of three variables and let c be a constant real number.

A *c-level surface* of f, denoted $f^{-1}(c)$, is the set of all allowable inputs pairs (x, y) with $f(x, y) = c$:

$$f^{-1}(c) := \{(x, y, z) \in \mathbb{R}^3 : f(x, y, z) = c\}$$

Level surfaces alias level sets of functions are primarily used by THORPE [36] and are also addressed in some detail in DINEEN[7].

Example 2. We plot the 1-level surface $f^{-1}(1)$ of the function $f \colon \mathbb{R}^3 \to \mathbb{R}$ defined by $f(x, y, z) = x^2 + y^2 - z^2$, i.e. the set $\{(x, y, z) \in \mathbb{R}^3 : x^2 + y^2 - z^2 = 1\}$ with CALCPLOT3D. In the Add to graph: menu we select Implicit surface and write in the input box [Eq: x^2+y^2-z^2=1].
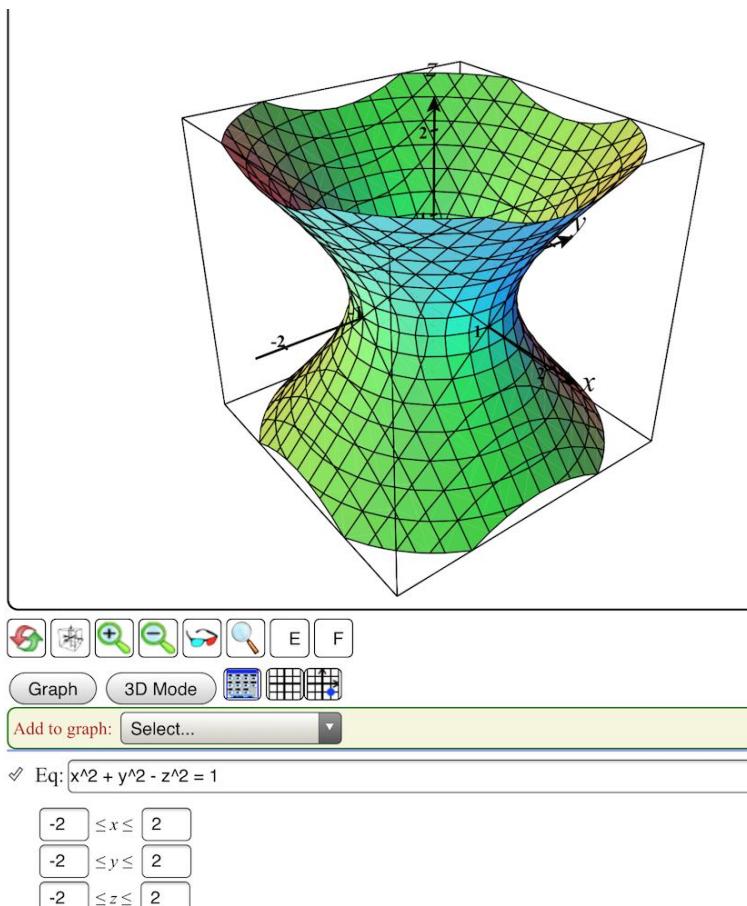
Figure 4: Level surface of $x^2 + y^2 - z^2 = 1$ over the cube $(x, y, z) \in [-2, +2] \times [-2, +2] \times [-2, +2]$.

We check some points in $\mathbb{R}^3$ for being members of the 1-level set of f with EIGENMATH:

```
# EIGENMATH
f(x,y,z) = x^2+y^2-z^2

f(1,1,1)                -- (1,1,1) is in level surface
f(sqrt(2),0,1)          -- (sqrt(2),0,1) is in level surface
f(2,1,2)                -- (2,1,2) is in level surface
```

▷ To open EIGENMATH **click** here.

Exercise 8. Let $f(x,x,z) = x^2 + y^2 - z^2$. Plot the level surface $f(x,y,z) = -1$.
Cf. [21, p.722].

Exercise 9. Let $f(x,y,z) = x^2 + 2y^2 + 3z^2 = 10$ be the ellipsoid realized as a level surface. Plot it.

Exercise 10. Let $f(x,y,z) = xe^y - z$. Plot the level surface $f^{-1}(0)$.

Exercise 11. Let $f(x,y,z) = \sqrt{x} + \sqrt{y} + \sqrt{z}$. Plot the level surface $f^{-1}(4)$.

1.4   Parametric Surfaces

Surfaces in 3-dimensional space are most described using a *parametric representation*. Each point p on a surface S is assigned a pair of parameter values u and v by a mapping ϕ, i.e. a parametrization ϕ maps a 2-dimensional region U of $\mathbb{R}^2$ to part of a surface in $\mathbb{R}^2$.

• For example, $\phi \colon [0, 2\pi] \times [o, \pi] \to \mathbb{R}^3$ defined through
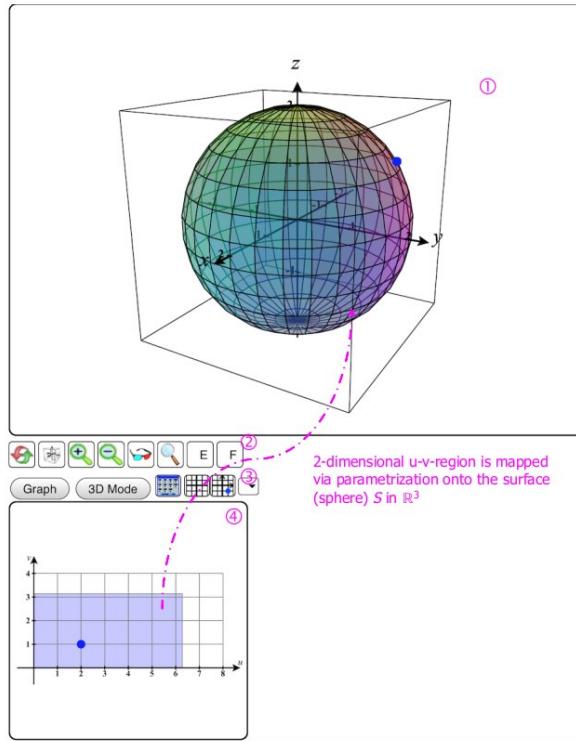
$$(u, v) \mapsto (2\sin(v)\cos(u), 2\sin(v)\sin(u), 2\cos(v))$$

patches the 2-dimensional sphere S^2. One also denote this mapping by

$$x(u, v) = 2\sin(v)\cos(u) \qquad y(u, v) = 2\sin(v)\sin(u) \qquad z(u, v) = 2\cos(v)$$

or shorter $x = 2\sin(v)\cos(u), y = 2\sin(v)\sin(u), z = 2\cos(v)$.
This surface is visualized by CALCPLOT3D in the following way: invoke the menu via
≡ Examples > Parametric Surfaces > Sphere > *and click the trace pane* t ⊞. The result is:

Parametric surface by ϕ with parameters u, v.
(1): the parametric surface $\phi([0, 2\pi] \times [o, \pi])$.
Figure 5: (2): some toggles for zoom, **E**dges, **F**aces etc.
(3): the trace pane to toggle (u, v) parameter region on/off.
(4): the parameter region $[0, 2\pi] \times [o, \pi]$ is 'on'.

We implement this parametrization ϕ using Eigenmath:

```
# EIGENMATH
Phi(u,v) = (2 sin(v) cos(u), 2 sin(v) sin(u), 2 cos(v))
Phi          -- show Phi
Phi(1,1)
float(Phi(1,1))
Phi(pi/3,pi/3)
Phi(a,b)
```

▷ Click here to RUN the code.
Eigenmath output:

$$\Phi = \begin{bmatrix} 2\cos(u)\sin(v) \\ 2\sin(u)\sin(v) \\ 2\cos(v) \end{bmatrix}$$

$$\begin{bmatrix} 2\cos(1)\sin(1) \\ 2\sin(1)^2 \\ 2\cos(1) \end{bmatrix}$$

$$\begin{bmatrix} 0.909297 \\ 1.41615 \\ 1.0806 \end{bmatrix}$$

$$\begin{bmatrix} \frac{1}{2}3^{1/2} \\ \frac{3}{2} \\ 1 \end{bmatrix}$$

In case you want to check, if u and v are admissible for input in the parametrization formula, you may test it in (1) before the calculation is done in (2) and otherwise through out an error message in (3):

```
# EIGENMATH
Phi(u,v) = test( and( and(0 <= u,u <= 2pi),             --(1)
                      and(0 <= v,v <= pi)),
          (2 sin(v) cos(u), 2 sin(v) sin(u), 2 cos(v)), --(2)
          "u or v not allowed." )                       --(3)

float(Phi(1,1))
Phi(-pi/3,pi/3)         -- output: 'u or v not allowed.'
```
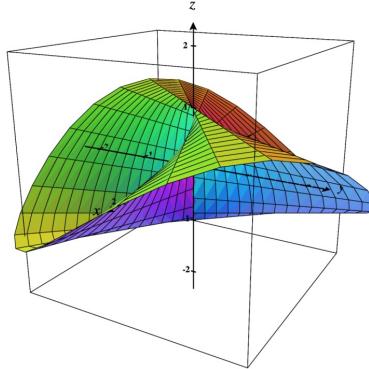
Remark. 1. Parametrization of surfaces is the most general concept to describe surfaces in space, cf. [7, p.100]. A parametrization is also called a surface *patch* or a *chart*.
More information is found e.g. in [23, p.59 ff], [36, p.121 ff] or [22, p.1 ff] or [32, p.34].
2. Because level surfaces and parametric surfaces are *locally* the same, cf. [36, p.121 ff], we use in the following mostly surface parametrizations.

3. THORPE [36, p.110] gives the definition: A *parametrized 2-surface* in $\mathbb{R}^3$ is a *smooth* map $\phi \colon U \to \mathbb{R}^3$, which is *regular*[2]. Herein U is a connected open set in $\mathbb{R}^2$.

Exercise 12. (PLÜCKER's Conoid)

a. Identify PLÜCKER's Conoid in the Parametric Surface list of CALCPLOT3D.

b. What is the parametric region for u, v ?

c. Give the definition of the parametrization ϕ.

d. Write down the parametrization ϕ in EIGENMATH and do some tests.

Exercise 13. (Catenoid) The *catenoid* is the 2D parametric surface defined by the following parametric equations:

$$\phi(u, v) = \begin{bmatrix} c \cosh \frac{v}{c} \cos u \\ c \cosh \frac{v}{c} \sin u \\ v \end{bmatrix}$$

where $u \in [-\pi, \pi)$ and $v \in \mathbb{R}$ and c is a non-zero real constant.
Plot the catenoid for $c = 1$ using CALCPLOT3D.

Exercise 14. (Helicoid) The *helicoid* is the 2D parametric surface defined by

$$\phi(u, v) = (v \cos(\alpha u), v \sin(\alpha u), u)$$

with patch region $u \in [-\pi, \pi)$, $v \in \mathbb{R}$ and α is a non-zero real constant.
Plot the helicoid surface for $\alpha = 1$ using CALCPLOT3D.

Exercise 15. (SCHERK's surface) The SCHERK surface is the 2D parametric surface defined by

$$\phi(u, v) = \log \left(\frac{\cos(u)}{\cos(v)} \right)$$

where $(u, v) \in \left(-\frac{\pi}{2}, +\frac{\pi}{2} \right) \times \left(-\frac{\pi}{2}, +\frac{\pi}{2} \right) \subset \mathbb{R}^2$.
Plot the SCHERK surface using CALCPLOT3D.

---

[2] i.e. the JACOBI matrix of all partial derivatives is non-singular, which allows to have locally non-zero normal vector fields, see below and §1.1. a.–d.

Exercise 16. (ENNEPER's surface) The ENNEPER surface is the 2D parametric surface defined by $\phi(u,v) = (u - \frac{1}{3}u^3 + uv^2, v - \frac{1}{3}v^2 + vu^2, u^2 - v^2)$ with patch region $u \in [-\pi, \pi)$, $v \in \mathbb{R}$. Plot the ENNEPER surface using CALCPLOT3D.

$\heartsuit$

Let's sum up:

LEXICON	*Math*	EIGENMATH
function surface for $f\colon U \subset \mathbb{R}^2 \to \mathbb{R}$:	$f(U) \subset \mathbb{R}^3$	f(x,y)
level surface for $f\colon U \subset \mathbb{R}^3 \to \mathbb{R}$:	$f^{-1}(c) \subset \mathbb{R}^3$	f(x,y,z)=c
parametric surface for $\phi\colon U \subset \mathbb{R}^2 \to \mathbb{R}^3$:	$\phi(U) \subset \mathbb{R}^3$	phi(u,v)

There is no commonly accepted notation for parametrization ϕ (=greek letter p for **p**arameter) and their partial derivatives $\phi_u \equiv \partial_u \phi \equiv \frac{\partial \phi}{\partial u}$, so we present this

LEXICON	*Math*	EIGENMATH
Parametrization ; partial derivative w.r.t. u:		
in BANCHOFF[2]:	$\vec{X}(u,v)$; $\vec{X}_u$	X(u,v) ; d(X,u)
in DINEEN[7]:	$\phi(u,v)$; ϕ_u	phi(u,v) ; d(phi,u)
in GRINFELD[11]:	S_α ; $\frac{\partial R}{\partial S_\alpha}$	S=(S1,S2) ; d(R,S1)
in HECKMANN [13], WHEELER [43]:	$\mathbf{r}(u,v)$; $\mathbf{r}_u$	r(u,v) ; d(r,u)
in PRESSLEY[23]:	$\sigma(u,v)$; σ_u	sigma(u,v) ; d(sigma,u)
in THORPE[36]:	$\varphi(u,v)$; φ_u	phi(u,v) ; d(phi,u)
in SHIFRIN[32]:	$x(u,v)$; x_u	x(u,v) ; d(x,u)

GRINFELD calls the parameter (u,v) the **S**urface coordinates, denoted $(S1, S2)$.
PRESSLEY denotes the parametrization σ after the greek lower case letter s for **S**urface.
HECKMANN and others denote the parametrization $\mathbf{r}$ after the position (di**r**ection) vector pointing to a point R on the surface ...
We use the convention in DINEEN.

$\heartsuit$

We should now have a solid feeling for what a 2D surface is.

$\heartsuit$

2   The GAUSS Map

We quote THORPE [36, p.31 ff], see also BANCHOFF [2, p.173] or SHIFRIN [32, p.45].
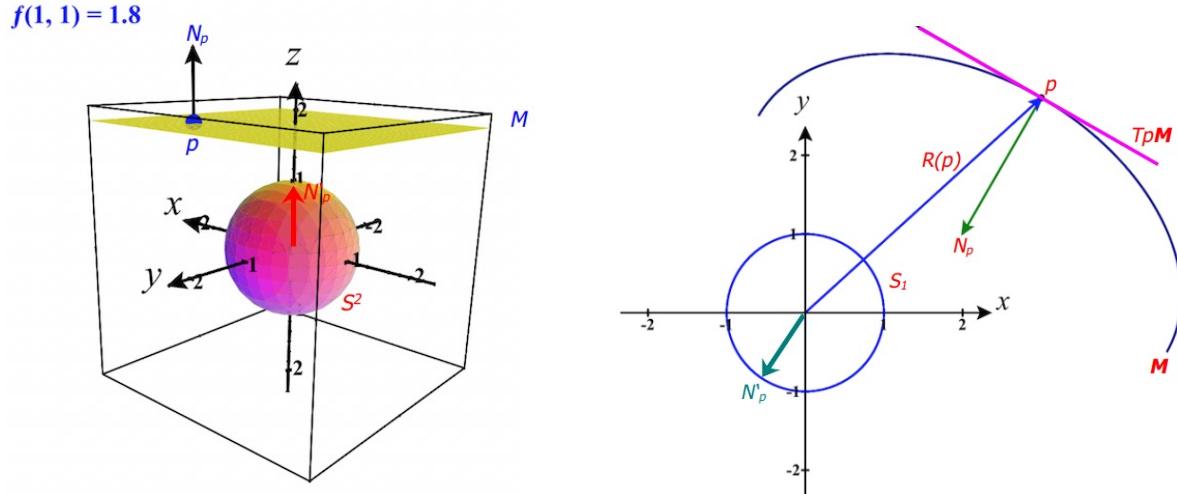
2.1   Definition

An *oriented* 2-surface S in $\mathbb{R}^3$ is a 2-surface together with a smooth unit normal vector field N on S. The function $N\colon S \to \mathbb{R}^3$ maps S into the unit sphere S^2. Think $N\colon S \to S^2$ as map which assigns to each point $p \in S$ the point in $\mathbb{R}^3$ obtained by translating the unit normal vector N_p to the origin.
This map N is called the GAUSS map.

Remark. This concept has two important consequences:

1. The Gauss map N allows for an *intuitive visual feeling* how (much) the surface is *'curved'*. Look at Fig. 6! We will comment about it in a moment.

2. The *exterior* normal vector N allows for *explicit calculation formulas* of the *curvature* of a surface - with EIGENMATH too. This will be a main theme of this booklet.

3. The normal N allows for the calculation and visualization of the tangent plane T_pS at a point p on the surface S. We will show how.


Figure 6:   Left: Image of GAUSS map in S^2 of a plane M. The blue normal vector N_p at p in plane M is mapped to the red vector N_p' in sphere S^2.
Right: Image of GAUSS map in S^1 of a curve (part of ellipse, a 1D surface) M. The normal vector N_p at p at tangential 'plane' T_pM is mapped to green vector N_p' in sphere S^1.

Comment. We interpret fig.6.

Left: The surface M is a plane in $\mathbb{R}^3$. At each point p of M the normal points in the same direction, i.e. the normal field is constant on the whole of M. Ergo the image of the GAUSS map remains the same constant normal N_p' on the sphere! This shows that the normal field does not change its value - the plane **is not 'curved'**. Using CALCPLOT3D we can move resp. trace the normal on the u-v-region and verify visually that N is constant and the coordinates of N, which are displayed at the green message pane above, remain the same.

Right: The 1D 'surface' M is part of an ellipse in $\mathbb{R}^2$. At the endpoint p of the position vector $R(p)$ the green exterior normal N_p is perpendicular to the tangent plane T_pM of the surface M. If you hike along M in direction of the letter 'y' the normal rests perpendicular to the changing T_pM's, but change its direction (and magnitude) always faster! Ergo N is not constant on M and its images N_p' builds a part of the sphere S^1. Ergo the surface M **is 'curved'**. Using CALCPLOT3D we can trace the normal N on the u-v-region and verify visually that N is permanently changing its direction and also its coordinates, which are displayed at the CALCPLOT3D's green message pane above. This explains en passant what '*field*' means ...

This should give you a first intuitive feeling what a *curvature* is.

LEXICON	*Math*	EIGENMATH
exterior surface normal vector field:	N	N
exterior surface normal vector at p:	N_p	N(p)

2.2 Examples of surface normals

In the following we give examples of normal vectors (fields) of surfaces S, using all three representations - normals on function surfaces, on level surfaces and on parametric surfaces. We also demonstrate how to calculate the corresponding tangential space T_pS using the normal vector at a specific point p.

2.2.1 Normal on a Function Surface

If we parameterize $S = graph(f)$ using the parameterization $\phi(u,v) = (u,v,f(u,v))$, then we can always choose

$$N := \frac{(-f_u, -f_v, 1)}{\sqrt{1 + f_u^2 + f_v^2}} \tag{2.1}$$

as unit normal (vector field) N. Remember: $f_u \equiv \frac{\partial f}{\partial u}$.

Example 3. Let $f \colon \mathbb{R}^2 \to \mathbb{R}$ defined by $f(x,y) = x^2 + y^2 - 4x - 6y + 13$, cf. [21, p.714].
a. Calculate the unit normal N at $p = (4,3)$.
b. Plot the function surface with unit normal N.

Solution: We use EIGENMATH. N is calculated by means of the differential operator
`d(f,x)`, which returns the partial derivative of f with respect to x.
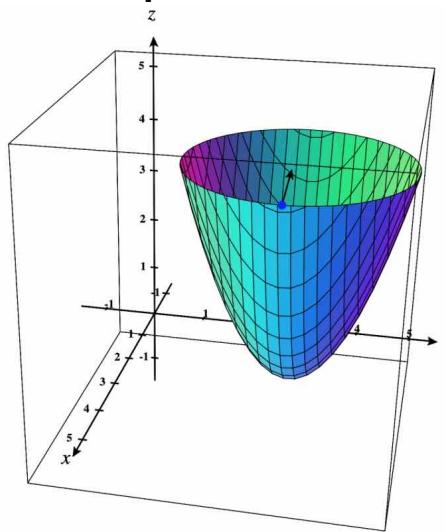
```
# EIGENMATH
f(u,v) = u^2+v^2-4u-6v+13
f
p = (4,3)
U = (u,v)      -- coordinates in input region U in R^2

fu = d(f,u)  -- partial derivative with respect to u
fv = d(f,v)

N = (-fu,-fv,1)/sqrt(1+fu^2+fv^2)        --(1) formula for N
Np = eval(N, u,4, v,3)                    --(2)  normal N at p
Np
```

● ▷ Click here to RUN the script.
EIGENMATH output: Np ≈ [-0.97,0,0.24].

Comment. In (1) we calculate the unit normal N using the given formula. With `eval(.)`
we put $u = 4$ and $v = 3$ in the return value for N. We get

$$N_p = (\frac{-4}{\sqrt{17}}, 0, \frac{1}{\sqrt{17}})$$

The plot of CALCPLOT3D shows a *paraboloid of revolution* with unit normal at $p = (4,3)$
pointing into the interior of the surface parallel to the $u = x$ axis with a tiny elevation
of -0.24 in the direction of the z-axis. The approx. value of N_p is also displayed in the
message pane of CALCPLOT3D as $\vec{N}(4,3) = -0.9701\hat{i} + 0.2425\hat{k}$.

Exercise 17. Verify the formula for N using the given parameterization and the general
formula for parametric surfaces in 2.2.3.

Exercise 18. (parabolic cylinder) Sketch the function surface given by $z = -y^2$, cf. [21, p.714]. Calculate and display the unit normal N at $p = (1, 2)$.
Trace N_p using u-v-region and watch the changing N.

Exercise 19. (volcano) Sketch the function surface given by

$$z = (x^2 + 3y^2) \cdot e^{1-x^2-y^2}$$

over the u-v-domain $[-2, +2] \times [-2, +2]$, cf. [21, p.717].
Calculate and display the unit normal N at $p = (1, 1)$.

2.2.2 Normal on a Level Surface

It is well known from calculus, cf. e.g. [36, p.8], that the gradient of a function f of three variables is orthogonal to the tangential plane T_pS at that point, where the gradient is calculated. Let p lie on the level surface S defined by $f(x, y, z) = c \in \mathbb{R}$, then $grad(f, p)$ is perpendicular ("normal") to T_pS, i.e. $grad(f, p) \perp T_pS$.
Therefore, if f is the defining function for a level surface $S = f^{-1}(c)$ we always choose the unit normed gradient[3] $grad(f)$ of f as the canonical unit normal vector field N:

$$N := \frac{grad(f)}{\|grad(f)\|} \tag{2.2}$$

Example 4. Let $f(x, y, z) = x^2 + y^2 - z^2$. Calculate $grad(f, p)$ for $p = (0, 0, 1)$.
Plot this gradient at the level surface $f(x, y, z) = -1$. c.f. [21, p.806].

Solution: We use EIGENMATH. $grad(f)$ is calculated by means of the differential operator `d(f,x)`, which returns the partial derivative of f with respect to x.

```
# EIGENMATH
f(x,y,z) = x^2 + y^2 - z^2
f
p = (0,0,1)
X = (x,y,z)   -- coordinates in ambient space

d(f,X)        -- (1)  d(f,X) = grad(f) i.e. all partial derivates

gradfp = eval(d(f,X), x,0, y,0, z,1)
gradfp        -- (2)
```
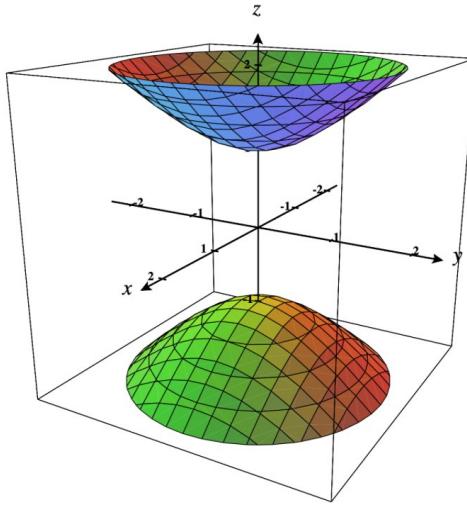
● ▷ Click here to RUN the script.
EIGENMATH output: [2x, 2y, -2z] and gradfp = (0,0,-2).

---

[3]We denote the gradient of f by $grad(f)$ and do not choose the common notation ∇f, because EIGENMATH does not have the nabla symbol ∇ available.

We plot the hyperboloid of two sheets using CALCPLOT3D.

Exercise 20. Calculate the unit normal N at p for the surface M in fig.6.left.

Exercise 21. Let $f(x, y, z) = x^2 + 2y^2 + 3z^2 = 10$ be the ellipsoid realized as a level surface.
a. Calculate a unit normal vector at each of the points $p_1 = (\sqrt{10}, 0, 0)$, $p_2 = (-\sqrt{10}, 0, 0)$ and $q_1 = (1, 0, \sqrt{3})$, $q_2 = (-1, 0, -\sqrt{3})$.
b. Do these normals point to the inside or outside of the ellipsoid?
c. Plot the complete scene with CALCPLOT3D and verify your findings, cf. [21, p.807].

Example 5. (Example 2.3 of [7, p.18]) Let S denote the set of all points in $\mathbb{R}^3$ which satisfy the equation $x^2 + 2y^2 - 5z^2 = 1$. Find the tangent space $T_p S$ and the normal at the point $(2, -1, 1)$ on S.

Solution: Our ansatz is:
1. Calculate the normal vector field N of S. In this case N must not be of unit length.
2. Calculate $T_p S$ using the fact that $N \perp T_p S$, i.e. $T_p S \colon (X - p) \bullet N = 0$ for all $X \in T_p M$.
We use EIGENMATH.

```
# EIGENMATH
f(x,y,z) = x^2+2y^2-5z^2
f
f(2,-1,1)      -- check if p in S, ok

p = (2,-1,1)
X = (x,y,z)  -- coordinates in ambient space R^3

d(f,X)         -- this is N=grad(f) normal vector field
```

```
gradfp = eval(d(f,X), x,2, y,-1, z,1)
gradfp                  -- this is Np = grad(f,p) the normal at p

TpS = dot(X-p,gradfp)  -- this must be zero, because N_|_TpS
TpS
```
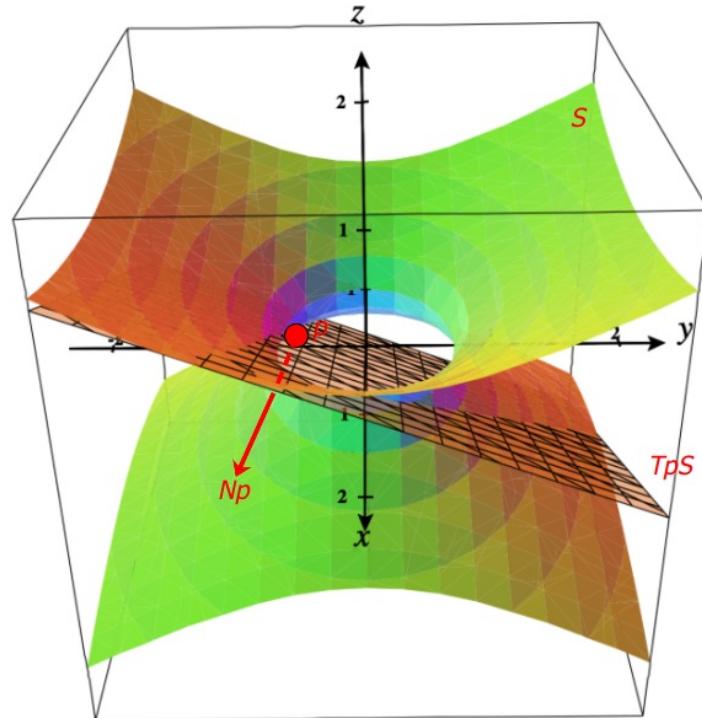
EIGENMATH output:

$$g_{radfp} = \begin{bmatrix} 4 \\ -4 \\ -10 \end{bmatrix}$$

$$T_{pS} = 4\,x - 4\,y - 10\,z - 2$$

● ▷ Click here to RUN the script.

Because $4x - 4y - 10z - 2 = 0$, we resume that $T_pS\colon 2x - 2y - 5z = 1$.
Let's plot the whole scene using CALCPLOT3D.

Remark. The normal N_p was added by hand ♡

2.2.3 Normal on a Parametric Surface
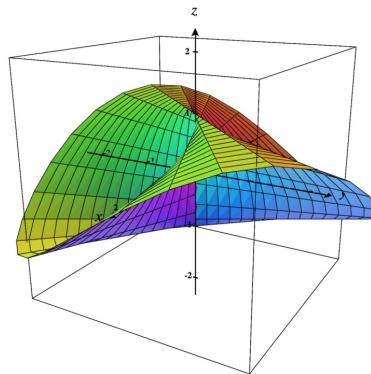
For a parameterized surface $\phi\colon U \to \mathbb{R}^3$ with $S = \phi(U)$ we always choose

$$N := \frac{\phi_u \times \phi_v}{\|\phi_u \times \phi_v\|} \tag{2.3}$$

as unit normal (vector field). Herein ϕ_u is the partial derivative of ϕ w.r.t. the first parameter u. It is well known that N is orthogonal to the tangential plane T_pS at the point, where N is calculated, i.e.

$$N_p \perp T_pS$$

Example 6. (PLÜCKER's Conoid)

PLÜCKER's Conoid is the parametrized surface defined by the parametric equations
$\phi(u,v) = (v \cdot \cos(u), v \cdot \sin(u), 2\cos(u) \cdot \sin(u))$, where $u \in [0, 2\pi]$ and $v \in [-3, 3]$.
Calculate the normal vector field $N_{pluecker}$.
Calculate the exterior surface unit normal N_p at $p = (\pi, 1)$.

Solution: We use EIGENMATH. N is calculated by means of the differential operator `d(phi,u)` which returns the partial derivative of the parameterization ϕ with respect to u.

```
# EIGENMATH
"PLUECKER conoid"
phi(u,v) = (v*cos(u), v*sin(u),2*cos(u)*sin(u))        -- parametrization
phi                 -- display phi
phi(pi,1)           -- the point p in S

phiu = d(phi,u)     -- partial derivative w.r.t. u
phiu

phiv = d(phi,v)
phiv

N = cross(phiu,phiv)   -- exterior normal vector
N
```

```
    N1 = N/abs(N)              -- exterior unit length normal
    N1

    Np = eval(N1, u,pi, v,1)   -- normal at point p
    Np
    float(Np)
```

EIGENMATH output:

$$
N_p = \begin{bmatrix} 0 \\ -\dfrac{2}{5^{1/2}} \\ -\dfrac{1}{5^{1/2}} \end{bmatrix}
$$

$$
N = \begin{bmatrix} -2\cos(u)^2\sin(u) + 2\sin(u)^3 \\ 2\cos(u)^3 - 2\cos(u)\sin(u)^2 \\ -v\cos(u)^2 - v\sin(u)^2 \end{bmatrix}
$$

$$
\begin{bmatrix} 0 \\ -0.894427 \\ -0.447214 \end{bmatrix}
$$

● ▷ Click here to RUN the script.

Comment. On the LHS we look at the normal vector field $N(u, v)$. On the RHS we see above the normal vector $N_p =$ N(pi,1) $\in \mathbb{R}^3$ and below its approximate value.

Example 7. (Cylinder) The *cylinder* is the 2D parametric surface defined by

$$
\phi(u, v) = (a\cos(v), a\sin(v), u)
$$

with patch region $v \in [-\pi, \pi)$, $u \in \mathbb{R}$ and a a non-zero real constant.
a. Reproduce the plot of the cylinder in WOLFRAM [46] for $a = 1$ using CALCPLOT3D.
Compare the parameterizations used by WOLFRAM with that of CALCPLOT3D at
≡ Examples > Parametric Surfaces > Cylinder >.
b. Calculate the normal vector field N for the cylinder.
c. Calculate the exterior surface normal N_p at $p = (1, \pi)$.

Solution **to b. and c.**   We use EIGENMATH.

```
    # EIGENMATH
    "Cylinder"
    phi(u,v) = (a cos(v), a sin(v), u)
    a = 1
    phi                       -- display phi
```

```
phi(1,pi)                    -- look at p=(pi,1) in S

phiu = d(phi,u)        -- partial derivative w.r.t. u
phiu

phiv = d(phi,v)
phiv

N = cross(phiu,phiv) -- exterior normal vector
N

absN = abs(N)
absN = 1                     -- simplify via c^2+s^2=1

absN
N1 = N/absN              -- exterior unit length normal
N1

Np = eval(N1, u,1, v,pi) -- normal at point p Np
float(Np)
```

EIGENMATH output of N_1 and $N_1(p)$:

$$N_1 = \begin{bmatrix} -\cos(v) \\ -\sin(v) \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

● ▷ Click here to RUN the script.

2.3   Exercises

Exercise 22. (Cone) The *cone* is the 2D parametric surface defined by

$$\phi(u, v) = (u \cos(v), u \sin(v), u)$$

with patch region $u \in [-\pi, \pi)$, $v \in \mathbb{R}$.
a. Reproduce the plot of the cone in WOLFRAM [45] using CALCPLOT3D.
Compare the parametrization used by WOLFRAM with that of CALCPLOT3D at
≡ Examples > Parametric Surfaces > Cone >.
b. Calculate the normal vector field N for the cone.
c. Calculate the exterior surface normal N_p at $p = (1, \pi)$.
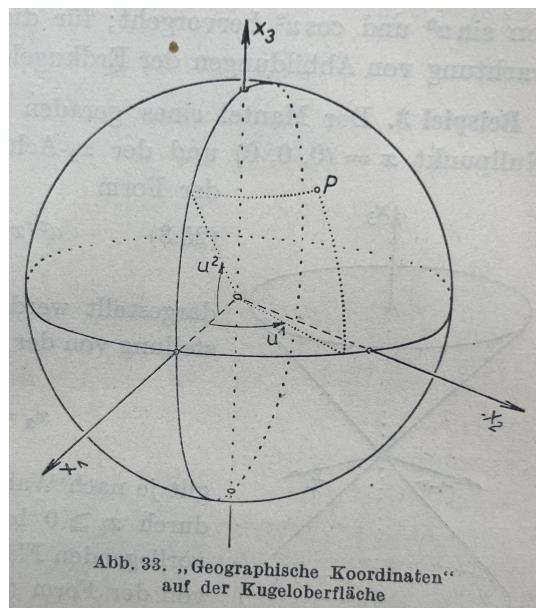
Exercise 23. (Sphere) The *r-sphere* is the 2D parametric surface defined by

$$\phi(u, v) = (r \cos u \sin v, r \sin u \sin v, r \cos v)$$

with patch region $u \in [0, 2\pi)$, $v \in [0, \pi)$ and r a positive real constant.

Abb. 33. „Geographische Koordinaten"
auf der Kugeloberfläche

Figure 7: Fine drawing of the sphere in an old book by KREYSZIG [15, p.93] with 'geographic coordinates' on the surface of the sphere. Lexicon: $\begin{array}{c} Kreyszig:\ u^1\ u^2 \\ here:\quad u\ \ v \end{array}$

a. Reproduce the plot of the sphere in WOLFRAM [52] using CALCPLOT3D and $r = 1$.
Compare the parametrization used by WOLFRAM with that of CALCPLOT3D at
≡ Examples > Parametric Surfaces > Sphere >.
b. Calculate the normal vector field N for the unit sphere i.e. $r = 1$.
c. Calculate the exterior surface normal N_p at $p = (\pi, \pi)$.

Exercise 24. (Helicoid) The *helicoid* is the 2D parametric surface defined by

$$\phi(u,v) = (v\cos(\alpha u), v\sin(\alpha u), u)$$

with patch region $u \in [-\pi, \pi)$, $v \in \mathbb{R}$ and α a non-zero real constant.
a. Reproduce the plot of the helicoid in WOLFRAM [48] for $\alpha = 1$ using CALCPLOT3D.
Compare the parametrization used by WOLFRAM with that of CALCPLOT3D at
$\equiv$ Examples > Parametric Surfaces > Helicoid >.
b. Calculate the normal vector field N for the helicoid.
c. Calculate the exterior surface normal N_p at $p = (\pi, 1)$.

Exercise 25. (Catenoid) The *catenoid* is the 2D parametric surface defined by the following parametric equations:

$$\phi(u,v) = \begin{bmatrix} c\cosh\frac{v}{c}\cos u \\ c\cosh\frac{v}{c}\sin u \\ v \end{bmatrix}$$

where $u \in [0, 2\pi)$ and $v \in \mathbb{R}$ and c is a non-zero real constant.
a. Reproduce the plot of the catenoid in WOLFRAM [44] for $c = 1$ using CALCPLOT3D.
Compare the parametrizations used by WOLFRAM with that of CALCPLOT3D at
$\equiv$ Examples > Parametric Surfaces > Catenoid >.
b. Calculate the normal vector field N for the catenoid.[4]
c. Calculate the exterior surface normal N_p at $p = (\pi, 1)$.

Exercise 26. (SCHERK's surface) The SCHERK surface is the 2D parametric surface defined by

$$\phi(u,v) = (u, v, \ln(\frac{\cos v}{\cos u}))$$

where $u, v \in [0, 2\pi)$.
a. Plot the SCHERK surface using CALCPLOT3D.
b. Calculate the normal vector field N for the SCHERK surface.
c. Calculate the exterior surface normal N_p at $p = (\pi, 1)$.

---
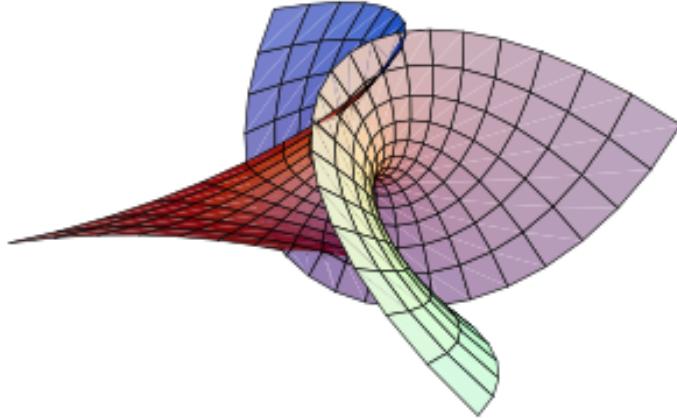
[4]WOLFRAM gives $N(u,v) = (-\cos(u)sech(v/c), -sech(v/c)\sin(u), \tanh(v/c))$,
cf. `https://mathworld.wolfram.com/HyperbolicSecant.html`. To get the concrete normal in c. you
don't has to try to get exactly this result.

Exercise 27. (ENNEPER's surface) The ENNEPER surface is the 2D parametric surface defined by

$$\phi(u,v) = (u - \frac{1}{3}u^3 + uv^2, v - \frac{1}{3}v^2 + vu^2, u^2 - v^2)$$

with patch region $u \in [-\pi, \pi)$ and $v \in \mathbb{R}$.

a.   Reproduce the plot of ENNEPER surface in WOLFRAM [47] using CALCPLOT3D.
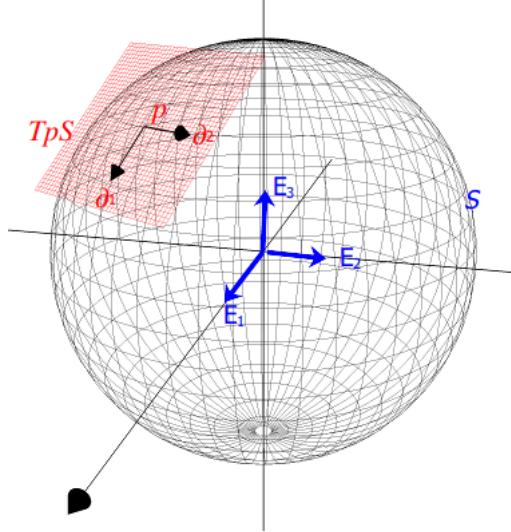
b. Calculate the normal vector field N for the ENNEPER surface.
c. Calculate the exterior surface normal N_p at $p = (1,1)$.
d. Try the parametrization Ψ which is given in WOLFRAM [47] to solve b. and c.

♡

We should now have a solid feeling
for what a normal vector (field) on a surface is.

♡

3   The First Fundamental Form

Figure 8: Visualization of the tangential plane T_pS at p on surface S with tangential basis vectors ∂_1 and ∂_2 and ambient space basis E_1, E_2, E_3 in $\mathbb{R}^3$, cf. REJBRAND [25].

To develop general methods to study properties of surfaces in ambient space, we describe the geometry of a surface by means of a parametrization $\phi\colon U \to \mathbb{R}^3$ and measurements, which are defined on the parameter set U. The so-called *metric tensor g alias the first fundamental form I is such a measurement of a surface parametrization ϕ*, that plays an crucial role in Differential Geometry and General Relativity. It allows to define and calculate distances and angles and other measurements on surfaces ('manifolds') in space, just as the inner product do it on a Euclidean space. But be warned: in LINDNER [16, p.33 ff] resp. GRINFELD [11, p.56 ff] we defined and calculated the metic tensor $\hat{g}$ for *the ambient space* $\mathbb{R}^3$ using a coordinate basis in $\mathbb{R}^3$, but now we define *a metric in a tangential space T_pS of a parametric surface S*! Looking at fig. 8 we will therefore use tangential basis vectors in T_pS to construct the metric tensor of the parametrization.

Remark. 1.   The components of the surface metric tensor g in a tangential space basis ∂_1, ∂_2 is a symmetric matrix (tensor), whose entries transform 'covariantly' under changes to the coordinate system. Thus a surface metric tensor is a covariant symmetric tensor of type $(0, 2)$.
2.   The surface metric tensor is heavily used to do so-called *index juggling*, cf. [11, p.88], i.e. tensor operations of *raising* or *lowering* an index, see [16], [2, p.335 ff], [22, p.25 ff].
For the definition of g only the tangential space basis ϕ_u, ϕ_v of ϕ will be necessary, the surface normal vector N will occur crucial in the definition of the *second fundamental form II* alias the 2nd fundamental tensor h in the next chapter.

3.1   Definition

Let $\phi\colon U \to \mathbb{R}^3$ be a parametrization of an 2-dimensional parametric surface S.
Let's denote the parameters (u, v) for the moment tensorial by (u^1, u^2) and the tangential
space basis vectors of T_pS by

$$\partial = (\partial_1, \partial_2) := (\phi_{u^1}, \phi_{u^2}) \equiv (\frac{\partial\phi}{\partial u^1}, \frac{\partial\phi}{\partial u^2}) \tag{3.1}$$

Then the (covariant) *surface metric tensor* $g = (g_{ij})$ alias the *first fundamental form* I (the
"fff") is defined as the matrix of the pairwise dot products of these tangential space basis
vectors, i.e.

$$g_{ij} := \partial_{\mathsf{i}} \bullet \partial_{\mathsf{j}} \tag{3.2}$$

Remark.
1. The metric tensor is also called GRAM's matrix, therefore the notion g_{ij}.
2. We have the explicit calculation formula using the parameterization ϕ of surface S

$$g_{ij} = \frac{\partial\phi}{\partial u^i} \bullet \frac{\partial\phi}{\partial u^j} \tag{3.3}$$

3. The following lexicon helps to memorize these conventions.

LEXICON	*Math*	EIGENMATH
covariant surface metric tensor g:	g_{ij}	gdd or simply g
contravariant surface metric tensor g^{-1}:	g^{ij}	guu or gu or inv(g)

In EIGENMATH we try to mimic the tensor notion of $_{ij}$ resp. ^{ij} resp. ^i_j by the identifier dd
(read: 'down down') resp. uu (read: 'up up') resp. ud (read: 'up down').[5]
4. The metric components of g are mostly denoted in classic GAUSS writing as

$$E := g_{11} = g(\partial_1, \partial_1)$$
$$F := g_{12} = g(\partial_1, \partial_2)$$
$$G := g_{22} = g(\partial_2, \partial_2)$$

Ergo we have $g = \begin{bmatrix} E & F \\ F & G \end{bmatrix}$ and $g^{-1} = \frac{1}{E \cdot G - F^2} \cdot \begin{bmatrix} G & -F \\ -F & E \end{bmatrix}$.
Cf. e.g. [23, p.98].
5. We prefer the notation *surface metric tensor* for g. But if one thinks metaphorically at
calculating small distances 'ds' between 'two infinitesimally close' points[6] on the surface
resp. the tangential plane, one gets the following expression for the points distance

$$I \equiv ds^2 = Edu^2 + 2Fdudv + Gdv^2 = E \cdot du \wedge du + 2F \cdot du \wedge dv + GE \cdot dv \wedge dv \tag{3.4}$$
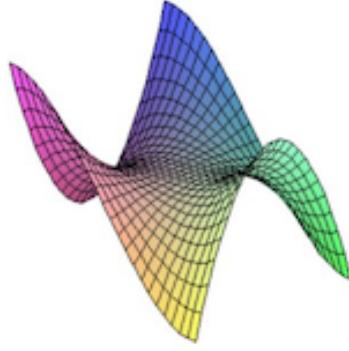
which is the bilinear noted variant of g. This motivates the name choice '1st fundamental
form' I. Using such differential forms to calculate line length, areas or volumes using
EIGENMATH is demonstrated in [16]. Therefore we will not discuss measurements like
distances or volumes here. Nevertheless you find an example here [8, p.14 ff], [19, p.172].

---

[5]This convention was introduced by G. WEIGT in his script about the SCHWARZSCHILD metric, cf. [42].
[6]More about this route e.g. in [4, p.170 ff] or [23, p.98].

3.2   Implementation and Examples

Example 8. (The surface metric tensor g for the monkey saddle) One of the most elementary examples is the 2-dimensional monkey saddle, cf. RECKZIEGEL [24, p.103],

or google for ▷wolfram monkey saddle.
A parametrization for the monkey saddle is

$$\phi \colon U \;\; \to \;\; \mathbb{R}^3$$
$$(u,v) \;\; \mapsto \;\; (u,v,u^3 - 3uv^2)$$

We use EIGENMATH to calculate the surface metric tensor g and the first fundamental form $I \equiv ds^2$.

```
# EIGENMATH
"Monkey saddle"

phi(u,v) = (u,v, u^3 - 3u v^2)

du = d(phi,u)      -- partial derivative of phi w.r.t. u, cf. eq. (3.2)
dv = d(phi,v)

E = dot(du,du)     -- dot product of basis vectors
E

F = dot(du,dv)
F

G = dot(dv,dv)
G

g=((E,F),(F,G))    -- fff in matrix shape
g

guu = inv(g)       -- the contravariant fff
guu
```

```
    -- alternativ construction

g = zero(2,2)
    g[1,1] = dot(du,du)
    g[1,2] = dot(du,dv)
    g[2,1] = g[1,2]
    g[2,2] = dot(dv,dv)


g

    -- fff I alias ds^2 in differential form (line element)
ds2 = E*dudu + 2*F*dudv + G*dvdv
ds2
```

● ▷ Click here to RUN this script.

EIGENMATH output:

Monkey saddle

$$E = 9\,u^4 - 18\,u^2\,v^2 + 9\,v^4 + 1$$

$$F = -18\,u^3\,v + 18\,u\,v^3$$

$$G = 36\,u^2\,v^2 + 1$$

$$g = \begin{bmatrix} 9\,u^4 - 18\,u^2\,v^2 + 9\,v^4 + 1 & -18\,u^3\,v + 18\,u\,v^3 \\[2mm] -18\,u^3\,v + 18\,u\,v^3 & 36\,u^2\,v^2 + 1 \end{bmatrix}$$

and (fff) displays the fff in classic differential form, where $ds2$ in (fff) means ds^2 etc.

$$d_{s2} = d_{udu} + 9\,d_{udu}\,u^4 - 18\,d_{udu}\,u^2\,v^2 + 9\,d_{udu}\,v^4 - 36\,d_{udv}\,u^3\,v + 36\,d_{udv}\,u\,v^3 + d_{vdv} + 36\,d_{vdv}\,u^2\,v^2$$

Exercise 28. a. Reproduce the plot of the monkey saddle with CALCPLOT3D.
b. A textbook gives the first fundamental form "I" of the monkey saddle by
$$I = \left[1 + \left(3\,u^2 - 3\,v^2\right)^2\right]d\,u^2 - 2\left[18\,u\,v\left(u^2 - v^2\right)\right]d\,u\,d\,v + \left(1 + 36\,u^2\,v^2\right)d\,v^2$$
How to reconstruct the coefficients E, F, G from I?

3.3   Exercises

Exercise 29. Calculate the surface metric tensor g for the 'saddle' surface $z = xy$. *Hint*: parametrization ϕ by $\phi(u, v) = (u, v, u \cdot v)$.

Exercise 30. Let S denote the surface parametrized by $\phi(u, v) = (u, v, u^2 + v^2)$, $u, v \in \mathbb{R}$. Calculate the surface metric tensor g and the fff I for this surface.

Exercise 31. (The surface metric tensor g for a *function surface*)
A parameterization of the graph of f with $z = f(x, y)$ is $\phi(u, v) = (u, v, f(u, v))$.
a. Verify with EIGENMATH:

$$E = 1 + f_u^2 \qquad F = f_u \cdot f_v \qquad G = 1 + f_v^2$$

b. Calculate the surface metric tensor g and the fff I for the function graph.

Exercise 32. (The surface metric tensor g for the *elliptic paraboloid*)
The *elliptic paraboloid* is the parametric surface defined by $\phi(u, v) = (u, v, u^2 + v^2)$.
a. Verify the coefficients of g using EIGENMATH: $E = 1 + 4u^2, F = 4uv, G = 1 + 4v^2$.
b. Then write down the fff as $I = ds^2 = \ldots$.

- You may cross-check your results of the following exercises in [17].

- *Calculate the coefficients of the surface metric tensor $g = \begin{bmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{bmatrix} = \begin{bmatrix} E & F \\ F & G \end{bmatrix}$ alias the 1st fundamental form I using* EIGENMATH *for the following surfaces . . .*

Exercise 33. the cylinder $\phi(u, v) = (a \cos(v), a \sin(v), u)$. Choose $a = 2$.

Exercise 34. the cone $\phi(u, v) = (u \cos(v), u \sin(v), u)$.
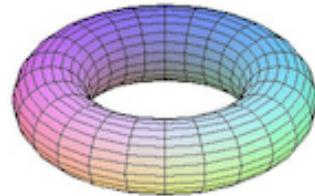
Exercise 35. hyperbolic paraboloid $\phi(u, v) = (u, v, u^2 - v^2)$.

Exercise 36. the helicoid $\phi(u, v) = (v \cos(\alpha u), v \sin(\alpha u), u)$. Choose $\alpha = 1$.

Exercise 37. the unit sphere $\phi(u, v) = (\cos(u) \sin(v), \sin(u) \sin(v), \cos(v))$.

Exercise 38. the catenoid $\phi(u, v) = (c \cosh \frac{v}{c} \cos u, c \cosh \frac{v}{c} \sin u, v)$.

Exercise 39. the torus $\phi(u, v) = ((b + a \cos(v)) \cos(u), (b + a \cos(v)) \sin(u), a \sin(v))$.

Verify: $E = (b + a \cos(v))^2$, $F = 0$, $G = a^2$.

Exercise 40. the PLÜCKER Conoid $\phi(u, v) = (v \cdot \cos(u), v \cdot \sin(u), 2 \cos(u) \cdot \sin(u))$.

Exercise 41. the SCHERK surface $\phi(u,v) = (u, v, \ln(\frac{\cos v}{\cos u}))$.

Exercise 42. the ENNEPER surface $\phi(u,v) = (u - \frac{1}{3}u^3 + uv^2, v - \frac{1}{3}v^2 + vu^2, u^2 - v^2)$.

Exercise 43. the BELTRAMI pseudosphere
$$\phi(u,v) = (\sin u \cos v, \sin u \sin v, \cos u + \ln(\sin u(\cos u + 1))).$$

Exercise 44. the hexenhut $\phi(u,v) = (\frac{\alpha}{\sqrt{u}} \cdot \cos v, \frac{\alpha}{\sqrt{u}} \cdot \sin v, u)$ where $\alpha^2 = \frac{2}{3\sqrt{3}}$.
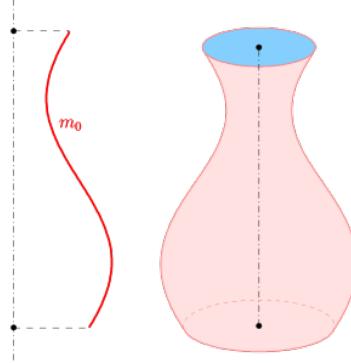
Exercise 45. the BIANCHI surface

$$\phi(u,v) = \begin{bmatrix} \frac{2 \cdot \sqrt{v^2+1} \cdot \sin(u) \cdot \cos(-v+\arctan(v))}{1+v^2 \sin(u)^2} \\ \frac{-2 \cdot \sqrt{v^2+1} \cdot \sin(u) \cdot \sin(-v+\arctan(v))}{1+v^2 \sin(u)^2} \\ \ln(\tan(\frac{1}{2u})) + \frac{2 \cdot \cos(u)}{1+v^2 \sin(u)^2} \end{bmatrix} \tag{3.5}$$

Remark. Check your results e.g. by visiting ▷ REJBRAND's Encyclopedia.

Exercise 46. (The surface metric tensor g for a *surface of revolution*)
A *surface of revolution* is obtained by rotating a graph of $f :]a, b[\to \mathbb{R}$ around an coordinate axis.

A parameterization is given by

$$\phi(u,v) = (f(u)\cos(v), f(u)\sin(v), g(u)) \tag{3.6}$$

Calculate the surface metric tensor g for a surface of revolution parametrized by (3.6).

Remark. For more information see e.g. ▷ Wolfram: SurfaceOfRevolution, [2, p.113], [15, p.218], [23, p.80], [24, p.77],

Exercise 47. (*The surface metric tensor g for a ruled surface*)
A *ruled surface* is a surface S that can be swept out by moving a line in space, i.e it is a surface which is a collection of straight lines, called the *rulings* of the surface:
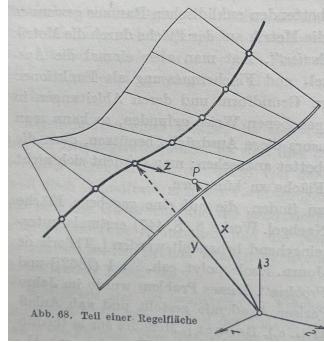
Abb. 68. Teil einer Regelfläche   KREYSZIG, p. 218

Let $c\colon\,]a,b[\to \mathbb{R}^3$ be a parametrization of a curve with $c(u) \in S$, i.e. each point of c meets a ruling. Let $E(u) \in \mathbb{R}^3$ be a non-zero unit vector positioned at $c(u)$.
Then each point of the surface S is given by the parametrization

$$\phi(u,v) = c(u) + v \cdot E(u) \tag{3.7}$$

• E.g. PLÜCKER's Conoid can be written as a ruled surface via the decomposition

$$\phi(u,v) = \begin{bmatrix} v \cdot \cos(u) \\ v \cdot \sin(u) \\ 2\cos(u) \cdot \sin(u) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 2\cos(u) \cdot \sin(u) \end{bmatrix} + v \cdot \begin{bmatrix} \cos(u) \\ \sin(u) \\ 0 \end{bmatrix} \tag{3.8}$$

Remark. ▷ wolfram: Ruled Surface

Exercise 48. (*ruled surfaces vs. surfaces of revolution*)
Some of the classical surfaces studied above are constructible as a surface of revolution or as ruled surface. Fill in and complete the following table:

LEXICON surface is . . .	*ruled with c and $\vec{r}$*	*revolution with f and g*
$c = (0,0,2\cos u \sin u)$ $E = (\cos u, \sin u, 0)$:	PLÜCKER*'s Conoid*	
$f(u) = g(u) = u$:		*cone*
$f(u) = r = const$ $g(u) = u$:		*cylinder*
$f = \cos u$ $g = \sin u$:		*unit sphere*
. . .:	. . .	. . .
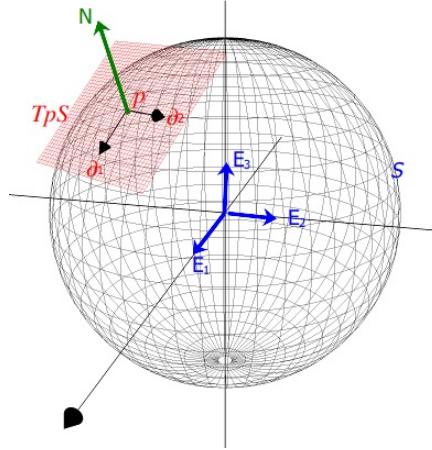
with $\ldots \in \{$helicoid, catenoid, hyp. paraboloid, torus, SCHERK, ENNEPER, pseudosphere, ..$\}$.

a. Calculate the surface metric tensor g for a general defined ruled surface by (3.7).
b. Verify the results of the calculated values in exercise $33 - 45$ using the general formula.

Remark. ▷ wiki: Ruled surface.

4   The Second Fundamental Form

Once again, now with N:

Figure 9:   Visualization of the tangential plane T_pS at p on surface S with tangential basis vectors $\partial_{\mathbf{1}}$ and ∂_2, ambient space basis E_1, E_2, E_3 and normal vector N in $\mathbb{R}^3$, cf. REJBRAND [25].

To develop general methods to study properties and the shape of surfaces in ambient space, we describe the geometry of a surface by means of a parametrization $\phi\colon U \to \mathbb{R}^3$ and measurements, which are defined on the parameter set U. The metric tensor g alias the fff I allowed to measure distances, angles and areas on S.

To measure how 'curved' a surface is, we introduced the GAUSS map resp. the surface normal vector field N and got a first feeling of the curvature of S by watching the moving N on the surface. To do this, we have to *consider the surface from the outside* by means of the *exterior* normal N, i.e. look at S as an object of the ambient space $\mathbb{R}^3$, see fig.9.

Various terms have emerged to cope with the curvature of S, each of which sheds light on different aspects: the GAUSSian curvature, the mean curvature, the normal curvature, the principal curvatures etc. This seems quite confusing at first, but fortunately all these characteristics can ultimately be traced back to the so-called *shape operator A* and the so-called *2nd fundamental tensor h* alias the 2nd *fundamental form* (the "sff") II.

So the precise tools to measure the curvature are the sff h and the shape operator A, which will be defined and implemented first. For the definition of the surface metric i.e. the fff g, there was only the tangential space basis of the parameterization ϕ necessary, whereas the surface normal vector N is now crucial in the construction of the 2nd *fundamental form h* and the shape operator A.

All this will be studied in this chapter.

4.1   Definition

Let $\phi\colon U \to \mathbb{R}^3$ be a parametrization of an 2-dimensional parametric surface S.
Let ϕ_u and ϕ_v be the canonical basis in the tangential space T_pS.
Let $g = g_{ij}$ be its surface metric tensor and $V := \sqrt{det(g)}$ ist volume element.
Let $\vec{n} := \frac{\phi_u \times \phi_v}{|\phi_u \times \phi_v|}$ be the unit normal vector field on T_pS.
Let's denote the parameters (u, v) for the moment 'tensorial' by (u^1, u^2).
We then define the coefficients of the *2nd fundamental tensor h* by

$$h_{ik} := \phi_{u^i u^k} \bullet \vec{n} \tag{4.1}$$

or more explicit

$$h_{ik} := \frac{\partial^2 \phi}{\partial u^i \partial u^k} \bullet N = \frac{1}{V} \cdot det(\frac{\partial \phi}{\partial u^1}, \frac{\partial \phi}{\partial u^2}, \frac{\partial^2 \phi}{\partial u^i \partial u^k}) \tag{4.2}$$

Remark.
1. Equation (4.1) is in PRESSLEY [23, p.124], equation (4.2) in RECKZIEGEL [24, p.110].
2. The components of the 2nd fundamental tensor h are denoted after GAUSS by

$$L := h_{11} = \phi_{uu} \bullet \vec{n}$$
$$M := h_{12} = \phi_{uv} \bullet \vec{n}$$
$$N := h_{22} = \phi_{vv} \bullet \vec{n}$$

Ergo we have $h = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} = \begin{bmatrix} L & M \\ M & N \end{bmatrix} = \begin{bmatrix} \phi_{uu} \bullet \vec{n} & \phi_{uv} \bullet \vec{n} \\ \phi_{uv} \bullet \vec{n} & \phi_{vv} \bullet \vec{n} \end{bmatrix}$.
The following lexicon helps to identify these coefficients L, M, N in other textbooks, which use sometimes other letters:

LEXICON	Math	EIGENMATH
GAUSS:	L, M, N	L,M,N
DINEEN [7, 201], SHIFRIN [32, p.45 ff]:	l, m, n	l,n,m
PRESSLEY [23, 124] or [22, p.94] or [19, p.175] or [15, p.146]:	L, M, N	L,M,N
BANCHOFF [2, 180], WOLFRAM *mathworld*:	e, f, g	e,f,g

Using EIGENMATH we prefer mostly L, M, N and therefore denote the normal vector in a mathematical context by $\vec{n}$ and in EIGENMATH scripts by n to avoid identifier collision with the surface normal N.
3. One calls the expression

$$II := Ldu2 + 2Mdudv + Ndv^2 \equiv L \cdot du \wedge du + 2M \cdot du \wedge dv + N \cdot dv \wedge dv$$

the *second fundamental form* (abbreviated sff) of ϕ.
4. For more information on $h = II$ see for example [23, p.98], [2, p.178 ff] or [19, p.171].

4.2   Implementation and Examples

These constructions should become clear by studying some examples with EIGENMATH.

Example 9. (the second fundamental tensor of the surface $z = xy$)
We use EIGENMATH to calculate the second fundamental tensor h and the second funda-
mental form II for the surface $S\colon z = xy$, cf. [7, p.202].
The parametrization for the function surface S is $\phi(u, v) = (u, v, u \cdot v)$.

Calculate the coefficients L, M, N of the 2nd fundamental tensor h. Determine I.

Solution.   We use EIGENMATH.
1st we use RHS of equation (4.2) and use the well known identity of linear algebra
$det(\phi_1, \phi_2, \phi_{ik}) = (\phi_1 \times \phi_2) \bullet \phi_{ik}$ in (1), wherein $\phi_1 \times \phi_2$ is the normal vector (not yet
normed to length 1):

```
# EIGENMATH
phi(u,v) = (u,v, u*v)
phi

"surface metric tensor"
g = zero(2,2)
g[1,1] = dot(d(phi,u), d(phi,u))
g[1,2] = dot(d(phi,u), d(phi,v))
g[2,1] = g[1,2]
g[2,2] = dot(d(phi,v), d(phi,v))
g

"inverse of g (could be simplified by mind)"
ginv = inv(g)
ginv

V= sqrt(det(g))
V

"we use defining equation (4.2)"
-- until now we do not use N explicit, so use name N for coefficient

L = 1/V * dot( cross(d(phi,u), d(phi,v)), d(phi,u,u))    --(1)
L
M = 1/V * dot( cross(d(phi,u), d(phi,v)), d(phi,u,v))
M
N = 1/V * dot( cross(d(phi,u), d(phi,v)), d(phi,v,v))
N
```

● ▷ Click here to RUN this script.

EIGENMATH output:

$$L = 0$$

$$M = \frac{1}{\left[u^2 + v^2 + 1\right]^{1/2}}$$

$$N = 0$$

$\mathbf{2}^{nd}$ We use LHS of equation (4.2) resp. equation (4.1) and to avoid the $det(..)$ – which incorporate implicit the normal vector $N = \vec{n}$ using the cross product – we now have to calculate explicit the normal vector $\vec{n}$:

```
# EIGENMATH
"surface metric tensor alias fff"
g = zero(2,2)
g[1,1] = dot(d(phi,u), d(phi,u))
g[1,2] = dot(d(phi,u), d(phi,v))
g[2,1] = g[1,2]
g[2,2] = dot(d(phi,v), d(phi,v))
g

"unit normal vector"
N    = cross(d(phi,u),d(phi,v))
N
absN = abs(N)
absN
N    = N/absN
N

"sff in matrix/tensor shape using (4.1)"
h = zero(2,2)
h[1,1] = dot( d(phi,u,u), N)
h[1,2] = dot( d(phi,u,v), N)
h[2,1] = h[1,2]
h[2,2] = dot( d(phi,v,v), N)
h

l = h[1,1]
l
m = h[1,2]
m
n = h[2,2]    -- use n because N is normal vector
n
```

● ▷ Click here to RUN this script.

EIGENMATH output:

$$l = 0$$
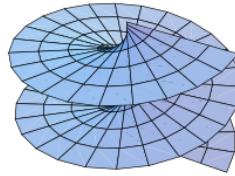
$$m = \frac{1}{\left[u^2 + v^2 + 1\right]^{1/2}}$$

$$n = 0$$

It follows: $I = 2 \cdot \frac{1}{\sqrt{u^2+v^2+1}}$.

Example 10. (the second fundamental tensor of the helicoid)
The *helicoid* is the 2D parametric surface defined by $\phi(u,v) = (v\cos(u), v\sin(u), u)$.

Calculate the coefficients l, m, n of the 2nd fundamental tensor h.

Solution.   We use EIGENMATH. We use equation (4.1). Therefore we have to calculate g, $N := \vec{n}$ and h. We name the coefficients of the second fundamental form l, m, n to avoid name clash with the normal vector N.

```
"HELICOID"
phi(u,v) = (v cos(u),v sin(u), u)
phi

"surface metric tensor"
g = zero(2,2)
g[1,1] = dot(d(phi,u), d(phi,u))
g[1,2] = dot(d(phi,u), d(phi,v))
g[2,1] = g[1,2]
g[2,2] = dot(d(phi,v), d(phi,v))
g

"unit normal vector"
N    = cross(d(phi,u),d(phi,v))
N
absN = abs(N)
absN
N    = N/absN
N

"h in matrix/tensor shape"
h = zero(2,2)
```

```
h[1,1] = dot( d(phi,u,u), N)
h[1,2] = dot( d(phi,u,v), N)
h[2,1] = h[1,2]
h[2,2] = dot( d(phi,v,v), N)
h

l = h[1,1]
l
m = h[1,2]
m
m = simplify(m)
m
n = h[2,2]
n
```

● ▷ Click here to RUN this script.

EIGENMATH output:

$$l = 0$$

$$m = \cfrac{1}{\left[v^2 \cos(u)^4 + 2\,v^2\,\text{co}\right.}$$

$$\cfrac{1}{\left[v^2 + 1\right]^{1/2}}$$

$$n = 0$$

4.3   Exercises

• You may cross-check your results of the following exercises in [17].

Exercise 49. Calculate the 2nd fundamental tensor h for the unit sphere S.
Look for ▷ WOLFRAM: sphere and use their parameterization.
Check your result with that given by WOLFRAM.
You may switch inside this page to WOLFRAM|alpha.

Exercise 50. Calculate the 2nd fundamental tensor h for the catenoid S.
Look for ▷ WOLFRAM: catenoid and use their parameterization.
Check your result with that given by WOLFRAM.
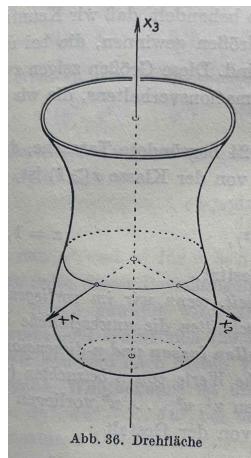
Exercise 51. Calculate the 2nd fundamental tensor h for the torus S.
Look for ▷ WOLFRAM: torus and use their parameterization with $a = 1$.
Check your result with that given by WOLFRAM.

Exercise 52. (The 2nd fundamental tensor h for a *surface of revolution*)
The surface is obtained by rotating the graph of $f\colon\,]a, b[\to \mathbb{R}$ around an axis.

Abb. 36. Drehfläche   taken from KREYSZIG, p.107

The parameterization is known to be $\phi(u, v) = (f(u)\cos(v), f(u)\sin(v), g(u))$.
Calculate the 2nd fundamental tensor h for a surface of revolution.

Exercise 53. (The 2nd fundamental tensor h for a *function surface*)
Let $z = f(x, y)$ be a function surface.
A parameterization of the graph of f is $\phi(u, v) = (u, v, f(u, v))$.
a. Verify with EIGENMATH the coefficients of h:

$$L = (1 + f_u^2 + f_u^2)^{-2} \cdot f_{uu}$$
$$M = (1 + f_u^2 + f_u^2)^{-2} \cdot f_{uv}$$
$$N = (1 + f_u^2 + f_u^2)^{-2} \cdot f_{uv}$$

b. Give h in matrix shape and write down the sff for the function graph.

Exercise 54. (The 2nd fundamental tensor h for the *monkey saddle*).
A parametrization for the monkey saddle is $\phi\colon U \to \mathbb{R}^3$ by $(u,v) \mapsto (u, v, u^3 - 3uv^2)$.
a. Give the second fundamental tensor h in matrix shape (h_{ij}).
b. Write down the sff in differential form $II = \ldots$.
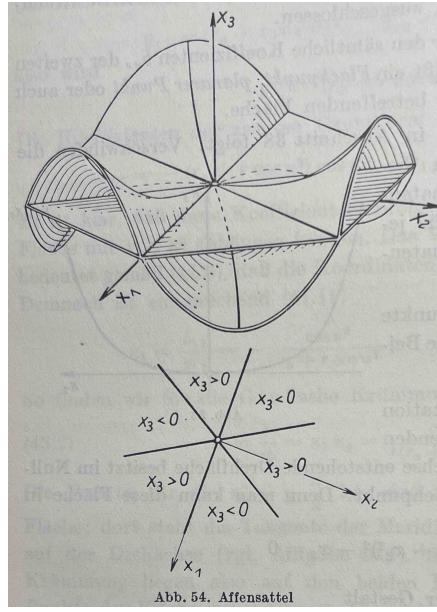
Abb. 54. Affensattel   taken from KREYSZIG, p.168

• You may cross-check your results of the following exercises in [17].
• *Calculate the coefficients of the 2nd fundamental tensor* $h = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} = \begin{bmatrix} L & M \\ M & N \end{bmatrix}$ *alias the*
2nd fundamental form II using EIGENMATH *for the following 2D surfaces* . . .

Exercise 55. the saddle $\phi(u,v) = (u, v, uv)$.

Exercise 56. the cylinder $\phi(u,v) = (a\cos(v), a\sin(v), u)$. Choose $a = 1$.

Exercise 57. the cone $\phi(u,v) = (u\cos(v), u\sin(v), u)$.

Exercise 58. the elliptic paraboloid $\phi(u,v) = (u, v, u^2 + v^2)$.

Exercise 59. the hyperbolic paraboloid $\phi(u,v) = (u, v, u^2 - v^2)$.

Exercise 60. the unit sphere $\phi(u,v) = (\cos(u)\sin(v), \sin(u)\sin(v), \cos(v))$.

Exercise 61. the PLÜCKER conoid $\phi(u,v) = (v \cdot \cos(u), v \cdot \sin(u), 2\cos(u) \cdot \sin(u))$.
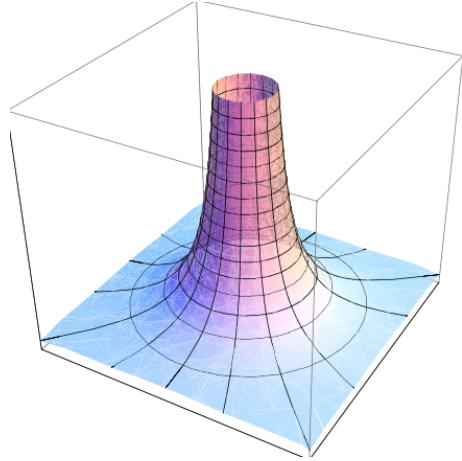
Exercise 62. the SCHERK surface $\phi(u,v) = (u, v, \ln(\frac{\cos v}{\cos u}))$.

Exercise 63. the ENNEPER surface $\phi(u,v) = (u - \frac{1}{3}u^3 + uv^2, v - \frac{1}{3}v^2 + vu^2, u^2 - v^2)$.
See [8, p.22].

Exercise 64. the BELTRAMI pseudosphere
$$\phi(u,v) = (\sin u \cos v, \sin u \sin v, \cos u + \ln(\sin u(\cos u + 1))).$$

Exercise 65. the hexenhut $\phi(u,v) = (\alpha/\sqrt{u} \cdot \cos v, \alpha/\sqrt{u} \cdot \sin v, u)$ where $\alpha^2 = \frac{2}{3\sqrt{3}}$, see WHEELER [43, p.21 ff].

Exercise 66. the BIANCHI surface

$$\phi(u,v) = \begin{bmatrix} \frac{2 \cdot \sqrt{v^2+1} \cdot \sin(u) \cdot \cos(-v+\arctan(v))}{1+v^2 \sin(u)^2} \\ \frac{-2 \cdot \sqrt{v^2+1} \cdot \sin(u) \cdot \sin(-v+\arctan(v))}{1+v^2 \sin(u)^2} \\ \ln\left(\tan\left(\frac{1}{2u}\right)\right) + \frac{2 \cdot \cos(u)}{1+v^2 \sin(u)^2} \end{bmatrix}$$

Remark. Check your results also e.g. by visiting ▷ REJBRAND Encyclopaedia.

5 Shape Operator and WEINGARTEN Map

We now implement the crucial tool to measure the curvature of a surface parametrized by the parametrization $\phi\colon U \to \mathbb{R}^3$, the so-called shape operator A. An also used name for A is WEINGARTEN map. To construct A we need both, the fff g and the sff h.

5.1 Definition

Let $\phi\colon U \to \mathbb{R}^3$ be a parametrization of an 2-dimensional parametric surface S.
Let ϕ_u and ϕ_v be the canonical basis in the tangential space T_pS.
Let $g = g_{ij}$ be its surface metric tensor and $g^{-1} = g^{ij}$, its inverse matrix.
Let $h = h_{ij}$ be its 2nd fundamental tensor.

Then we define the coefficients of the *shape operator $A \in \mathbb{R}^{2\times2}$*
alias the WEINGARTEN map A by

$$A_{ik} := \sum_{j=1}^{2} g^{ij} \cdot h_{ik} \tag{5.1}$$

where $i = 1, 2$ and $k = 1, 2$.

Remark.
1. Equation (5.1) is from RECKZIEGEL [24, p.137].

2. Using the coefficients E, F, G of g and L, M, N of h we have explicit

$$A = g^{-1} \bullet h = \begin{bmatrix} E & F \\ F & G \end{bmatrix}^{-1} \bullet \begin{bmatrix} L & M \\ M & N \end{bmatrix} \tag{5.2}$$

$$= \frac{-1}{E \cdot G - F^2} \cdot \begin{bmatrix} G & -F \\ -F & E \end{bmatrix} \bullet \begin{bmatrix} L & M \\ M & N \end{bmatrix}$$

i.e.

$$A = \frac{1}{EG - F^2} \cdot \begin{bmatrix} GL - FM & GM - FN \\ EM - FL & EN - FM \end{bmatrix} \tag{5.3}$$

cf. OLOFF [22, p.108].

3. For more information about the shape operator see e.g. BANCHOFF [2, p.186], THORPE [36, p.55], PRESSLEY [23, p.139], OLOFF [22, p.104 ff] or SHIFRIN [32, p.45 ff].

5.2 Examples

Example 11. (the shape operator of the saddle surface $z = xy$)
A parametrization for this function surface S is $\phi\colon U \to \mathbb{R}^3$ with $(u, v) \mapsto (u, v, u \cdot v)$.
Use EIGENMATH to calculate the shape operator A for the surface $S\colon z = xy$.

Solution.   We use equation (5.2) and calculate at first g and h, then A.

```
# EIGENMATH
phi(u,v) = (u,v, u*v)

"fff alias surface metric tensor g"
g = zero(2,2)
g[1,1] = dot(d(phi,u), d(phi,u))
g[1,2] = dot(d(phi,u), d(phi,v))
g[2,1] = g[1,2]
g[2,2] = dot(d(phi,v), d(phi,v))

"inverse of g (could be simplified by mind)"
ginv = inv(g)
ginv

V = 1/sqrt(det(g))

"sff in tensor shape"
h = zero(2,2)
h[1,1] = 1/V* dot( cross(d(phi,u), d(phi,v)), d(phi,u,u))
h[1,2] = 1/V* dot( cross(d(phi,u), d(phi,v)), d(phi,u,v))
h[2,1] = h[1,2]
h[2,2] = 1/V* dot( cross(d(phi,u), d(phi,v)), d(phi,v,v))

"shape operator"
A = dot(ginv, h)
A
```

● ▷ Click here to RUN this script.

EIGENMATH output:

$$
A = \begin{bmatrix} -\dfrac{u\,v}{\left[u^2 + v^2 + 1\right]^{1/2}} & \dfrac{u^2}{\left[u^2 + v^2 + 1\right]^{1/2}} + \dfrac{1}{\left[u^2 + v^2 + 1\right]^{1/2}} \\[4ex] \dfrac{v^2}{\left[u^2 + v^2 + 1\right]^{1/2}} + \dfrac{1}{\left[u^2 + v^2 + 1\right]^{1/2}} & -\dfrac{u\,v}{\left[u^2 + v^2 + 1\right]^{1/2}} \end{bmatrix}
$$

Example 12. (the shape operator of the helicoid)
The *helicoid* is the 2D parametric surface defined by $\phi(u,v) = (v\cos(u), v\sin(u), u)$.
Use EIGENMATH to calculate the shape operator A for the helicoid.

Solution.   We use EIGENMATH. We know from previous examples the coefficients of the first and the second fundamental form of ϕ.

```
"shape operator HELICOID"
-- coefficients of surface metric g
E = 1
F = 0
G = v^2+1

-- coefficients of h
L = 0
M = (v^2+1)^(-1/2)
N = 0

-- we use (5.2)
A = dot( inv(((E,F),(F,G))) , ((L,M),(M,N)) )
A
```

● ▷ Click here to RUN this script.

EIGENMATH output:

shape operator HELICOID

$$A = \begin{bmatrix} 0 & \dfrac{1}{\left[v^2 + 1\right]^{1/2}} \\ \dfrac{1}{\left[v^2 + 1\right]^{3/2}} & 0 \end{bmatrix}$$

Exercise 67. (the shape operator of the elliptic paraboloid)
The *elliptic paraboloid* is the parametric surface defined by $\phi(u,v) = (u, v, u^2 + v^2)$.
a. Verify using EIGENMATH: $E = 1 + 4u^2, F = 4uv, G = 1 + 4v^2$.
b. Verify using EIGENMATH: $L = 2(4u^2 + 4v^2 + 1)^{-1/2}, M = 0, N = L$.
c. Calculate the shape operator for the elliptic paraboloid along the pattern of example 11 and alternative along example 12.

5.3   Exercises

- You may cross-check your results of the following exercises in [17].
- *Calculate the shape operator $A = g^{-1} \bullet h$ using* EIGENMATH *for the following 2D surfaces* ...

Exercise 68. the cylinder $\phi(u,v) = (a\cos(v), a\sin(v), u)$. Choose $a = 1$.

Exercise 69. the cone $\phi(u,v) = (u\cos(v), u\sin(v), u)$.

Exercise 70. the hyperbolic paraboloid $\phi(u,v) = (u, v, u^2 - v^2)$.

Exercise 71. the monkey saddle $\phi(u,v) = (u, v, u^3 - 3uv^2)$.

Exercise 72. the unit sphere $\phi(u,v) = (\cos(u)\sin(v), \sin(u)\sin(v), \cos(v))$.

Exercise 73. the catenoid $\phi(u,v) = (c\cosh\frac{v}{c}\cos u, c\cosh\frac{v}{c}\sin u, v)$.

Exercise 74. the torus $\phi(u,v) = ((b + a\cos(v))\cos(u), (b + a\cos(v))\sin(u), a\sin(v))$.

Exercise 75. the function graph $\phi(u,v) = (u, v, f(u,v))$.

Exercise 76. the surface of revolution $\phi(u,v) = (f(u)\cos(v), f(u)\sin(v), g(u))$.

Exercise 77. the PLÜCKER conoid $\phi(u,v) = (v \cdot \cos(u), v \cdot \sin(u), 2\cos(u) \cdot \sin(u))$.

Exercise 78. the SCHERK surface $\phi(u,v) = (u, v, \ln(\frac{\cos v}{\cos u}))$.

Exercise 79. the ENNEPER surface $\phi(u,v) = (u - \frac{1}{3}u^3 + uv^2, v - \frac{1}{3}v^2 + vu^2, u^2 - v^2)$.

Exercise 80. the BELTRAMI pseudosphere
$$\phi(u,v) = (\sin u\cos v, \sin u\sin v, \cos u + \ln(\sin u(\cos u + 1))).$$

Exercise 81. the hexenhut $\phi(u,v) = (\alpha/\sqrt{u} \cdot \cos v, \alpha/\sqrt{u} \cdot \sin v, u)$.

Exercise 82. the BIANCHI surface

$$\phi(u,v) = \begin{bmatrix} \frac{2\cdot\sqrt{v^2+1}\cdot\sin(u)\cdot\cos(-v+\arctan(v))}{1+v^2\sin(u)^2} \\ \frac{-2\cdot\sqrt{v^2+1}\cdot\sin(u)\cdot\sin(-v+\arctan(v))}{1+v^2\sin(u)^2} \\ \ln(\tan(\frac{1}{2u})) + \frac{2\cdot\cos(u)}{1+v^2\sin(u)^2} \end{bmatrix}$$

Remark. One can check some of the results also by visiting ▷ REJBRAND Encyclopaedia.

6   Scalar Curvature Measures

There are many concepts that have emerged to cope with the curvature of a surface S, each of which sheds light on different aspects of the idea of curvature: the GAUSSian curvature, the mean curvature, the normal curvature, the principal curvatures etc. We now define some of these characteristics of a surface S by means of the *shape operator A* .

6.1   Definition

Let $\phi\colon U \to \mathbb{R}^3$ be a parametrization of an 2-dimensional parametric surface S.
Let g be the surface metric tensor of ϕ with the coefficients E, F, G.
Let h be the 2nd fundamental tensor of ϕ with the coefficients L, M, N.
Let A be the shape operator of ϕ.
We then define

a. The GAUSS curvature K is the real-valued function $K\colon U \to \mathbb{R}$ with

$$p \mapsto K(p) := \det(A(p)) = \frac{LN - M^2}{EG - F^2} \tag{6.1}$$

b. The mean curvature H is the real-valued function $H\colon U \to \mathbb{R}$ with

$$p \mapsto H(p) := \frac{1}{2} \cdot \operatorname{trace}(A(p)) = \frac{EN + GL - 2FM}{2(EG - F^2)} \tag{6.2}$$

c. The principal curvatures $\kappa_i \in \mathbb{R}$ of S are the roots of the equation

$$\det(A - \kappa \cdot I) = 0 \tag{6.3}$$

where $I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \in \mathbb{R}^{2\times 2}$ is the 2-by-2 identity matrix, i.e the principal curvatures are the eigenvalues of the shape operator A.
The *eigenvectors* corresponding to the the principal values κ_i are called the *principal vectors* alias *principal curvature directions*, denoted $\vec{\kappa}$.

We have the fact

Theorem. *The first and second principal curvatures κ_1 and κ_2 are the real-valued functions $\kappa_i\colon U \to \mathbb{R}$ with*

$$\kappa_1 := H - \sqrt{H^2 - K} \tag{6.4}$$

$$\kappa_2 := H + \sqrt{H^2 - K} \tag{6.5}$$

Remark. 1. Equations (6.1), (6.2) and (6.4), (6.5) are from RECKZIEGEL [24, p.115], ad equation (6.3) cf. PRESSLEY [23, p.132].
2. For more information about the curvature measures see e.g. THORPE [36, p.86–89], PRESSLEY [23, p.132; p.147 ff], OLOFF [22, p.108 ff] or SHIFRIN [32, p.49].

6.2   Examples

Example 13. (Gaussian, mean and principal curvature of the helicoid)
The *helicoid* is the 2D parametric surface defined by $\phi(u,v) = (v\cos(u), v\sin(u), u)$.
1.   Use Eigenmath to calculate the Gaussian curvature, the mean curvature and the principal curvatures for the helicoid.
2.   How much is the Gaussian and mean curvature at the point $p = (1,1)$?
3.   Write Gaussian K, mean H and principal curvatures λ as a function of an arbitrary point $p = (u,v)$ on the helicoid.
4.   Calculate the principal curvature directions $\vec{\kappa}_i(p)$ at $p = (1,1)$.

Solution.   This example is a bit longer.

$\mathbf{1}^{st}$ We use (6.1) and (6.2). We comment after the code lines and the output.

```
# EIGENMATH
"GAUSS and mean curvature of HELICOID"
-- coefficients of fff
E =1
F = 0
G = v^2+1

-- coefficients of sff
L = 0
M = (v^2+1)^(-1/2)
N = 0

-- shape operator via (5.2)
A = dot( inv(((E,F),(F,G))) , ((L,M),(M,N)) )
A                                   --(0)

"GAUSS curvature via (6.1)"
K = det(A)
K                                   --(1)

"mean curvature via (6.2)"
H = contract(A)
H                                   --(2)

"principal curvature via (6.3)"
det(A - lambda unit(2))            --(3)

lambda1 = 1/(v^2+1)                 --(4)
lambda1
lambda2 = -1/(v^2+1)
lambda2
```

EIGENMATH output:

GAUSS and mean curvature of HELICOID

$$A = \begin{bmatrix} 0 & \dfrac{1}{\left[v^2 + 1\right]^{1/2}} \\[2ex] \dfrac{1}{\left[v^2 + 1\right]^{3/2}} & 0 \end{bmatrix}$$

GAUSS curvature via (6.1)

$$K = -\dfrac{1}{\left[v^2 + 1\right]^2}$$

mean curvature via (6.2)

$$H = 0$$

principal curvature via (6.3)

$$\lambda^2 - \dfrac{1}{\left[v^2 + 1\right]^2}$$

$$\lambda_1 = \dfrac{1}{v^2 + 1}$$

$$\lambda_2 = -\dfrac{1}{v^2 + 1}$$

Comment. We took the coefficients of the fff and the sff from results before. Then the shape operator is calculated in (0) via formula (5.2). In (1) the GAUSS curvature K is calculated and in (2) the mean curvature, whereby the trace of A is given by the tensorial denoted and generalized EIGENMATH function `contract` - if wanted, one can use the self written alias function `trace(M)=contract(M)`. In (3) we determine the values of principal curvatures $\lambda_i \equiv \kappa_i$, see (4), which we get easily by mind from the presented equation.

$\mathbf{2}^{nd}$ We use EIGENMATH's `eval` function to evaluate K, H and λ at a specific point in S.

```
"K and H and lambda at p=(1,1)"
--            p:1     1
Ap = eval(A, u,1, v,1)        --(5)
Ap

Kp = det(Ap)                  --(6)
Kp

Hp = contract(Ap)            --(7)
Hp

lambda1p = eval(lambda1, u,1, v,1)  -- (8)
lambda1p
lambda2p = eval(lambda2, u,1, v,1)
lambda2p
```

EIGENMATH output:

<p style="text-align:center">K and H and lambda at p=(1,1)</p>

$$A_p = \begin{bmatrix} 0 & \dfrac{1}{2^{1/2}} \\[2ex] \dfrac{1}{2\,2^{1/2}} & 0 \end{bmatrix}$$

$$K_p = -\frac{1}{4}$$

$$H_p = 0$$

$$\lambda_{1p} = \frac{1}{2}$$

$$\lambda_{2p} = -\frac{1}{2}$$

Comment. (5) evaluates the shape operator A at $p = (1,1)$ with the values $u = 1$ and $v = 1$, which are substituted in (0). Therefore K_p, H_p and λ_p are automatically returned with its values evaluated.

3rd We again use (6.1) and (6.2) to generalize.

```
"K and H and lambda as functions at p=(u,v)"
Ap(p) = eval(A, u,p[1], v,p[2])        --(9)
Ap
Ap(1,1)

Kp(p) = det(Ap(p))                     --(10)
Kp
Kp(1,1)     -- -1/4

Hp(p) = contract(Ap(p))                --(11)
Hp(1,1)     -- 0

lambdap(p) = det(Ap(p)-x*unit(2))      --(12)
lambdap
lambdap(1,1)    -- x^2-1/4
```

EIGENMATH output:

$$\text{K and H and lambda as functions at p=(u,v)}$$

$$A_p = \begin{bmatrix} 0 & \dfrac{1}{\left[p^2 + 1\right]^{1/2}} \\[2em] \dfrac{1}{\left[p^2 + 1\right]^{3/2}} & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & \dfrac{1}{2^{1/2}} \\[1.5em] \dfrac{1}{2\,2^{1/2}} & 0 \end{bmatrix}$$

$$K_p = -\dfrac{1}{\left[p^2 + 1\right]^2}$$

$$-\frac{1}{4}$$

$$0$$

$$\lambda_p = x^2 - \dfrac{1}{\left[p^2 + 1\right]^2}$$

$$x^2 - \frac{1}{4}$$

Comment. We generalize what was done in (5) for a special point. Therefore in (9) we write p undetermined as arbitrary point on the LHS, meaning a point $p = (p[1], p[2])$ with two coordinates, which can then be used on the RHS of the defining equation (9). The special value of the GAUSS curvature at (6) is then realized via a function call with the input $(1,1)$ like $K_p(1,1)$ in (10). So we will be able to easily calculate a bundle of values of K! Look at the next example. The same is done for H_p and λ_p, which are now also functions for arbitrary points on U.

4th We use the new functions K_p, H_p and λ_p. Sorry, EIGENMATH has at the moment no build-in function for arbitrary matrices to calculate their eigenvalues and eigenvectors. So we recur on the definition of these concepts and do a small step by mind or with paper & pencil.

```
# EIGENMATH
"calculate principal directions at p=(1,1)"

roots(lambdap(1,1),x)                    --(13)
          -- [-0.5, 0.5]
Ap(1,1)-1/2*unit(2)                       --(14)
          -- ansatz for EigVevctors

dot(Ap(1,1)-1/2*unit(2), (x,y))          --(15)

dot(Ap(1,1)+1/2*unit(2), (x,y))          --(16)
```

EIGENMATH output:

<div align="center">

calculate principal directions at p=(1,1)

$$\begin{bmatrix} -\frac{1}{2} \\[2mm] \frac{1}{2} \end{bmatrix}$$

$$\begin{bmatrix} -\frac{1}{2} & \frac{1}{2^{1/2}} \\[2mm] \frac{1}{2\,2^{1/2}} & -\frac{1}{2} \end{bmatrix}$$

$$\begin{bmatrix} \frac{y}{2^{1/2}} - \frac{1}{2}x \\[2mm] \frac{x}{2\,2^{1/2}} - \frac{1}{2}y \end{bmatrix}$$

</div>

Comment. We now from command (12), that $\lambda_p(1,1) = x^2 - \frac{1}{4}$. So EIGENMATH's `roots` function calculates the roots of this quadratic equation (which we had done before in mind, but want to show its principe here ♡). Ok. In (15) we use the well known ansatz $(A - \lambda \cdot E) = O$ for the determination of eigenvectors by known eigenvalues λ_i. We take from the first line of the returned equation matrix the line

$$\frac{y}{\sqrt{2}} - \frac{1}{2}x = 0$$

which gives e.g. the eigenvector

$$\begin{bmatrix} -1 \\ \frac{-1}{\sqrt{2}} \end{bmatrix}$$

for $\lambda_1 = -1/2$.

● ▷ Click here to run RUN script.

Fact. *The principal curvatures κ_i are the roots of the equation* $\det(\mathsf{sff} - \kappa \cdot \mathsf{fff}) = 0$, *i.e.*

$$\det(h - \kappa \cdot g) = 0 \tag{6.6}$$

i.e.

$$\begin{vmatrix} L - \kappa E & M - \kappa F \\ M - \kappa F & N - \kappa G \end{vmatrix} = 0 \tag{6.7}$$

This is a simplified version of (6.3), which does not use the shape operator A at all. Cf. Pressley [23, p.132] or Sochi [31, p.121].

Exercise 83. Calculate the principal curvatures κ of the helicoid using (6.7).

Remark. In case one want a fully automatic solution for the last task, please give MAXIMA or MAXIMA$^{on\ line}$ a chance:

Figure 10:

Left: calculation of characteristic polynomial of matrix A and their associated eigenvalues and eigenvectors with CAS MAXIMA. Output line (%o3) gives the list $[-1/2, 1/2]$ of the two eigen*values* with their corresponding multiplicity of $[1, 1]$. Output line (%o4) repeats line $o3$ and then displays the list of the two eigen*vectors*.

Right: calculation of the same task using MAXIMA ONLINE via ▷ Click here to invoke MaximaOn. One has to input the line (%i4) in the input region.

Example 14. (GAUSS and mean curvature of the elliptic paraboloid)
The *elliptic paraboloid* is the 2D function surface defined by

$$\phi(u, v) = (u, v, u^2 + v^2) \tag{6.8}$$
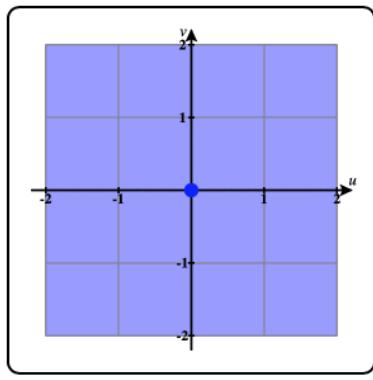
with patch region $(u, v) \in \mathbb{R}^2$ or by the parametric ansatz defined by

$$\phi(u, v) = (a\sqrt{u}\cos(v), b\sqrt{u}\sin(v), cu) \tag{6.9}$$

with real constants a, b, c and patch region $(u, v) \in \mathbb{R}^+ \times [0, 2\pi]$, cf. [31, p.14].
For this example we use the parametrization ϕ in (6.8).
a. Verify using EIGENMATH: $E = 1 + 4u^2, F = 4uv, G = 1 + 4v^2$.
b. Verify using EIGENMATH: $L = 2(4u^2 + 4v^2 + 1)^{-1/2}, M = 0, N = 2(4u^2 + 4v^2 + 1)^{-1/2}$.
c. Calculate the GAUSS curvature K and mean curvature H for the integer grid points of
the square $[-2, 2] \times [-2, 2]$ in $\mathbb{R}^2$. Discuss your observations.

d. Display a tableau of the values of the curvature K for 5 different user input points.

Solution.
a: and **b**: see above. So we have:

```
# EIGENMATH
phi(u,v) = (u,v, u^2+v^2)

-- coefficients of surface metric tensor
E = 1 + 4u^2
F = 4 u v
G = 1 + 4v^2

-- coefficients of 2nd fundamental tensor
L = 2(4u^2+4v^2+1)^(-1/2)
M = 0
N = 2(4u^2+4v^2+1)^(-1/2)

-- shape operator
```

```
    A = dot( inv(((E,F),(F,G))) , ((L,M),(M,N)) )
    K = det(A)
    H = contract(A)
```

c: we define functions Ap, Kp and Hp and prepare the grid in (1):

```
    # ...
    "K and H as functions of p=(u,v)"
    Ap(p) = eval(A, u,p[1], v,p[2])
    Kp(p) = det(Ap(p))
    Kp
    Hp(p) = contract(Ap(p))
    Hp

    "prepare grid Kuv, i.e. entry K(u,v)"
    Kuv = zero(6,6)                          --(1)
    do( Kuv[1,1]=0, Kuv[1,2]=-2, Kuv[1,3]=-1, Kuv[1,4]=0, Kuv[1,5]=1, Kuv[1,6]=2)
    do( Kuv[1,1]=0, Kuv[2,1]=-2, Kuv[3,1]=-1, Kuv[4,1]=0, Kuv[5,1]=1, Kuv[6,1]=2)
    Kuv                                      --(2)
```

EIGENMATH output:

K and H as functions of p=(u,v)

$$K_p = \frac{32\,p^2}{\left[8\,p^2 + 1\right]^3} + \frac{4}{\left[8\,p^2 + 1\right]^3}$$

$$H_p = \frac{16\,p^2}{\left[8\,p^2 + 1\right]^{3/2}} + \frac{4}{\left[8\,p^2 + 1\right]^{3/2}}$$

prepare grid

$$K_{uv} = \begin{bmatrix} 0 & -2 & -1 & 0 & 1 & 2 \\ -2 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Comment. We set the two borders of tablau (matrix) as 'axis' in (1), which are intended to be the coordinates, where the curvature values are to be read off, e.g. curvature $K(-1, 1)$ will be displayed later at position $(5, 3)$ of the matrix tableau.

Now we fill the grid with the corresponding curvature values, so we receive a distribution of K_p:

```
# ...
"fill grid"
for( i,2,6,  Kuv[2,i] = Kp(-2,i-4))   --(3) row number two
for( i,2,6,  Kuv[3,i] = Kp(-1,i-4))
for( i,2,6,  Kuv[4,i] = Kp(-0,i-4))
for( i,2,6,  Kuv[5,i] = Kp(+1,i-4))
for( i,2,6,  Kuv[6,i] = Kp(+2,i-4)) -- last row
Kuv                                   --(4)
```

EIGENMATH output:

fill grid

$$
K_{uv} =
\begin{bmatrix}
0 & -2 & -1 & 0 & 1 & 2 \\
-2 & \frac{4}{1089} & \frac{4}{1089} & \frac{4}{1089} & \frac{4}{1089} & \frac{4}{1089} \\
-1 & \frac{4}{81} & \frac{4}{81} & \frac{4}{81} & \frac{4}{81} & \frac{4}{81} \\
0 & 4 & 4 & 4 & 4 & 4 \\
1 & \frac{4}{81} & \frac{4}{81} & \frac{4}{81} & \frac{4}{81} & \frac{4}{81} \\
2 & \frac{4}{1089} & \frac{4}{1089} & \frac{4}{1089} & \frac{4}{1089} & \frac{4}{1089}
\end{bmatrix}
$$

Comment. You may read off the curvature value $Kp(1,-1) = \frac{4}{81}$ at $\texttt{Kuv}|_{45}$.

In (3), i.e. row 2 of matrix K_{uv}, we follow the curvature distribution for $u = -2$ fixed. We observe that $K_p|_{u=-2}$ is of constant value $4/1089 \approx 0.004$ - a very small value. So the surface is along $u = -2$ only minimal curved. In line $u = 0$ the curvature has a maximal value of 4: this is at the bottom of the surface, where the surface is obviously most curved. All this is in accordance with our intuitive view.

You may follow the coordinate line $u = -2$ on the u-by-v-region in CALCPLOT3D by tracing the movable point with your mouse to verify the observations above.

Remark. In an analogous way, preparing and filling the grid H_{uv}, one gets the value distribution of the mean curvature H:

$$
H_{uv} =
\begin{bmatrix}
0 & -2 & -1 & 0 & 1 & 2 \\
-2 & 0.358705 & 0.358705 & 0.358705 & 0.358705 & 0.358705 \\
-1 & 0.740741 & 0.740741 & 0.740741 & 0.740741 & 0.740741 \\
0 & 4 & 4 & 4 & 4 & 4 \\
1 & 0.740741 & 0.740741 & 0.740741 & 0.740741 & 0.740741 \\
2 & 0.358705 & 0.358705 & 0.358705 & 0.358705 & 0.358705
\end{bmatrix}
$$

d: We construct the tableau of the values K_p for 5 different user input points p using a `zero(3,6)` matrix container, where the first two columns are filled with the headings.

```
# ...
"function table for K with free choosen points"
Kuv = zero(3,6)
Kuv[1] = ("u:"       ,1      ,3.1                     ,-1       ,0         ,5)
Kuv[2] = ("v:"       ,2      ,4                       ,5        ,0         ,5)
Kuv[3] = ("Kp(u,v):",Kp(1,2), Kp(3.1,4), float(Kp(-1,5)), Kp(0,0), Kp(5,5))
Kuv
```

EIGENMATH output:

function table for K with free choosen points

$$
K_{uv} = \begin{bmatrix}
\text{u:} & 1 & 3.1 & -1 & 0 & 5 \\
\text{v:} & 2 & 4 & 5 & 0 & 5 \\
\text{Kp(u,v):} & \frac{4}{81} & 0.00065949 & 0.0493827 & 4 & \frac{4}{40401}
\end{bmatrix}
$$
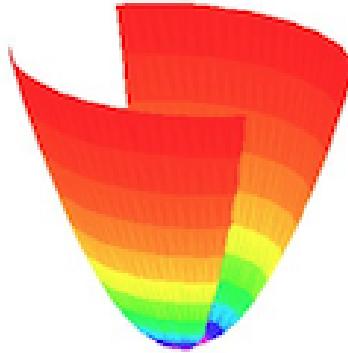
Figure 11: *K* max. Distribution of the values of Gauss curvature K using a color code. Equal color means equal K. Blue color means higher value of K, red smaller values.
The plot was done with MAPLE V.4 using RECKZIEGEL's `diffgeo` packet.

● ▷ Click here to RUN this script.

In summa: we are now able to calculate the GAUSS K and mean curvature H at individual point on the surface, for a sequence (1D vector) of points or a full 2D tableau (matrix) of points to observe the distribution of the K resp. H values.

Exercise 84. Solve the previous example using the parametrization (6.9).
For your exploration choose $a = b = c = 1$.
a. Verify the

Coefficients of	*metric tensor g*	*2nd fundamental tensor h*
	$E = 1 + \frac{1}{4u}$	$M = \frac{1}{2u\sqrt{1+4u}}$
	$F = 0$	N=0
	$G = u$	$N = \frac{2u}{\sqrt{1+4u}}$

b. Verify: $K = \frac{4}{(1+4u)^2}$

6.3 Exercises

● You may cross-check your results of the following exercises in [17].

Exercise 85. (the GAUSS, mean and principal curvature of the saddle $z = xy$)
A parametrization for this function surface S is $\phi\colon U \to \mathbb{R}^3$ with $(u,v) \mapsto (u, v, u \cdot v)$.

a. Use EIGENMATH to calculate the GAUSSian and mean curvature for the saddle.
b. Calculate the principal curvatures and the corresponding principal curvature directions.
c. Calculate the GAUSSian and mean curvature at the point $p = (1, 1)$.

Exercise 86. (GAUSS, mean and principal curvature of the hyperbolic paraboloid)
The *hyperbolic paraboloid* is the 2D parametric surface defined by $\phi(u,v) = (u, v, u^2 - v^2)$.
Recall: $g = \begin{bmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{bmatrix} = \begin{bmatrix} E & F \\ F & G \end{bmatrix} = \begin{bmatrix} 1+4u^2 & -4uv \\ -4uv & 1+4v^2 \end{bmatrix}$
$h = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} = \begin{bmatrix} L & M \\ M & N \end{bmatrix} = \begin{bmatrix} 2(4u^2+4v^2+1)^{-1/2} & 0 \\ 0 & -2(4u^2+4v^2+1)^{-1/2} \end{bmatrix}$.
a. Use EIGENMATH to calculate the Gaussian and mean curvature functions.
b. Calculate GAUSSian and mean curvature at the points $p_i = (i, 5/i), i \in \{1, 2, 3, 4, 5\}$.
c. Calculate the principal curvatures and the corresponding principal curvature directions.

Exercise 87. (GAUSS, mean and principal curvature of the *?* surface)
The *? surface* is the parametric surface defined by $\phi(u,v) = (u + v, u - v, uv)$.
a. Determine the 1st and 2nd fundamental tensors g and h.
b. Verify: at $p = (1, 1)$ we have $K = -1/16$ and $H = 1/16\sqrt{2}$.
c. Calculate GAUSSian and mean curvature at the points $p_i = (i, i^2), i \in \{-2, -1, 0, 1, 2\}$.
d. Calculate the principal curvatures and the corresponding principal curvature directions.

Exercise 88. (GAUSS, mean and principal curvature of torus)
The *torus* is the parametric surface defined by – $R = r_o$ in the figure –

$$\phi(u,v) = ((R + r\cos(v))\cos(u), (R + r\cos(v))\sin(u), r\sin(v))$$

with patch region $(u,v) \in \mathbb{R}^2$.
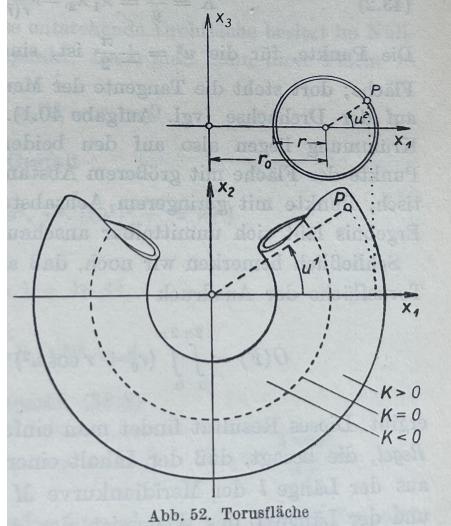
Abb. 52. Torusfläche   KREYSZIG [15, p.165]

Recall: the metric tensor g has coefficients $E = (R + r\cos(v))^2, F = 0, G = r^2$.
the 2nd fundamental tensor h has $L = -(R + r\cos(v))\cos(v), M = 0, N = -r$.
a. Verify: $K = \frac{\cos(v)}{r(R + r\cos(v))}$.
b. Calculate $K(p_i)$ and $H(p_i)$ at the points $p_i = (i, i^2), i \in \{-2, -1, 0, 1, 2\}$.
c. Calculate the principal curvatures and the corresponding principal curvature directions.
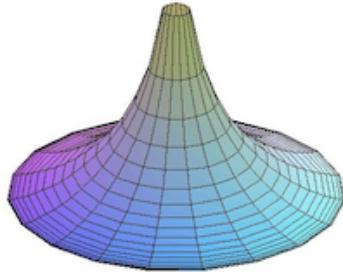
Exercise 89. (GAUSS, mean and principal curvature of BELTRAMI's pseudosphere)
BELTRAMI*'s pseudosphere* is the parametric surface defined by

$$\phi(u,v) = (\sin u \cos v, \sin u \sin v, \cos u + \ln(\sin u(\cos u + 1)))$$
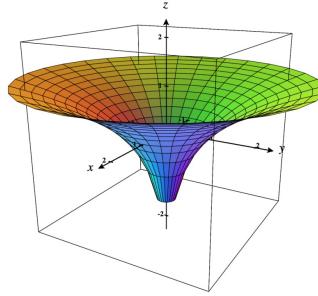
with patch region $(u,v) \in \mathbb{R}^+$.

a. Determine the 1st and 2nd fundamental tensors (g_{ij}) resp. (h_{ij}).
b. Calculate $K(p_i)$ and $H(p_i)$ at the points $p_i = (i, i^2), i \in \{-2, -1, 0, 1, 2\}$.
c. Calculate the principal curvatures and the corresponding principal curvature directions.

Exercise 90. (GAUSS, mean and principal curvature of a funnel, cf. [2, p.156])
The *funnel* is the parametric surface defined by $\phi(u,v) = (v\cos(u), v\sin(u), \ln(v))$, with
patch region $(u,v) \in [0, 2\pi] \times \mathbb{R}^2$.

a. Plot the funnel using CALCPLOT3D.
b. Determine the 1st and 2nd fundamental tensors g resp. h.
   Verify: $g = \begin{bmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{bmatrix} = \begin{bmatrix} v^2 & 0 \\ 0 & 1+1/v^2 \end{bmatrix}$.
c. Calculate $K(p_i)$ and $H(p_i)$ at the points $p_i = (i, i^2), i \in \{-2, -1, 0, 1, 2\}$.
d. Calculate the principal curvatures κ_i.

• You may cross-check your results of the following exercises in [17].
• *Calculate the* GAUSS *curvature* K, *the mean curvature* H *and principal curvatures* κ_i *for the following 2D surfaces using* EIGENMATH ...

Exercise 91. the elliptic cylinder $\phi(u,v) = (a\cos(u), b\sin(u), v)$ over $[0, 2\pi[\times \mathbb{R}$.
Recall: g has the components $E = a^2 \sin^2 u + b^2 \cos^2 u, F = 0, G = 1$.
   h has the components $L = \dfrac{-ab}{\sqrt{a^2 \sin^2 u + b^2 \cos^2 u}}, M = 0, N = 0$.
a. Verify: $K = 0$ and $H = \dfrac{-ab}{(a^2 \sin^2 u + b^2 \cos^2 u)^{3/2}}$.
b. Verify: $\kappa_1 = \dfrac{-ab}{(a^2 \sin^2 u + b^2 \cos^2 u)^{3/2}}$ and $\kappa_2 = 0$.

Exercise 92. the cone $\phi(u,v) = (u\cos(v), u\sin(v), u)$ with $u > 0$.
Recall: g has the components $E = 2, F = 0, G = u^2$.
   h has the components $L = \frac{u}{\sqrt{2}}, M = 0, N = 0$.
a. Verify: $K = 0$ and $H = \frac{\sqrt{2}}{4u}$.
b. Verify: $\kappa_1 = 0$ and $\kappa_2 = 2H$.

Exercise 93. the monkey saddle $\phi(u,v) = (u, v, u^3 - 3uv^2)$.
Recall: g has the components $E = 1 + 9(u^2 - v^2)^2, F = 18uv(v^2 - u^2), G = 1 + 36u^2v^2$.
   Define for the moment: $a := 1 + 9(u^2 + v^2)^2$. Then
   h has the components $L = \frac{6u}{\sqrt{a}}, M = \frac{-6v}{\sqrt{a}}, N = \frac{-6u}{\sqrt{a}}$.
Verify: $K = \frac{-36(u^2+v^2)}{a^2}$ and $H = \frac{27u(-u^4+2u^2v^2+3v^4)}{a3/2}$.

Exercise 94. the unit sphere $\phi(u,v) = (1\cos(u)\sin(v), 1\sin(u)\sin(v), 1\cos(v))$.

Exercise 95. the catenoid $\phi(u,v) = (c\cosh\frac{v}{c}\cos u, c\cosh\frac{v}{c}\sin u, v)$ with $c = 1$.

Exercise 96. the function graph $\phi(u,v) = (u,v,f(u,v))$.

Exercise 97. the surface of revolution $\phi(u,v) = (f(u)\cos(v), f(u)\sin(v), g(u))$.

Exercise 98. the PLÜCKER conoid $\phi(u,v) = (v\cdot\cos(u), v\cdot\sin(u), 2\cos(u)\cdot\sin(u))$.

Exercise 99. the SCHERK surface $\phi(u,v) = (u,v,\ln(\frac{\cos v}{\cos u}))$.

Exercise 100. the ENNEPER $\phi(u,v) = (u - \frac{1}{3}u^3 + uv^2, v - \frac{1}{3}v^2 + vu^2, u^2 - v^2)$.

Exercise 101. the hexenhut $\phi(u,v) = (\alpha/\sqrt{u}\cdot\cos v, \alpha/\sqrt{u}\cdot\sin v, u)$ where $\alpha^2 = \frac{2}{3\sqrt{3}}$.
Recall: g has the components $E = 1 + 9(u^2 - v^2)^2, F = 18uv(v^2 - u^2), G = 1 + 36u^2v^2$.
    Define for the moment: $a := 1 + 9(u^2 + v^2)^2$. Then
    h has the components $L = \frac{6u}{\sqrt{a}}, M = \frac{-6v}{\sqrt{a}}, N = \frac{-6u}{\sqrt{a}}$.

Exercise 102. the BIANCHI surface

$$\phi(u,v) = \left(\frac{2\cdot\sqrt{v^2+1}\cdot\sin(u)\cdot\cos(-v+\arctan(v))}{1+v^2\sin(u)^2}, \frac{-2\cdot\sqrt{v^2+1}\cdot\sin(u)\cdot\sin(-v+\arctan(v))}{1+v^2\sin(u)^2}, \ln(\tan(\frac{1}{2u})) + \frac{2\cdot\cos(u)}{1+v^2\sin(u)^2}\right)$$

7 CHRISTOFFEL **Symbols and** RIEMANN **Tensors**
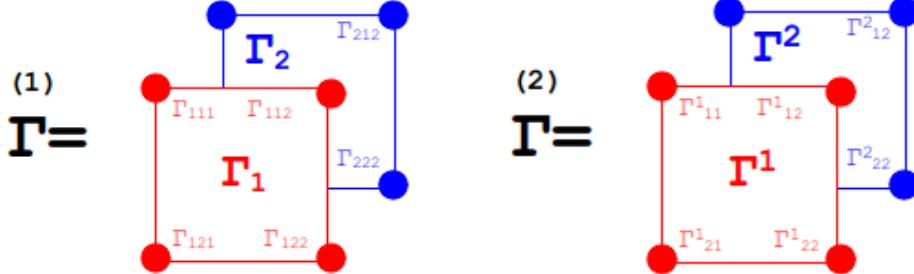
Figure 12: How to imagine the CHRISTOFFEL symbol of first kind $\Gamma^{(1)}$ and the CHRISTOFFEL symbol of second kind $\Gamma^{(2)}$ as $2^3 = 8$ numbers arranged in a tensorial manner in 2 layers. This structure is respected by the output of EIGENMATH.

The CHRISTOFFEL symbol plays a prominent role in Differential Geometry and General Relativity, where we define the RIEMANN curvature tensor $R^a{}_{bcd}$, the RICCI tensor R_{bd} and the EINSTEIN curvature scalar R with the help of the CHRISTOFFEL symbols of 1st and 2nd kind. We will need the RIEMANN curvature tensor $R^a{}_{bcd}$ to formulate the famous GAUSSian *theorema egregium*.

7.1 Definition

Let $\phi \colon U \to \mathbb{R}^3$ be a parametrization of a 2-dimensional parametric surface S.
For the purpose of this definition we need to heavily index the parameters in a tensorial manner and therefore we use the notation $(u^1, u^2) \in U$ instead of (u, v) as before.
Let $g = (g_{ij})$ be the surface metric tensor of ϕ with inverse matrix $g^{-1} = (g^{ij})$.

a. The CHRISTOFFEL *symbol of 1st kind, denoted* $\Gamma_{ij,k} \colon U \to \mathbb{R}$ is defined by

$$\Gamma_{ij,k} := \frac{\partial^2 \phi}{\partial u_i \partial u_j} \bullet \frac{\partial \phi}{\partial u_k} \equiv \phi_{u^i u^j} \bullet \phi_{u^k} \tag{7.1}$$

where $i, j, k \in \{1, 2\}$[7].
Using the fundamental metric tensor g we may write equation (7.1) in equivalent form[8]

$$\Gamma_{ij,k} := \frac{1}{2} \cdot \left(g_{jk,i} + g_{ik,j} - g_{ij,k} \right) \tag{7.2}$$

where $g_{ij,k} \equiv \frac{\partial g_{ij}}{\partial u^k}$ denotes the partial derivative of g_{ij} with respect to the coordinate u^k.

---

[7]cf. COXETER [5, p.439], PRESSLEY [23, p.241], SOCHI [31, p.34]
[8]cf. COXETER [5, p.439], RECKZIEGEL [24, p.146] or SOCHI [31, p.34]

b. The CHRISTOFFEL *symbol of 2nd kind, denoted* $\Gamma^k{}_{ij} \colon U \to \mathbb{R}$ is defined by using the inverse metric tensor g^{ij} and memorizing the EINSTEIN summation convention[9] as

$$\Gamma^k_{ij} := \frac{1}{2} \cdot g^{km} \cdot (g_{mi,j} + g_{mj,i} - g_{ij,m}) \tag{7.3}$$

• Decoding the EINSTEIN summation convention and to prepare for the implementation of these formulas in EIGENMATH we write equation (7.3) explicit as

$$\Gamma^k_{ij} = \frac{1}{2} \cdot \sum_{m=1}^{2} g^{km} \cdot (g_{mi,j} + g_{mj,i} - g_{ij,m}) \tag{7.4}$$

Now we are able to define the RIEMANN curvature tensors of 1st and 2nd kind. We denote the partial derivative with respect to the coordinate u^j in short by ∂_i, because we then may write the formulas for R in a better memorizable shape, cf. [22, p.91]:

c. The RIEMANN *curvature tensor of 1st kind, denoted* $R_{ijkl} \colon U \to \mathbb{R}$ is defined by

$$R_{ijkl} := \partial_k \Gamma_{jli} - \partial_l \Gamma_{jki} + \Gamma_{ilr} \cdot \Gamma^r_{jk} - \Gamma_{ikr} \cdot \Gamma^r_{jl} \tag{7.5}$$

and the RIEMANN *curvature tensor of 2nd kind, denoted* R^i_{jkl} is defined by

$$R^i_{jkl} := \partial_k \Gamma^i_{jl} - \partial_l \Gamma^i_{jk} + \Gamma^r_{jl} \cdot \Gamma^i_{rk} - \Gamma^r_{jk} \cdot \Gamma^i_{rl} \tag{7.6}$$

Remark.
1. We remark that the latin indices i, j, k, l used in the previous equations do not imply that we are in 3D ambient space, but are meant here to denote the indexed parameters of the parametric surface following a usual convention, cf. chapter *1*.
2. The CHRISTOFFEL symbol of 2nd kind may also be defined by means of decoding with the EINSTEIN summation convention

$$\Gamma_{ijk} := g_{li} \cdot \Gamma^l{}_{jk} \tag{7.7}$$

This last definition is an example of lowering the index l by multiplying with the metric tensor g ('index juggling').
3. We have

LEXICON	*Math*	EIGENMATH
metric tensor g:	g_{ij}	g or gdd
inverse of metric tensor g^{-1}:	g^{ij}	ginv or guu or gu
CHRISTOFFEL *symbol of 1st kind*:	Γ_{ijk}	Gammaddd or GammaI
CHRISTOFFEL *symbol of 2nd kind*:	$\Gamma^k{}_{ij}$	Gammaudd or GammaII
RIEMANN *curvature tensor of 1st kind*:	R_{ijkl}	Rdddd or RI
RIEMANN *curvature tensor of 2nd kind*:	$R^i{}_{jkl}$	Ruddd or RII

Remember: in EIGENMATH we encode the tensorial notation 'index up (upper index)' as 'u' and 'index down (lower index)' as 'd', therefore in EIGENMATH's output window Γ_{ij}^k is printed as Γ_{udd}. But we often prefer also the notation Γ^{II}.

4. The RIEMANN curvature tensor R of type $(4,0)$ resp. $(3,1)$ has $2^4 = 16$ coefficients arranged in 4 2-by-2 matrices, analogue to figure 12.

5. As for equations (7.5) and (7.6) see for example KREYSZIG [15, p.176] , LIPSCHUTZ [19, p.214 ff; 224] , STEEB [34, p.62] or THORPE [36, p.230 ff].

6. Because all objects $\Gamma_{ijk}, \Gamma_{ij}^k, R_{ijkl}, R_{jkl}^i$ are constructed using only the coefficients of metric tensor g_{ij}, these objects belongs to the so-called *intrinsic geometry* of surface S.

$\heartsuit$

Implementing formulas (7.1) ... (7.6) in EIGENMATH and calculating some examples should enhance a clear understanding of these mathematical constructs. A CAS like EIGENMATH will relieve us of the tedious calculations, which are involved in the terms for the CHRISTOFFEL symbols and the RIEMANN curvature tensors.

$\heartsuit$

7.2   Example - Γ and R^i_{jkl} for the surface $z = xy$

We now demonstrate two implementations of the CHRISTOFFEL symbols and the RIEMANN curvature tensors: one closely oriented on formulae **a**. resp. **b**. and its term structure. The first implementation will become slowly more abstract, starting with simple functions to reflect the structure of the definition of the concepts and then using EIGENMATHs repeat control structure `for` to make the CAS representation more comfortable. The other implementation uses clever compact tensor techniques such as `contract` and index juggling and was given by G. WEIGT in [42].

7.2.1   *Calculation of Γ and R using functions*

We show first the implementation and comment it afterwards:

```
# EIGENMATH
-- HYPERBOLIC PARABOLOID
phi(u,v) = (u , v, u*v)                   --(1)
--          x    y   z

U = (u,v)                                 --(2)

g(i,j) = dot(d(phi, U[i]), d(phi,U[j]))   --(3)
g(1,1)
eval(g(1,1), u1,2, u2,3)                  --(4)
```

```
g = ((g(1,1),g(1,2)),                              --(5)
     (g(2,1),g(2,2)) )
g

"CHRISTOFFEL symbol Gijk of 1st kind without using g, see (7.1)"

Gamma1(i,j,k) = dot( d(phi, U[i], U[j]), d(phi, U[k]) )   --(6)

Gamma1(1,2,2)                                      --(7)
Gamma1(2,2,2)

-- index pattern for 8 entries
Gamma1 = ( ((Gamma1(1,1,1), Gamma1(1,1,2)),    --(8) 1st matrix
            (Gamma1(1,2,1), Gamma1(1,2,2))),
           ((Gamma1(2,1,1), Gamma1(2,1,2)),    --(9) 2nd matrix
            (Gamma1(2,2,1), Gamma1(2,2,2))) )
Gamma1                                             --(10)


"CHRISTOFFEL symbol Gijk of 1st kind with g, see (7.2)"

GammaI(i,j,k) = 1/2 * ( d(g[j,k],U[i]) +       --(11)
                        d(g[i,k],U[j]) -
                        d(g[i,j],U[k]) )

GammaI(1,2,2)                                      --(12)
GammaI(2,2,1)

GammaI = ( ((GammaI(1,1,1),GammaI(1,1,2)),     --(13)
            (GammaI(1,2,1),GammaI(1,2,2))),
           ((GammaI(2,1,1),GammaI(2,1,2)),
            (GammaI(2,2,1),GammaI(2,2,2))) )

GammaI
```

● ▷ Click here to RUN this script.

Comment. We define the surface via the parameterization in (1) and choose $U = (u, v)$ as coordinates, where the access $U[i] \equiv u_i$ in (3) reflects the notion in the text. Equation (3) is the translation of the dot product of the tangential basis vectors ϕ_i to EIGENMATH. (4) evaluates the metric coordinate g_{11} at the point $p = (u, v) = (2, 3)$ of U. In (5) the whole metric is saved in a 2-by-2 matrix. Equation (6) use (7.1) to define the CHRISTOFFEL symbol of 1st kind once again, denoted `Gamma1`, *without making use of the metric*. We test this in (7) to calculate the value Γ_{122}. The whole CHRISTOFFEL symbol is then saved in two matrices in (8) resp. (9), respecting the structure of figure 12. Equation (11) use

(7.2) to give an alternative definition of the CHRISTOFFEL symbol of 1st kind once again, denoted now `GammaI`, *with making use of the metric*! We test this in (12) to calculate the value `GammaI(1,2,2)`$= \Gamma_{122}$. The whole CHRISTOFFEL symbol is then saved in two matrices in (13).

EIGENMATH output:

CHRISTOFFEL symbol Gijk of 1st kind (without g)

u

0

$$\Gamma_1 = \begin{bmatrix} \begin{bmatrix} 0 & 0 \\ v & u \end{bmatrix} \\ \begin{bmatrix} v & u \\ 0 & 0 \end{bmatrix} \end{bmatrix}$$

• We now implement the CHRISTOFFEL symbol of 2nd kind. Equation (15) use (7.4): the components are calculated using both the metric g and the 'contravariant' metric tensor gu a.k.a. the inverse of the metric tensor in (14). All values are collected in tensor Γ^{II}, see (16).

```
"CHRISTOFFEL symbol Gi^jk of 2nd kind"

gu = inv(g)      -- .u means up i.e. inverse of metric  (14)
gu

-- GammaII as function GammaII()
--                                      index pattern   (+)
--     gu[k,m]*( d(g[m,i],u[j]) +       ij
--               d(g[m,j],u[i]) -       ji
--               d(g[i,j],u[m])) )      ij

GammaII(k,i,j) = 1/2* sum(m,1,2,                         --(15)
                gu[k,m]*( d(g[m,i],U[j]) +
                          d(g[m,j],U[i]) - d(g[i,j],U[m]) ))

GammaII(2,2,2)
GammaII(2,2,2)

GammaII = ( ((GammaII(1,1,1),GammaII(1,1,2)),            --(16)
             (GammaII(1,2,1),GammaII(1,2,2))),
            ((GammaII(2,1,1),GammaII(2,1,2)),
             (GammaII(2,2,1),GammaII(2,2,2))) )

GammaII
```

● ▷ Click here to RUN this script.

EIGENMATH output:

$$\Gamma_{II} = \begin{bmatrix} \begin{bmatrix} 0 & \dfrac{v}{u^2 + v^2 + 1} \\ \dfrac{v}{u^2 + v^2 + 1} & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & \dfrac{u}{u^2 + v^2 + 1} \\ \dfrac{u}{u^2 + v^2 + 1} & 0 \end{bmatrix} \end{bmatrix}$$

● As for the RIEMANN curvature tensors of 1st and 2nd kind, denoted RI(i,j,k,l) resp. RII(i,j,k,l) in the EIGENMATH code, the implementation in equations (17) and (18) is straight forward; recall, that GammaI resp. GammaII is now a tensor (multi-matrix) via (16) with [..]-access:

```
"RIEMANN curature tensor Rijkl of 1st kind as function"
RI(b,m,s,q) = d(GammaI[b,m,q],u[s]) -                    --(17)
              d(GammaI[b,m,s],u[q]) +
                  sum(n,1,2, GammaI[b,n,s]*GammaII[n,m,q] ) -
                  sum(n,1,2, GammaI[b,n,q]*GammaII[n,m,s] )

RI1212 = RI(1,2,1,2)
RI1212

"RIEMANN curature tensor Ri^jkl of 2nd kind as function"
RII(b,m,s,q) =  d(GammaII[b,m,q],u[s]) -            --(18)
                d(GammaII[b,m,s],u[q]) +
                    sum(n,1,2, GammaII[b,n,s]*GammaII[n,m,q] ) -
                    sum(n,1,2, GammaII[b,n,q]*GammaII[n,m,s] )

RII1212 = RII(1,2,1,2)
RII1212
```

● ▷ Click here to RUN this script.

EIGENMATH output:

RIEMANN curature tensor Rijkl of 1st kind as function

$$R_{I1212} = -\frac{u^2}{u^2 + v^2 + 1} + 1$$

RIEMANN curature tensor Ri^jkl of 2nd kind as function

$$R_{II1212} = \frac{2\,u\,v}{\left[u^2 + v^2 + 1\right]^2} - \frac{v^2}{\left[u^2 + v^2 + 1\right]^2}$$

7.2.2 *Calculation of* Γ *and* R *without tensor techniques*

• We calculate the CHRISTOFFEL symbols in equations (1) resp. (2) using `for` loops to fill the containers `GammaI` und `GammaII` automatically with the corresponding entries.

```
# EIGENMATH
-- hyperbolic PARABOLOID, version 2
phi(u,v) = (u , v, u*v)

U        = (u,v)
gij(i,j) = dot(d(phi,U[i]), d(phi,U[j]))

"metric tensor g"
g = zero(2,2)
    for(i,1,2, for( j,1,2, g[i,j] = gij(i,j)))
g

gu = inv(g)
gu

"------------------------------------------- Gamma I"
GammaI = zero(2,2,2)                          -- (1) Gijk
for( i,1,2,
    for( j,1,2,
        for( k,1,2,
            GammaI[i,j,k] = 1/2 *
                            ( d(g[j,k],U[i]) +
                              d(g[i,k],U[j]) -
                              d(g[i,j],U[k]) ))))

GammaI
GammaI[2,2,2]              -- test, ok

"------------------------------------------- Gamma II"
GammaII=zero(2,2,2)                           -- (2) Gi^jk
for( k,1,2,
    for( l,1,2,
        for( m,1,2,
            GammaII[k,l,m] =
                sum(i,1,2, gu[k,i]*1/2 *
                        ( d(g[m,i],U[l]) +
                          d(g[l,i],U[m]) -
                          d(g[m,l],U[i]) )
                ))))

GammaII
GammaII[2,2,2]
```

● ▷ Click here to RUN this script.

EIGENMATH output: same as above.

● We calculate the RIEMANN curvature tensors in equations (3) resp. (4) in a completely analogous way.

```
# EIGENMATH
"----------------------------- Riemann I"
RI = zero(2,2,2,2)             -- (3)  Rijkl
for( b,1,2,
    for( m,1,2,
        for( s,1,2,
            for( q,1,2,
            RI[b,m,s,q] =
                d(GammaI[b,m,q],U[s]) -
                d(GammaI[b,m,s],U[q]) +
                sum(n,1,2, GammaI[b,n,s]*GammaII[n,m,q] ) -
                sum(n,1,2, GammaI[b,n,q]*GammaII[n,m,s] )
                ))))

RI
RI[1,2,1,2]              -- test, ok

"----------------------------- Riemann II"
RII = zero(2,2,2,2)            -- (4)  Ri^jkl

for( b,1,2,
    for( m,1,2,
        for( s,1,2,
            for( q,1,2,
            RII[b,m,s,q] =
                d(GammaII[b,m,q],U[s]) -
                d(GammaII[b,m,s],U[q]) +
                sum(n,1,2, GammaI[b,n,s]*GammaII[n,m,q] ) -
                sum(n,1,2, GammaI[b,n,q]*GammaII[n,m,s] )
                ))))
RII
RII[1,2,1,2]
```

● ▷ Click here to RUN this script.

EIGENMATH output:

-------------------------------- Riemann I

$$
R_I = \begin{bmatrix}
\begin{bmatrix} 0 & \dfrac{u\,v}{u^2+v^2+1} \\[2ex] -\dfrac{u\,v}{u^2+v^2+1} & 0 \end{bmatrix} &
\begin{bmatrix} 0 & -\dfrac{u^2}{u^2+v^2+1} \\[2ex] \dfrac{u^2}{u^2+v^2+1} & 0 \end{bmatrix} \\[6ex]
\begin{bmatrix} 0 & \dfrac{v^2}{u^2+v^2+1} \\[2ex] -\dfrac{v^2}{u^2+v^2+1} & 0 \end{bmatrix} &
\begin{bmatrix} 0 & -\dfrac{u\,v}{u^2+v^2+1} \\[2ex] \dfrac{u\,v}{u^2+v^2+1} & 0 \end{bmatrix}
\end{bmatrix}
$$

---------------------------- Riemann II

$$
R_{II} = \begin{bmatrix}
\begin{bmatrix} 0 & \dfrac{u\,v}{u^2+v^2+1} - \dfrac{2\,u\,v}{\left[u^2+v^2+1\right]^2} \\[3ex] -\dfrac{u\,v}{u^2+v^2+1} + \dfrac{2\,u\,v}{\left[u^2+v^2+1\right]^2} & 0 \end{bmatrix} &
\begin{bmatrix} \dfrac{u^2}{u^2+v^2+1} - \cdots \\ \cdots \end{bmatrix} \\[6ex]
\begin{bmatrix} 0 & -\dfrac{2\,u^2}{\left[u^2+v^2+1\right]^2} + \dfrac{v^2}{u^2+v^2+1} + \dfrac{1}{u^2+v^2+1} \\[3ex] \dfrac{2\,u^2}{\left[u^2+v^2+1\right]^2} - \dfrac{v^2}{u^2+v^2+1} - \dfrac{1}{u^2+v^2+1} & 0 \end{bmatrix} &
\begin{bmatrix} \cdots \\ -\dfrac{\cdots}{u} \end{bmatrix}
\end{bmatrix}
$$

Comment. The output of R_{II} exceeds the screen. If you like to see also the not-displayed values use the matrix access to individual entries e.g. `RII[2,2,1,2]`.

Remark. Pay attention: the call of the CHRISTOFFEL symbol or RIEMANN curvature tensor *functions* uses round brackets `(..)`, but the direct access to the elements of the corresponding *tensor matrices* uses square brackets `[..]`.

7.2.3 *Calculation of* Γ *and* R *with tensor techniques*

We now implement both the CHRISTOFFEL symbols and the RIEMANN curvature tensors using compact tensor techniques following G. WEIGT.

```
# EIGENMATH
-- hyperbolic PARABOLOID
phi(u,v) = (u , v, u*v)                          --(1)

U = (u,v)    -- region of phi                    --(2)

gij(i,j) = dot( d(phi,U[i]), d(phi,U[j]))        --(3)

g = zero(2,2)
    for(i,1,2, for( j,1,2, g[i,j] = gij(i,j))) --(4)
g
```

```
gu = inv(g)                                          --(5)
gu

Dg = d(g, U)                                         --(6)
Dg

"CHRISTOFFEL SYMBOL 1st kind"
-- Analyse defining formula:                         --(7)
--
-- Gamma    = 1/2 (g      + g      - g    )    Formula
--      abc          ab,c    ac,b    bc,a      INDEX-PATTERN
--      123          12 3    13 2    23 1      INDEX-PATTERN

GammaI = 1/2 (Dg + transpose(Dg,2,3) - transpose(Dg,2,3,1,2))   --(8)
          -- ok   transpose    c.b     transpose c.a and b.a
GammaI

"CHRISTOFFEL SYMBOL 2nd kind"
-- Analyse defining formula:
--
--      a       au
-- Gamma    = g    . Gamma
--      bc               ubc

GammaII = dot(gu, GammaI)                            --(9)
GammaII
```

Comment. We define the surface via the parameterization in (1) and choose $U = (u, v)$ as coordinates, where the access $U[i] \equiv u_i$ in (2) reflects the notion in the text. Equation (3) is the translation of the dot products of the tangential basis vectors ϕ_i to EIGENMATH. (4) constructs the surface metric tensor $g = (g_{ij})$ and (5) calculates its corresponding inverse matrix, the so-called *contravariant* metric tensor gu. (6) computes all partial derivatives of the metric tensor g w.r.t. all coordinates $U = (u, v)$ and saves the result in Dg. (8) translates equation (7.4) to EIGENMATH and (9) the equation (7.7).

EIGENMATH output: same as above.

● ▷ Click here to RUN this script.

• Now for the implementation of the RIEMANN curvature tensors using compact tensor techniques. The explanation is inline.

```
# EIGENMATH
"RIEMANN CURVATURE tensor Ri^jkl 2nd kind"
-- Analyze formula, G=Gamma
--   a         a          a          a       u        a       u
-- R      = G        - G        + G      * G      - G    * G
--   bcd      bd,c       bc,d       uc       bd       ud      bc
--
-- Index pattern:
-- Ra.bcd = Ga.bd,c - Ga.bc,d  + Ga.uc*Gu.bd - Ga.ud*Gu.bc
--  1 234     1 24 3    1 23 4      T2             T2
--            1 u3      u 24       1  4           23

T1 = d(GammaII, U)                               --(10) 1st temporary result
T2 = dot(transpose(GammaII,2,3), GammaII)  --(11) 2nd temporary result

RII = transpose(T1,3,4) -T1+ transpose(T2,2,3) - transpose(T2,2,3,3,4)
      -- transpose d.c   ok     transpose c.b    transp. d.b then d.c
RII

"RIEMANN CURVATURE tensor Rijkl 1st kind"
-- with lowering the index (index juggling) via (7.7)

RI = dot(g, RII)                                 --(11)
RI
```

• ▷ Click here to RUN this script.

EIGENMATH output: same as above.

Remark. 1. Both the CHRISTOFFEL symbols and the RIEMANN curvature tensors are implemented using only the surface metric tensor g. Hence both objects belongs to the so-called *intrinsic geometry* of the surface $S = \phi$.

2. The CHRISTOFFEL symbol of 2nd kind plays a big role in the construction of the so-called LEVI-CIVITA *connection* ∇ alias the *covariant derivation* $\nabla_X Y$ of a vector field Y w.r.t. X, cf. OLOFF [22, p. 90 ff] or RECKZIEGEL [24, p.147 ff] or THORPE [36, p. 225 ff].

3. The above presented tensorial version of the calculation of CHRISTOFFEL symbol and RIEMANN curvature tensor are most elegant and effective: you does not have to specify the dimension of the surface. So one may calculate the metric and the derived objects in the same way in higher dimensions, which is essential in General Relativity, see e.g.

calculations in the 4D SCHWARZSCHILD space time, cf. OLOFF [22, p. 96 ff] or WEIGT [42]. Using the `for`-loop constructs - which are perhaps more simple and insightful - you must manually specify the considered dimension, i.e.

```
n=3
for( i,1,n, do ...
```

♡ This was a long example. But we wanted to present a gentle rise of abstraction in order to fully understand what is going on and to equip the user which a selection of means to be choose from. The following examples and exercises may therefore be realized according to your mood ♡.

7.3   Example - Γ and $R^i{}_{jkl}$ for the Helicoid

Sometimes it is necessary that the user should help EIGENMATH to to deal with complicated output and to simplify especially terms involving trigonometric functions. This is demonstrated in the following

Example 15. (the calculation of Γ and R for the helicoid)
The *helicoid* is the 2D parametric surface defined by $\phi(u,v) = (v\cos(u), v\sin(u), u)$.

a. Use EIGENMATH to calculate the CHRISTOFFEL symbol for the helicoid.
b. Use EIGENMATH to calculate the RIEMANN curvature tensors for the helicoid.
c. Use EIGENMATH to calculate the value of R^{II}_{1212} for the helicoid. Use both the Riemann curvature function `RII(1,2,1,2)` and the RIEMANN curvature tensor `RII[1,2,1,2]`.
What is the value of `RII(1,2,1,2)|(u,v)=(pi,1)`?
Check your results with WOLFRAM.mathworld [48] or Rejbrand Encyclopaedia.

Solution.   To determine the CHRISTOFFEL symbols and the RIEMANN curvature tensor we need the surface metric g. The coefficients of g should be simplified, before the calculation of the tensors are done - otherwise the results would be very long and uncomfortable and could not be crosschecked with the literature.

```
# EIGENMATH

phi(u,v) = (u cos(v), u sin(v), c v)    -- HELICOID
U =(u,v)   -- coordinates of surface

g = zero(2,2)
    g[1,1] = dot(d(phi,u), d(phi,u))
    g[1,1]    -- cu^2+su^2=1
    g[1,1] = 1

    g[1,2] = dot(d(phi,u), d(phi,v))
    g[1,2]
```

```
        g[2,1] = g[1,2]
        g[2,1]

        g[2,2] = dot(d(phi,v), d(phi,v))
        g[2,2]    -- (c^2+u^2)*(cos(v)^2+sin(v)^2)
        g[2,2] = c^2 + u^2
    g

    gu = inv(g)
    gu

    Dg = d(g, U)
    Dg

    "CHRISTOFFEL SYMBOL 1st kind"
    GammaI = 1/2 (Dg + transpose(Dg,2,3) - transpose(Dg,2,3,1,2))
    GammaI

    "CHRISTOFFEL SYMBOL 2nd kind"
    GammaII = dot(gu, GammaI)
    GammaII

    "RIEMANN CURVATURE tensor Ri^jkl 2nd kind"
    T1 = d(GammaII, U)
    T2 = dot(transpose(GammaII,2,3), GammaII)

    RII = transpose(T1,3,4)- T1 + transpose(T2,2,3) - transpose(T2,2,3,3,4)
    RII

    "RIEMANN CURVATURE tensor Rijkl 1st kind"
    RI = dot(g, RII)
    RI

    -- c)
    RiemII(b,m,s,q) =  d(GammaII[b,m,q],u[s]) -
                       d(GammaII[b,m,s],u[q]) +
                       sum(n,1,2, GammaI[b,n,s]*GammaII[n,m,q] ) -
                       sum(n,1,2, GammaI[b,n,q]*GammaII[n,m,s] )

    RiemII(1,2,1,2)
    eval( RiemII(1,2,1,2), u,pi, v,1)
```

Comment. To simplify the returned expressions for g[1,1] and g[2,2] we look at the associated output and set the new value into the metric tensor by means of an assignment g[1,1]=1 resp. g[2,2]=$c^2 + u^2$.

● ▷ Click here to RUN this script.

Eigenmath output:

CHRISTOFFEL SYMBOL 2nd kind

$$\Gamma_{II} = \begin{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & -u \end{bmatrix} \\ \begin{bmatrix} 0 & \dfrac{u}{c^2+u^2} \\ \dfrac{u}{c^2+u^2} & 0 \end{bmatrix} \end{bmatrix}$$

RIEMANN CURVATURE tensor Ri^jkl 2nd kind

$$R_{II} = \begin{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 \\ -\dfrac{u^2}{c^2+u^2} \end{bmatrix} \\ \begin{bmatrix} 0 & -\dfrac{u^2}{\left[c^2+u^2\right]^2} + \dfrac{1}{c^2+u^2} \\ \dfrac{u^2}{\left[c^2+u^2\right]^2} - \dfrac{1}{c^2+u^2} & 0 \end{bmatrix} \end{bmatrix}$$

Comment. On the LHS we display `GammaII=` $\Gamma^{k}{}_{lm}$.

On the RHS we see part of `RII=` $R^{i}{}_{jkl}$.

The tensor `RI=` R_{ijkl} ist displayed in the 1st line in the last output-part and the special value of `RiemII(1,2,1,2)` is calculated via a function call to the function `RiemII` - without the necessity to calculate all 16 entries of `RII`:

RIEMANN CURVATURE tensor Rijkl 1st kind

$$R_{I} = \begin{bmatrix} \begin{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & \dfrac{u^2}{c^2+u^2} - 1 \\ -\dfrac{u^2}{c^2+u^2} + 1 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & -\dfrac{u^2}{c^2+u^2} + 1 \\ \dfrac{u^2}{c^2+u^2} - 1 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

$$\dfrac{u^2}{c^2+u^2} - 1$$

$$\dfrac{\pi^2}{c^2+\pi^2} - 1$$

7.4   Exercises

Let's calculate the CHRISTOFFEL symbols and the RIEMANN curvature tensors for our small collection of classical 2D surfaces. Sometimes you should help EIGENMATH to simplify trigonometric expressions or to simplify fractions.

• You may cross-check your results of the calculation of the CHRISTOFFEL symbols of the following exercises in [17]. For R^I and R^{II} you have to use EIGENMATH.

• Calculate the (1) the CHRISTOFFEL symbols Γ^I and Γ^{II}

(2) the curvature tensor R^I and

(3) the curvature tensor R^{II} and

(4) the value RII(1,2,1,2) at $(u,v) = (\pi, 1)$

for the following surfaces using EIGENMATH ...

Exercise 103. ... the elliptic cylinder $\phi(u,v) = (a\cos(u), b\sin(u), v)$ over $[0, 2\pi[\times \mathbb{R}$.
Recall: g has the components $E = a^2 \sin^2 u + b^2 \cos^2 u, F = 0, G = 1$.
Verify: $\Gamma^1_{11} = \frac{(a^2 - b^2)\sin(2u)}{2a^2 \sin^2 u + 2b^2 \cos^2 u}$ and all other coefficients are zero.

Exercise 104. the cone $\phi(u,v) = (u\cos(v), u\sin(v), u)$ with $u > 0$.
Recall: g has the components $E = 2, F = 0, G = u^2$.
Verify: $\Gamma^1 \begin{bmatrix} 0 & 0 \\ 0 & -u/2 \end{bmatrix}$ and $\Gamma^2 = \begin{bmatrix} 0 & 1/u \\ 1/u & 0 \end{bmatrix}$.
See [2, p.232].

Exercise 105. the elliptic paraboloid $\phi(u,v) = (u, v, u^2 + v^2)$.

Exercise 106. the monkey saddle $\phi(u,v) = (u, v, u^3 - 3uv^2)$.
Recall: g has the components $E = 1 + 9(u^2 - v^2), F = 18uv(u^2 - v^2), G = 1 + 36u^2v^2$.

Exercise 107. the unit sphere $\phi(u,v) = (1\cos(u)\sin(v), 1\sin(u)\sin(v), 1\cos(v))$.
See ▷ Cox: R.

Exercise 108. the catenoid $\phi(u,v) = (c\cosh\frac{v}{c}\cos u, c\cosh\frac{v}{c}\sin u, v)$. Choose $c = 1$.

Exercise 109. the torus $\phi(u,v) = ((b + a\cos(v))\cos(u), (b + a\cos(v))\sin(u), a\sin(v))$.
See [15, p.180, p.383].

Exercise 110. the function graph $\phi(u,v) = (u, v, f(u,v))$.
See [15, p.180, p.383].

Exercise 111. the surface of revolution $\phi(u,v) = (f(u)\cos(v), f(u)\sin(v), g(u))$.

Exercise 112. the PLÜCKER conoid $\phi(u,v) = (v \cdot \cos(u), v \cdot \sin(u), 2\cos(u) \cdot \sin(u))$.

Exercise 113. the SCHERK surface $\phi(u,v) = (u, v, \ln(\frac{\cos v}{\cos u}))$.
Recall: g has coefficients $E = \sec^2 u, F = -\tan u \tan v, G = \sec^2 v$, where $\sec = \frac{1}{\cos}$.
Verify:

$$\Gamma^1 = \begin{bmatrix} \tan u & 0 \\ 0 & -\sin u \cos u \sec^2 v \end{bmatrix}$$

$$\Gamma^2 = \begin{bmatrix} -\sin v \cos v \sec^2 u & 1 \\ 1 & \tan v \end{bmatrix}$$

Exercise 114. the ENNEPER $\phi(u,v) = (u - \frac{1}{3}u^3 + uv^2, v - \frac{1}{3}v^2 + vu^2, u^2 - v^2)$.
Recall: the metric tensor g: $E = G = 1 + 2u^2 + 2v^2 + u^4 + 2u^2v^2 + v^4, F = 0$.
Verify:

$$\Gamma^1 = \begin{bmatrix} \frac{2u}{u^2+v^2+1} & \frac{2v}{u^2+v^2+1} \\ \frac{2v}{u^2+v^2+1} & \frac{-2u}{u^2+v^2+1} \end{bmatrix}$$

Exercise 115. the BELTRAMI pseudosphere
$$\phi(u,v) = (\sin u \cos v, \sin u \sin v, \cos u + \ln(\sin u(\cos u + 1)))\ .$$
Recall: g has coefficients $E = \frac{-\cos^2 u}{\cos^2 u - 1}, F = 0, G = 1 - \cos^2 u$.
Verify:

$$\Gamma^1 = \begin{bmatrix} \frac{\sin u}{-\cos u + \cos^3 u} & 0 \\ 0 & \frac{\sin u(\cos^u - 1)}{\cos u} \end{bmatrix}$$

Exercise 116. the hexenhut $\phi(u,v) = (\alpha \cdot \frac{\cos v}{\sqrt{u}}, \alpha \cdot \frac{\sin v}{\sqrt{u}}, u)$ where $\alpha^2 = \frac{2}{3\sqrt{3}}$.
Recall: g has coefficients $E = \frac{1+27u^3}{27u^3}, F = 0, G = \frac{4}{27u}$.
$\qquad h$ has coefficients $(L, M, N) = (-\frac{3}{2u^3 \cdot \sqrt{\frac{1+27u^3}{u^4}}}, 0, \frac{2}{u \cdot \sqrt{\frac{1+27u^3}{u^4}}})$.
Verify:

$$\Gamma^1 = \begin{bmatrix} \frac{-3}{2(1+27u^3)} & 0 \\ 0 & \frac{2}{1+27u^3} \end{bmatrix}$$

$$\Gamma^2 = \begin{bmatrix} 0 & \frac{-1}{2u} \\ \frac{-1}{2u} & 0 \end{bmatrix}$$

See: WHEELER [43, p.22 ff]

Exercise 117. the 'hexe' surface $\phi(u,v) = (\sqrt{u} \cdot \cos v, \sqrt{u} \cdot \sin v, u)$.
Verify:
a. g has coefficients $(E, F, G) = (\frac{1+4u}{4u}, 0, u)$.
b. h has coefficients $(L, M, N) = (\frac{1}{2u \cdot \sqrt{1+4u}}, 0, \frac{2u}{\sqrt{1+4u}})$.
c.

$$\Gamma^1 = \begin{bmatrix} \frac{-1}{2u(1+4u)} & 0 \\ 0 & \frac{-2u}{1+4u} \end{bmatrix}$$

$$\Gamma^2 = \begin{bmatrix} 0 & \frac{1}{2u} \\ \frac{1}{2u} & 0 \end{bmatrix}$$

Exercise 118. the BIANCHI surface

$$\phi(u,v) = \begin{bmatrix} \frac{2 \cdot \sqrt{v^2+1} \cdot \sin(u) \cdot \cos(-v+\arctan(v))}{1+v^2 \sin(u)^2} \\ \frac{-2 \cdot \sqrt{v^2+1} \cdot \sin(u) \cdot \sin(-v+\arctan(v))}{1+v^2 \sin(u)^2} \\ \ln(\tan(\frac{1}{2u})) + \frac{2 \cdot \cos(u)}{1+v^2 \sin(u)^2} \end{bmatrix}$$

Exercise 119. (The GAUSS **Relations for** Γ**)**
The Christoffel symbols are given by the following *Gauss relations* as functions of the coefficients E, F, G of the metric tensor g and their first order derivations. We have the

Fact. *Let* $\phi \colon U \to \mathbb{R}^3$ *be a parametrization of a 2-surface* S.
Let $g = \begin{bmatrix} \phi_u \bullet \phi_u & \phi_v \bullet \phi_u \\ \phi_u \bullet \phi_v & \phi_v \bullet \phi_v \end{bmatrix} \equiv \begin{bmatrix} E & F \\ F & G \end{bmatrix}$ *be the metric tensor of* ϕ.
Then we have

$$\Gamma_{11}^1 = \frac{GE_u - 2FF_u + FE_v}{2(EG - F^2)} \tag{7.8}$$

$$\Gamma_{12}^1 = \frac{GE_v - FG_u}{2(EG - F^2)} \tag{7.9}$$

$$\Gamma_{22}^1 = \frac{2GF_v - GG_u - FG_v}{2(EG - F^2)} \tag{7.10}$$

$$\Gamma_{11}^2 = \frac{2EF_u - EE_v - FE_u}{2(EG - F^2)} \tag{7.11}$$

$$\Gamma_{12}^2 = \frac{EG_u - FE_v}{2(EG - F^2)} \tag{7.12}$$

$$\Gamma_{22}^2 = \frac{EG_v - 2FF_v + FG_u}{2(EG - F^2)} \tag{7.13}$$

a. Program the Gauss relations in EIGENMATH for the helicoid.
Hint: here is a sketch, $\Gamma_{11}^1 \; \|^{Math} = $ `Ga111` $\|^{EigenMath} \ldots$:

```
phi(u,v)= ..
E = d(phi,u)
..
Ga111 = 1/2 (G d(E,u) - 2 F d(F,u) + F d(E,v) / (E G - F^2)
..
Ga222 = 1/2 (E d(G,v) -2 F d(F,v) + F d(G,u)/(EG - F^2 )

-- and if you like to have the tensor GammaII
Ga1 = ((Ga111, Ga112),(Ga121,Ga122))
Ga2 = ..

GammaII = (Ga1, Ga2)
```

Remark. A proof is in HECKMANN [13, p.25].

8   Intrinsic Curvature

The *intrinsic curvature* of a 2D surface can be calculated using only the coefficients of the metric tensor g - without recourse to the shape operator who make use of the *extrinsic* concepts of the sff h.

8.1   Definition

Let U be the parameter region of the metric tensor $g = (g_{ij}) = g(u, v)$.
Let $\partial_u := \frac{\partial}{\partial_u}$ be the partial derivative w.r.t. the variable u and $\partial_v := \frac{\partial}{\partial_v}$ ....
Let $\Gamma^{II} := (\Gamma^1, \Gamma^2)$ be the CHRISTOFFEL symbol of 2nd kind associated with g.
Then we define the intrinsic curvature $K_{intrinsic}$ by

$$K_{intrinsic}(u, v) := \partial_v \Gamma^2_{11} - \partial_u \Gamma^2_{21} + \Gamma^2_{21} \cdot (\Gamma^1_{11} - \Gamma^2_{12}) + \Gamma^2_{11} \cdot (\Gamma^2_{22} - \Gamma^1_{21}) \tag{8.1}$$

Remark. 1. The definition (8.1) is in RECKZIEGEL [24, p. 160].
2. In §9 we show, how the GAUSS curvature K can be calculated using the intrinsic curvature $K_{intrinsic}$.
3. Verify: $K_{intrinsic} = R_{1212}$

8.2   Implementation and Examples

We now give some examples and exercises to calculate the intrinsic curvature $K_{intrinsic}$. We start with the so-called hyperbolic half plane, whose results are clear and short and allows a focus on the intermediate steps.

Example 16. (The intrinsic curvature of the hyperbolic upper half plane H)
Let $H := \{(u, v) \in \mathbb{R}^2 | v > 0\}$ be the *hyperbolic half plane* with 1st fundamental form $ds^2 = \frac{du^2 + dv^2}{v^2}$ resp. its metric tensor $g(u, v) = \begin{bmatrix} u^2 & 0 \\ 0 & v^2 \end{bmatrix}$.
Calculate the the intrinsic curvature $K^H_{intrinsic}(u, v)$ of the hyperbolic half plane H.

Solution   We use EIGENMATH. First we compute the CHRISTOFFEL symbols using known tensor techniques. Then we implement the intrinsic curvature by formula (8.1).

```
# EIGENMATH
"HYPERBOLIC HALF PLANE"
U = (u,v)

g   = ((1/v^2, 0),(0, 1/v^2))
gu = inv(g)
Dg = d(g,U)

GammaI = 1/2 (Dg + transpose(Dg,2,3) - transpose(Dg,2,3,1,2))
GammaI
```

```
GammaII = dot(gu, GammaI)
GammaII

"intrinsic curvature"
Ch111 = GammaII[1,1,1]
Ch121 = GammaII[1,2,1]
Ch211 = GammaII[2,1,1]
Ch221 = GammaII[2,2,1]
Ch222 = GammaII[2,2,2]

Kintrinsic = 1/g[1,1]*( d(Ch211,v) - d(Ch221,u) +
                        Ch221*(Ch111-Ch221)      +
                        Ch211*(Ch222-Ch121)      )

Kintrinsic
```

EIGENMATH output:

$$g = \begin{bmatrix} \frac{1}{v^2} & 0 \\ 0 & \frac{1}{v^2} \end{bmatrix}$$

$$g_u = \begin{bmatrix} v^2 & 0 \\ 0 & v^2 \end{bmatrix}$$

$$D_g = \begin{bmatrix} \begin{bmatrix} 0 & -\frac{2}{v^3} \\ 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \\ 0 & -\frac{2}{v^3} \end{bmatrix} \end{bmatrix}$$

…

CHRISTOFFEL Symbol 1st kind

$$\Gamma_I = \begin{bmatrix} \begin{bmatrix} 0 & -\frac{1}{v^3} \\ -\frac{1}{v^3} & 0 \end{bmatrix} \\ \begin{bmatrix} \frac{1}{v^3} & 0 \\ 0 & -\frac{1}{v^3} \end{bmatrix} \end{bmatrix}$$

CHRISTOFFEL Symbol 2nd kind

$$\Gamma_{II} = \begin{bmatrix} \begin{bmatrix} 0 & -\frac{1}{v} \\ -\frac{1}{v} & 0 \end{bmatrix} \\ \begin{bmatrix} \frac{1}{v} & 0 \\ 0 & -\frac{1}{v} \end{bmatrix} \end{bmatrix}$$

intrinsic curvature

$$K_{intrinsic} = -1$$

● ▷ Click here to RUN this script.

Comment. The result $K^H_{intrinsic}(u, v) \equiv -1$ demonstrates, that this curvature is of constant value, because the result is not dependent of both u and v. Ergo: The hyperbolic half plane has constant intrinsic curvature, cf. BANCHOFF [2, p.227] or RECKZIEGEL [24, p.161].

Exercise 120. In case you don't like tensor techniques: Calculate the the intrinsic curvature $K^H_{intrinsic}(u, v)$ of the hyperbolic half plane H without tensor techniques, by using the iterative computation of the CHRISTOFFEL symbols instead.

Example 17. (The intrinsic curvature of the helicoid)
The metric tensor g of the helicoid is known to have the coefficients $E = 1, F = 0, G = 1 + v^2$. Calculate the the intrinsic curvature $K_{intrinsic}^{helicoid}(u, v)$ of the helicoid.

Solution   We use EIGENMATH. First we compute the CHRISTOFFEL symbols using compact tensor techniques. Then we determine the intrinsic curvature by formula (8.1).

```
# EIGENMATH
"HELICOID"
U = (u,v)
-- coefficients of fff
E =1
F = 0
G = v^2+1

g = zero(2,2)
    g[1,1] = E
    g[1,2] = F
    g[2,1] = F
    g[2,2] = G

gu = inv(g)

Dg = d(g,U)
Dg

"CHRISTOFFEL Symbol 1st kind"
GammaI = 1/2 (Dg + transpose(Dg,2,3) - transpose(Dg,2,3,1,2))
GammaI

"CHRISTOFFEL Symbol 2nd kind"
GammaII = dot(gu, GammaI)
GammaII

"intrinsic curvature"
Ch111 = GammaII[1,1,1]
Ch121 = GammaII[1,2,1]
Ch211 = GammaII[2,1,1]
Ch221 = GammaII[2,2,1]
Ch222 = GammaII[2,2,2]

Kintrinsic = 1/g[1,1]*( d(Ch211,v) - d(Ch221,u) +
                        Ch221*(Ch111-Ch221)     +
                        Ch211*(Ch222-Ch121)     )

Kintrinsic
```

EIGENMATH output:

CHRISTOFFEL Symbol 2nd kind

$$\Gamma_{II} = \begin{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \\ 0 & \dfrac{v}{v^2+1} \end{bmatrix} \end{bmatrix}$$

intrinsic curvature

$$K_{intrinsic} = 0$$

● ▷ Click here to RUN this script.

Comment. The result $K^{helicoid}_{intrinsic}(u,v) \equiv 0$ demonstrates, that the helicoid has constant intrinsic curvature 0 at every point.

8.3 Exercises

Recall: the CHRISTOFFEL symbols Γ_{II} were calculated in §7.4.

● You may cross-check your results of the following exercises in [17].

Task: Calculate the intrinsic curvature $K_{intrinsic}(u,v)$ for the following surfaces using EIGENMATH . . .

Exercise 121. the cylinder $\phi(u,v) = (a\cos(v), a\sin(v), u)$. Choose $a = 1$.
Verify: $K^{cylinder}_{intrinsic}(u,v) = 0$

Exercise 122. the cone $\phi(u,v) = (u\cos(v), u\sin(v), u)$.
Verify: $K^{cone}_{intrinsic}(u,v) = 0$

Exercise 123. the elliptic paraboloid $\phi(u,v) = (u, v, u^2 + v^2)$.
Verify: $K^{ell.para.}_{intrinsic}(u,v) = \frac{4}{(4u^2+4v^2+1)^2}$

Exercise 124. the hyperbolic paraboloid $\phi(u,v) = (u, v, u^2 - v^2)$.
Verify: $K^{hyp.para.}_{intrinsic}(u,v) = \frac{-4}{(4u^2+4v^2+1)^2}$

Exercise 125. the monkey saddle $\phi(u,v) = (u, v, u^3 - 3uv^2)$.
Verify: $K^{monkey}_{intrinsic}(u,v) = -36\frac{u^2+v^2}{(9u^4+18u^2v^2+9v^4+1)^2}$

Exercise 126. the unit sphere $\phi(u,v) = (\cos(u)\sin(v), \sin(u)\sin(v), \cos(v))$.
Verify: $K^{sphere1}_{intrinsic}(u,v) = 1$

Exercise 127. the catenoid $\phi(u,v) = (c\cosh\frac{v}{c}\cos u, c\cosh\frac{v}{c}\sin u, v)$. Choose $c = 1$.
Verify: $K^{catenoid}_{intrinsic}(u,v) = 0$

Exercise 128. the torus $\phi(u,v) = ((b + a\cos(v))\cos(u), (b + a\cos(v))\sin(u), a\sin(v))$.
Verify: $K_{intrinsic}^{torus}(u,v) = \frac{\cos v}{2 + \cos v}$

Exercise 129. the function graph $\phi(u,v) = (u, v, f(u,v))$.
Verify: $K_{intrinsic}^{graph}(u,v) = \frac{f_{uu}f_{vv} - f_{uv}^2}{(f_u{}^2 + f_v{}^2 + 1)^2}$

Exercise 130. the surface of revolution $\phi(u,v) = (f(u)\cos(v), f(u)\sin(v), g(u))$.
Verify: $K_{intrinsic}^{revolution}(u,v) = \frac{g_u(-f_{uu}g_u + g_{uu}f_u)}{f(u)(f_u{}^2 + g_u{}^2)^2}$

Exercise 131. the PLÜCKER conoid $\phi(u,v) = (v \cdot \cos(u), v \cdot \sin(u), 2\cos(u) \cdot \sin(u))$.
Verify: $K_{intrinsic}^{Pluecker}(u,v) = \frac{-4(4\cos^4 u - 4\cos^2 u + 1)}{(\cos^4 v + 2\cos^2 u \cos^2 v - ...)}$

Exercise 132. the SCHERK surface $\phi(u,v) = (u, v, \ln(\frac{\cos v}{\cos u}))$.
Verify: $K_{intrinsic}^{Scherk}(u,v) = \frac{-\cos^2 u \cos^2 v}{(\cos^4 v + 2\cos^2 u \cos^2 v - ...)}$

Exercise 133. the ENNEPER $\phi(u,v) = (u - \frac{1}{3}u^3 + uv^2, v - \frac{1}{3}v^2 + vu^2, u^2 - v^2)$.
Verify: $K_{intrinsic}^{Enneper}(u,v) = \frac{-4}{(1 + 2u^2 + 2v^2 + u^4 + 2v^2u^2 + v^4)(u^2 + v^2 + 1)^2}$

Exercise 134. the BELTRAMI pseudosphere
$$\phi(u,v) = (\sin u \cos v, \sin u \sin v, \cos u + \ln(\sin u(\cos u + 1))).$$
Verify: $K_{intrinsic}^{Beltrami}(u,v) = -1$

Exercise 135. the hexenhut $\phi(u,v) = (\alpha \cdot \frac{\cos v}{\sqrt{u}}, \alpha \cdot \frac{\sin v}{\sqrt{u}}, u)$ where $\alpha^2 = \frac{2}{3\sqrt{3}}$.
Verify: $K_{intrinsic}^{hexenhut}(u,v) = \frac{-2187u^4}{(1 + 27u^3)^2}$

Exercise 136. the 'hexe' surface $\phi(u,v) = (\sqrt{u} \cdot \cos v, \sqrt{u} \cdot \sin v, u)$.
Verify: $K_{intrinsic}^{hexe}(u,v) = \frac{4}{(1 + 4u)^2}$

Exercise 137. the BIANCHI surface

$$\phi(u,v) = \begin{bmatrix} \frac{2 \cdot \sqrt{v^2 + 1} \cdot \sin(u) \cdot \cos(-v + \arctan(v))}{1 + v^2 \sin(u)^2} \\ \frac{-2 \cdot \sqrt{v^2 + 1} \cdot \sin(u) \cdot \sin(-v + \arctan(v))}{1 + v^2 \sin(u)^2} \\ \ln(\tan(\frac{1}{2u})) + \frac{2 \cdot \cos(u)}{1 + v^2 \sin(u)^2} \end{bmatrix}$$

Verify: $K_{intrinsic}^{cone}(u,v) = -1$

Exercise 138. Implement the intrinsic curvature $K_{intrinsic}(u,v)$ in EIGENMATH as a function `Kintrinsic(u,v)`, which only uses the functions `GammaI(i,j,k)` and `GammaII(i,j,k)`.
• ▷ Click here to SEE a solution.

9 RICCCI **Tensor and** GAUSS **Theorema egregium**

The *intrinsic curvature* of a 2D surface can be calculated using only the coefficients of the metric tensor g - without recourse to the shape operator who make use of the *extrinsic* concepts of the sff h.

9.1 **Definition -** RICCI **curvature**

Let $\phi \colon U \to \mathbb{R}^2$ be the parameterization of a surface S.
Let $g = (g_{ij})$ be the surface metric tensor of ϕ with inverse matrix $g^{-1} = (g^{ij})$.
Let $R^i{}_{jkl}$ be the RIEMANN curvature tensor of 2nd kind.
Then we define ...

a. ... the RICCI tensor Ric via a contraction w.r.t. the middle index of the RIEMANN curvature tensor by

$$\text{Ric}_{ik} := \sum_{j=1}^{2} R^j{}_{ijk} \tag{9.1}$$

where the EINSTEIN summation convention is decoded to prepare for the implementation of the formula in EIGENMATH.

b. ... the mixed RICCI tensor Ric via index juggling of the RICCI tensor by

$$\text{Ric}^j{}_i := \sum_{k=1}^{2} g^{jk} \text{Ric}_{ik} \tag{9.2}$$

c. ... the RICCI curvature (scalar) R via index contraction of the RICCI tensor by

$$\text{R} := \sum_{i=1}^{2} \sum_{k=1}^{2} g^{ik} \text{Ric}_{ik} \tag{9.3}$$

alias $\text{R} := \text{Ric}^i_i$.

Remark. The RICCI tensor Ric is often also called the *scalar curvature* of the surface S resp. of the metric g.

9.2   Gauss's Theorema Egregium

Theorem. (Gauss 1827)
LET $\phi\colon U \to \mathbb{R}^2$ be the parameterization of a surface S.
    Let $g = (g_{ij})$ be the surface metric tensor of ϕ with coefficients E, F, G.
    Let K be the Gauss *curvature and $K_{intrinsic}$ be the intrinsic curvature of g.*
    Let R be the Ricci *curvature scalar.*

THEN we have:

A. *The* Gauss *curvature is half of the* Ricci *curvature scalar, i.e.*

$$R = 2K \tag{9.4}$$

B. *The intrinsic curvature of g is g_{11}-times of the* Gauss *curvature, i.e.*

$$K_{intrinsic} = K \cdot g_{11} \tag{9.5}$$

C. *The* Gauss *curvature is explicit calculated via* Brioschi*'s formula from 1852, i.e.*

$$K = \frac{\det\begin{pmatrix} -\frac{1}{2}\frac{\partial^2 E}{\partial v^2} + \frac{\partial^2 F}{\partial u \partial v} - \frac{1}{2}\frac{\partial^2 G}{\partial u^2} & \frac{1}{2}\frac{\partial E}{\partial u} & \frac{\partial F}{\partial u} - \frac{1}{2}\frac{\partial E}{\partial v} \\ \frac{\partial F}{\partial v} - \frac{1}{2}\frac{\partial G}{\partial u} & E & F \\ \frac{1}{2}\frac{\partial G}{\partial v} & F & G \end{pmatrix} - \det\begin{pmatrix} 0 & \frac{1}{2}\frac{\partial E}{\partial v} & \frac{1}{2}\frac{\partial G}{\partial u} \\ \frac{1}{2}\frac{\partial E}{\partial v} & E & F \\ \frac{1}{2}\frac{\partial G}{\partial u} & F & G \end{pmatrix}}{(EG - F^2)^2} \tag{9.6}$$

Comment. The main statement of the **Theorema Egregium** from 1827 is: *the* Gauss
curvature K of a surface parameterization is a measure of its intrinsic geometry, i.e. K
can be calculated of the coefficients E, F, G of the metric tensor g alone! This is explicitly
stated in the Brioschi formula (9.6).
While the Weingarten map alias the shape operator describes the shape of a surface as
part of the ambient space, the numbers E, F, G and therefore the measurement of length
or area on the surface do not take any borrowing from the ambient space.
The Ricci tensor, the Ricci curvature and the Ricci curvature scalar are therefore con-
cepts of the intrinsic geometry of the surface. In contrast the Weingarten map alias the
shape operator are concepts of the extrinsic geometry of the surface – by using the normal
vector field of S.
It is the merit of C. F. Gauss to have shown that the Gauss curvature can be traced
back to the 3 real-valued functions E, F, G and their derivatives.

Remark. 1. A proof of A. is in Oloff [22, p.110].
2. A proof of B. is in Reckziegel [24, p.160].
3. A proof of C. is in Banchoff [2, p.230], Heckmann [13, p.30] or Pressly [23, p.233].

9.3   Implementation and Examples

Example 18. (Calculation of the RICCI curvature scalar R of the helicoid)
We do the calculation as before with/without tensor techniques in EIGENMATH. The
metric tensor g of the helicoid is known to have the coefficients $E = 1, F = 0, G = 1 + v^2$.
Calculate the the RICCI curvature scalar R of the helicoid.

Solution
1. We use EIGENMATH. First we compute the RICCI curvature Ric with compact tensor
techniques, cf. G. WEIGT [42]. We use formula (9.3) with intermediate steps by (9.1) and
(9.2).

```
# EIGENMATH
"HELICOID"
U = (u,v)

E =1                    -- coefficients of metric tensor g
F = 0
G = v^2+1

g = zero(2,2)
    g[1,1] = E
    g[1,2] = F
    g[2,1] = F
    g[2,2] = G

gu = inv(g)
Dg = d(g,U)

GammaI = 1/2 (Dg + transpose(Dg,2,3) - transpose(Dg,2,3,1,2))
GammaII = dot(gu, GammaI)

"Step 1. Calculate RIEMANN tensor Ri^jkl of 2nd kind."
T1 = d(GammaII, U)
T2 = dot(transpose(GammaII,2,3), GammaII)
RII = transpose(T1,3,4) - T1 + transpose(T2,2,3) - transpose(T2,2,3,3,4)
RII

"Step 2. Calculate RICCI tensor by (9.1). "
Ric = contract(RII,1,3)
Ric

"Step 3. Calculate RICCI curvature SCALAR by (9.3)."
R = contract( dot( gu, transpose(Ric)))
R
```

EIGENMATH output:

Step 1. Calculate Riemann tensor Ri^jkl of 2nd kind.

$$R_{II} = \begin{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \end{bmatrix}$$

Step 2. Calculate Ricci tensor by (9.1).

$$R_{ic} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

Step 3. Calculate Ricci curvature scalar by (9.3).

$$R = 0$$

● ▷ Click here to RUN this script.

Comment. The result $R^{helicoid} = 0$ demonstrates, that the helicoid has a vanishing RICCI curvature scalar at every point.

2. First we compute the RIEMANN curvature tensor Ric again using compact tensor techniques, because we have shown above how to do it using for-loops (one could include it here). We then use formula (9.3) with intermediate steps by (9.2) - but this time we demonstrate the attention of the EINSTEIN summation convention by explicit use of the sum()-command of EIGENMATH.

```
# EIGENMATH
"HELICOID"
U = (u,v)
g = ((1,0),(0,v^2+1))
gu = inv(g)
Dg = d(g,U)

GammaI = 1/2 (Dg + transpose(Dg,2,3) - transpose(Dg,2,3,1,2))
GammaII = dot(gu, GammaI)

"Step 1. Calculate Riemann tensor Ri^jkl of 2nd kind."
T1 = d(GammaII, U)
T2 = dot(transpose(GammaII,2,3), GammaII)
RII = transpose(T1,3,4) - T1 + transpose(T2,2,3) - transpose(T2,2,3,3,4)
RII

"Step 2. Calculate Ricci tensor using explicit for-loops."
Ric =zero(2,2)
for( m,1,2,
   for( q,1,2,                --   1   3 : contract 1-3
      Ric[m,q] = sum( s,1,2, RII[s,m,s,q]) ))
```

```
    Ric

    "Step 3. Calculate Ricci scalar R using explicit for-loops."
    R  = sum( i,1,2, sum(j,1,2, gu[i,j]*Ric[i,j]) )
    R
```

● ▷ Click here to RUN this script.
EIGENMATH output: same as above.

Exercise 139. (BRIOSCHI's formula)
Program BRIOSCHI's formula (9.6) in EIGENMATH and compute the GAUSS curvature K
for the helicoid with it.

Solution: sketch. Fill in the .. .

```
    # EIGENMATH
    phi(u,v) = (u cos(v), u sin(v), c v)  -- the HELICOID

    g = ((1,0),(0,c^2+u^2))                -- surface metric tensor
    do( E=g[1,1], F=g[1,2], G=g[1,1])    -- define E,F,G

    --define the 2 matrices
    M1 = ((-1/2*d(E,v,v)+d(F,u,v)-1/2d(G,u,u), .. , ..),
          (d(F,v)-1/2d(G,u)                   , .. , ..),
          (1/2 d(G,v)                         , .. , G ) )
    M1

    M2= ((0          , .. , .. ),
         (1/2 d(E,v), E  , F  ),
         (1/2 d(G,u), F  , G  ))
    M2

    -- define GAUSS curvature
    K = (det(M1)-det(M2))/(E G -F^2)
    K
```

Exercise 140. Compute the GAUSS curvature K for the 'saddle' surface $z = uv$ with
BRIOSCHI's formula.

9.4 The sine-Gordon Equation

$$\frac{\partial^2 u}{\partial x \partial y} = \sin(u)$$

This is the so-called *sine*-GORDON *equation*, which was considered in the study of surfaces of constant GAUSSian curvature $K = -1$, called *pseudospherical surfaces*. We show how this equation can be derived using the CAS EIGENMATH. In this example we demonstrate how to get this equation from the metric tensor resp. the 1st fundamental form of a surface with corresponding metric tensor g.

Task. Let $U \subset \mathbb{R}^2$ be the region of the metric tensor g, which is given by its 1st fundamental form

$$I = dx \wedge dx + \cos(u(x,y)) \cdot dx \wedge dy + \cos(u(x,y)) \cdot dy \wedge dx + dy \wedge dy$$

alias its metric coefficients

$$g_{11} = g_{22} = 1, \qquad g_{12} = g_{21} = \cos(u(x,y))$$

Calculate the RICCI curvature scalar of g , i.e. verify

$$R = -\frac{2}{\sin(u)} \frac{\partial^2 u}{\partial x \partial y}$$

Solution

We compute the RICCI scalar curvature R using tensor techniques. First we simplify the contravariant metric tensor $\begin{smallmatrix} Math: & g^{-1} \\ EigenM: & \mathtt{gu} \end{smallmatrix}$ having seen the intermediate result in (1). Then we go on.

```
# EIGENMATH
"sine GORDON equation"
X = (x,y)                              -- u and v depends on x and y
U = (u(x,y), v(x,y))
g = ((1,cos(u(x,y))), (cos(u(x,y)),1))    --(1)
g
gu = inv(g)
gu                                        --(2)
     gu[1,1] = 1/sin(u(x,y))^2
     gu[1,2] = - cos(u(x,y))/sin(u(x,y))^2
     gu[2,1] = gu[1,2]
     gu[2,2] = 1/sin(u(x,y))^2
gu

Dg = d(g,X)
Dg
```

```
GammaI = 1/2 (Dg + transpose(Dg,2,3) - transpose(Dg,2,3,1,2))
GammaI
GammaII = dot(gu, GammaI)
GammaII

"RIEMANN Ri^jkl tensor."
T1 = d(GammaII, X)
T2 = dot(transpose(GammaII,2,3), GammaII)
RII = transpose(T1,3,4) - T1 + transpose(T2,2,3) - transpose(T2,2,3,3,4)
RII

"RIEMANN Rijkl tensor."
RI = dot(g, RII)
RI

"RICCI tensor."
Ric = contract(RII,1,3)
Ric

"RICCI curvature scalar"
R = contract( dot( gu, transpose(Ric)))
R
```

EIGENMATH output:

RICCI curvature scalar

$$R = -\frac{2\cos(u(x,y))^3\, d(u(x,y),x)\, d(u(x,y),y)}{\sin(u(x,y))^4} + \frac{2\cos(u(x,y))^2\, d(d(u(x,y),x),y)}{\sin(u(x,y))^3} - \frac{2\cos(u(x,y))\, d(u(x,y),x)\, d(u(x,y),y)}{\sin(u(x,y))^2} + \frac{2\cos(u(x,y))\, d(u(x,y),x)\, d(u(x,y),y)}{\sin(u(x,y))^4} - \frac{2\, d(d(u(x,y),x),y)}{\sin(u(x,y))^3}$$

● ▷ Click here to RUN this script.

Comment. The long result for R has to be simplified by the user. Therefore we translate the return value of EIGENMATH in math language. We use the shortcut $c := \cos(u(x,y))$ and $s := \sin(u(x,y))$. Then the output of EIGENMATH reads:

$$
\begin{aligned}
R &= -\frac{2c^3 u_x u_y}{s^4} + \frac{2c^2 u_{xy}}{s^3} - \frac{2c\cdot u_x u_y}{s^2} + \frac{2c\cdot u_x u_y}{s^4} - \frac{2u_{xy}}{s^3} \\
&= \ldots \qquad \text{(use trigo formulae } c^2 + s^2 = 1 \text{ and } 1 - c^2 = s^2\text{)} \\
&= \frac{-2u_{xy}}{s} \\
&= -\frac{2}{\sin(u)}\frac{\partial^2 u}{\partial x \partial y} \qquad \text{q.e.d.}
\end{aligned}
$$

Remark. 1 Our presentation of the sine-Gordon equation followed STEEB in [35, p. 64–67 and p. 363–365], where an implementation with for-loops is given in REDUCE and C++.
2 For more info see e.g. wiki: Sine-Gordon equation
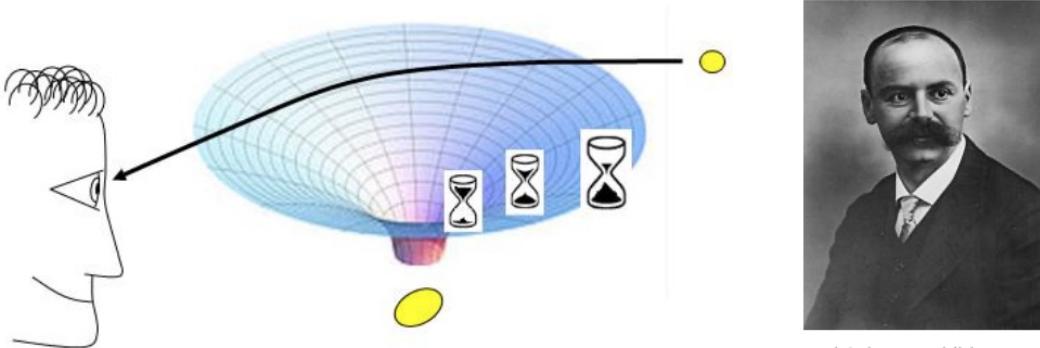
9.5   The Ricci scalar of the Schwarzschild Spacetime Metric

Über das Gravitationsfeld eines Massenpunktes
nach der Einsteinschen Theorie.

Von K. Schwarzschild.

(Vorgelegt am 13. Januar 1916 [s. oben S. 42].)

Karl Schwarzschild

Figure 13: Descriptive representation of the Schwarzschild metric: A two-dimensional area of space is considered. In the vicinity of the mass, the distances are lengthened. This is illustrated by the fact that the surface has been stretched in a visually existing further direction. Time slows down near the mass. This is illustrated by the fact that relatively little sand has trickled through the hourglass. The ray of light entering the eye from the light source would have to traverse a very stretched surface if it were to travel closer to the mass. Therefore, the light beam makes an arc around the mass, to quote Carmesin [3].

Exercise 141. In Einstein's theory of General Relativity, the Schwarzschild metric (also known as *the Schwarzschild solution*) is the first solution of Einstein's field equations, which was found in 1916 by the German astrophysicist Karl Schwarzschild (1873 - 1916). It describes the gravitational field of a homogeneous, non-charged and non-rotating sphere and is an example of the curvature of space-time, where light paths and trajectories follow the curvature of the space-time metric.

The Schwarzschild metric is a spherically symmetric Lorentzian metric defined on the subset $U = \mathbb{R} \times (2M, \infty) \times S^2$[10] in ambient 4D spacetime.
In Schwarzschild coordinates $(t, r, \theta, \phi) \in U$ the Schwarzschild *1st fundamental form*

---

[10]where $S^2 \subset \mathbb{R}^3$ is the unit sphere in $\mathbb{R}^3$.

has the expression[11] with sign convention $(-,+,+,+)$

$$\mathrm{d}s^2 = -\left(1 - \frac{2M}{r}\right)\mathrm{d}t^2 + \frac{1}{1 - \frac{2M}{r}}\mathrm{d}r^2 + r^2\mathrm{d}\theta^2 + r^2\sin^2\theta\mathrm{d}\phi^2 \tag{9.7}$$

resp. the SCHWARZSCHILD metric tensor is the matrix g with $h(r) := 1 - 2\frac{M}{r}$ and sign convention $(+,-,-,-)$

$$g = \begin{bmatrix} h(r) & 0 & 0 & 0 \\ 0 & -\frac{1}{h(r)} & 0 & 0 \\ 0 & 0 & -r^2 & 0 \\ 0 & 0 & 0 & -r^2\sin(\theta)^2 \end{bmatrix} \tag{9.8}$$

a. Verify using the arbitrary $h(r)$ term:

RICCI curvature scalar

$$R = \frac{4\,\mathrm{d}(h(r),r)}{r} - \frac{2}{r^2} + \frac{2\,h(r)}{r^2} + \mathrm{d}(\mathrm{d}(h(r),r),r)$$

i.e.

$$R = \frac{4\frac{\partial h(r)}{\partial r}}{r} - \frac{2}{r^2} + \frac{2h(r)}{r^2} + \frac{\partial^2 h(r)}{\partial r\partial r}$$

b. Verify: if $h(r) := 1 - 2\frac{M}{r}$ with $M \in \mathbb{R}_+$ arbitrary, then $R = 0$.

● ▷ Click here to RUN the solution.

Remark. 1. In this example EIGENMATH works in a 4D space with coordinates (t, r, θ, ϕ), but using tensor techniques to calculate the CHRISTOFFEL tensors, the RIEMANN curvature tensors or the RICCI curvature scalar gives the expected results.
2. We use the conventions and notations in [22, p.5, p.52, p.96].
3. An EIGENMATH solution to this problem is given by G. WEIGT in [42].

Exercise 142. (Post-EINSTEIN-SCHWARZSCHILD-Metric)
a. Read the text ▷ Post-Einstein-Schwarzschild-Metrik . Although it is written in German, you should be able to recognize and understand all formulas.
b. Using the metric tensor in (3) on p.2 of this script, calculate the CHRISTOFFEL tensors, the RIEMANN curvature tensors or the RICCI curvature scalar using tensor techniques by EIGENMATH.

---

[11]By replacing M with GM/c^2, with G as the gravitational constant and t with $c \cdot t$ (c speed of light), one reconnects to the physical system of measurements.

9.6 The RICCI scalar of the GÖDEL Spacetime Metric

An Example of a New Type of Cosmological Solutions of Einstein's Field Equations of Gravitation

KURT GÖDEL
Institute for Advanced Study, Princeton, New Jersey

448   K U R T   G Ö D E L

where a is a positive number. The matrices of the g_{ik} and g^{ik}, therefore, are the two:

$$a^2 \cdot \begin{Vmatrix} 1 & 0 & e^{x_1} & 0 \\ 0 & -1 & 0 & 0 \\ e^{x_1} & 0 & e^{2x_1}/2 & 0 \\ 0 & 0 & 0 & -1 \end{Vmatrix}, \quad \frac{1}{a^2} \cdot \begin{Vmatrix} -1 & 0 & 2e^{-x_1} & 0 \\ 0 & -1 & 0 & 0 \\ 2e^{-x_1} & 0 & -2e^{-2x_1} & 0 \\ 0 & 0 & 0 & -1 \end{Vmatrix}.$$

Owing to the fact that only two of the forty $\partial g_{ik}/\partial x_l$ are $\neq 0$, namely $\partial g_{22}/\partial x_1$ and $\partial g_{02}/\partial x_1$, the $\Gamma_{i,kl}$ and $\Gamma_{kl}{}^i$ can very easily be computed. One obtains the values:

$$\Gamma_{0,12} = -\Gamma_{1,02} = \Gamma_{2,01} = (a^2/2)e^{x_1},$$
$$\Gamma_{1,22} = -\Gamma_{2,12} = -(a^2/2)e^{2x_1},$$
$$\Gamma_{01}{}^0 = 1, \quad \Gamma_{12}{}^0 = \Gamma_{02}{}^1 = e^{x_1}/2,$$
$$\Gamma_{22}{}^1 = e^{2x_1}/2, \quad \Gamma_{01}{}^2 = -e^{-x_1}.$$

These $\Gamma_{i,kl}$ and $\Gamma_{kl}{}^i$, and those obtained from them by interchanging the last two (or the lower two) indices are the only ones that do not vanish.

Using for R_{ik} the formula[7]

$$R_{ik} = \frac{\partial}{\partial x_\sigma}\Gamma_{ik}{}^\sigma - \frac{1}{2}\frac{\partial^2 \log g}{\partial x_i \partial x_k} + \frac{1}{2}\Gamma_{ik}{}^\sigma \frac{\partial \log g}{\partial x_\sigma} - \Gamma_{\sigma i}{}^\rho \Gamma_{\rho k}{}^\sigma,$$

and taking account of the fact that $\partial/\partial x_i$, except for $i=1$, vanishes for every magnitude of the solution, and that $g = (a^8/2)e^{2x_1}$, we obtain

$$R_{ik} = \frac{\partial}{\partial x_1}\Gamma_{ik}{}^1 + \Gamma_{ik}{}^1 - \Gamma_{\rho i}{}^\sigma \Gamma_{\sigma k}{}^\rho.$$

This yields the values for the R_{ik}

$$R_{00} = 1, \quad R_{22} = e^{2x_1}, \quad R_{02} = R_{20} = e^{x_1};$$

all other R_{ik} vanish. Hence,

$$R = 1/a^2.$$

Exercise 143. For the given metric tensor (g_{ik}), check the results of GÖDEL's calculations for the CHRISTOFFEL tensors, the RICCI tensor and the RICCI curvature scalar $R = \frac{1}{a^2}$.

9.7 Exercises

Let's calculate the RICCI curvature tensor Ric and the RICCI curvature scalar R for our small collection of classical 2D surfaces. Sometimes you should help EIGENMATH to simplify trigonometric expressions or to simplify fractions.

● *Calculate the* RICCI *curvature tensor* Ric *and* RICCI *curvature scalar* R *for the following surfaces using* EIGENMATH *...*

Exercise 144. ...the saddle $\phi(u,v) = (u,v,uv)$.

Exercise 145. the cylinder $\phi(u,v) = (a\cos(v), a\sin(v), u)$. Choose $a = 1$.

Exercise 146. the cone $\phi(u,v) = (u\cos(v), u\sin(v), u)$.

Exercise 147. the elliptic paraboloid $\phi(u,v) = (u,v,u^2 + v^2)$.

Exercise 148. the hyperbolic paraboloid $\phi(u,v) = (u,v,u^2 - v^2)$.

Exercise 149. the monkey saddle $\phi(u,v) = (u,v,u^3 - 3uv^2)$.

Exercise 150. the unit sphere $\phi(u,v) = (\cos(u)\sin(v), \sin(u)\sin(v), \cos(v))$.

Exercise 151. the catenoid $\phi(u,v) = (c\cosh\frac{v}{c}\cos u, c\cosh\frac{v}{c}\sin u, v)$. Choose $c = 1$.

Exercise 152. the torus $\phi(u,v) = ((b + a\cos(v))\cos(u), (b + a\cos(v))\sin(u), a\sin(v))$.

Exercise 153. the function graph $\phi(u,v) = (u,v,f(u,v))$.

Exercise 154. the surface of revolution $\phi(u,v) = (f(u)\cos(v), f(u)\sin(v), g(u))$.

Exercise 155. the PLÜCKER conoid $\phi(u,v) = (v\cdot\cos(u), v\cdot\sin(u), 2\cos(u)\cdot\sin(u))$.

Exercise 156. the SCHERK surface $\phi(u,v) = (u,v,\ln(\frac{\cos v}{\cos u}))$.

Exercise 157. the ENNEPER surface $\phi(u,v) = (u - \frac{1}{3}u^3 + uv^2, v - \frac{1}{3}v^2 + vu^2, u^2 - v^2)$.

Exercise 158. the BELTRAMI pseudosphere
$$\phi(u,v) = (\sin u\cos v, \sin u\sin v, \cos u + \ln(\sin u(\cos u + 1))).$$

Exercise 159. the hexenhut $\phi(u,v) = (\alpha/\sqrt{u}\cdot\cos v, \alpha/\sqrt{u}\cdot\sin v, u)$ where $\alpha^2 = \frac{2}{3\sqrt{3}}$.

Exercise 160. the 'hexe' surface $\phi(u,v) = (\sqrt{u}\cdot\cos v, \sqrt{u}\cdot\sin v, u)$.

Exercise 161. the BIANCHI surface

$$\phi(u,v) = \begin{bmatrix} \frac{2\cdot\sqrt{v^2+1}\cdot\sin(u)\cdot\cos(-v+\arctan(v))}{1+v^2\sin(u)^2} \\ \frac{-2\cdot\sqrt{v^2+1}\cdot\sin(u)\cdot\sin(-v+\arctan(v))}{1+v^2\sin(u)^2} \\ \ln(\tan(\frac{1}{2u})) + \frac{2\cdot\cos(u)}{1+v^2\sin(u)^2} \end{bmatrix}$$

10   The Fundamental Theorem

First we quote the well known equations from WEINGARTEN, GAUSS and CODAZZI-MAINARDI. Then we formulate the legendary *Fundamental Theorem of surface theory.*

10.1   The WEINGARTEN Equations

The WEINGARTEN equations express the partial derivatives of the surface normal $\vec{n}$ as a linear combination of the tangential basis vectors, where its coefficient are calculated only using the coefficients of the 1st and 2nd fundamental forms. These equation were found by Weingarten in 1861.

Theorem. (WEINGARTEN Equations)
LET $\phi \colon U \to S \subset \mathbb{R}^3$ be a parameterization of a surface S in $\mathbb{R}^3$.
   Let $(\phi_u, \phi_v) = (\frac{\partial \phi}{\partial u}, \frac{\partial \phi}{\partial v})$ be the local basis in T_pS.
   Let $g = (g_{ij}) = \begin{bmatrix} E & F \\ F & G \end{bmatrix}$ be the surface metric tensor of ϕ.
   Let $h = (h_{ij}) = \begin{bmatrix} L & M \\ M & N \end{bmatrix}$ be the 2nd fundamental tensor of ϕ.
   Let $\vec{n} := \frac{\phi_u \times \phi_v}{|\phi_u \times \phi_v|}$ be the unit normal of S in p.[12]

THEN we have the two WEINGARTEN equations

$$\vec{n}_u = \frac{MF - LG}{EG - F^2} \cdot \phi_u + \frac{LF - ME}{EG - F^2} \cdot \phi_v \tag{10.1}$$

$$\vec{n}_v = \frac{NF - MG}{EG - F^2} \cdot \phi_u + \frac{MF - NE}{EG - F^2} \cdot \phi_v \tag{10.2}$$

Remark. For a proof cf. [13, p.26].

Example 19. (Check the WEINGARTEN Equations for the helicoid)
We consider the parametric surface helicoid with shape constant c.
To verify the validity of the WEINGARTEN Equations we use the surface basis ϕ_u, ϕ_v.

```
# EIGENMATH
phi(u,v) = (u cos(v), u sin(v), c v)  -- HELICOID

U =(u,v)                              -- coordinates of surface

g = ((1,0),(0,c^2+u^2))               -- surface metric tensor
do( E=g[1,1], F=g[1,2], G=g[2,2])     --(1) name it E, F, G

h=((0, -c/sqrt(c^2+u^2)), (-c/sqrt(c^2+u^2),0))   -- known 2nd fu. tensor
```

---

[12] To avoid name clash, we use the symbol $\vec{n}$ for the normal vector, because N is used in the formulae to denote the 4th coefficient of the sff *h*.

```
do( L=h[1,1], M=h[1,2], N=h[2,2])       --(2) name it L, M, N

-- surface normal
n = cross(d(phi,u), d(phi,v))           --(3)
n[3]=u                                   --(4)
n
-- surface unit normal
an = abs(n)                              --(5)
an = sqrt(c^2+u^2)
an
n  = n/an                               --(6)
n
-- compare LHS and RHS of WEINGARTEN eq. (10.1)
RHS101 = 1/(E G - F^2)*(M F - L G)*d(phi,u) +
          1/(E G - F^2)*(L F - M E)*d(phi,v)
RHS101

LHS101 = d(n,u)
LHS101                                   --(7)
```

● ▷ Click here to RUN the solution.

EIGENMATH output:

$$
L_{HS101} = \begin{bmatrix} -\dfrac{c\,u\,\sin(v)}{\left[c^2 + u^2\right]^{3/2}} \\[2ex] \dfrac{c\,u\,\cos(v)}{\left[c^2 + u^2\right]^{3/2}} \\[2ex] -\dfrac{u^2}{\left[c^2 + u^2\right]^{3/2}} + \dfrac{1}{\left[c^2 + u^2\right]^{1/2}} \end{bmatrix}
\qquad
R_{HS101} = \begin{bmatrix} -\dfrac{c\,u\,\sin(v)}{\left[c^2 + u^2\right]^{1/2}} \\[2ex] \dfrac{c\,u\,\cos(v)}{\left[c^2 + u^2\right]^{1/2}} \\[2ex] \dfrac{c^2}{\left[c^2 + u^2\right]^{1/2}} \end{bmatrix}
$$

Comment. The 1st and 2nd ff g and h are known and given here directly in their matrices. In (1) we give the coefficients of the fff g their classic names E, F, G and in (2) we name the coefficients of the sff h classic as L, M, N. Line (3) constructs and simplifies (by mind) the normal vector n by means of the cross product, calculates his length in (5) and normalize n to unit length (6).

We then write down the 1st Weingarten equation (10.1) separeted in the LHS term named LHS101 and the RHS term abbreviated RHS101 in EIGENMATH. We have to verify, that RHS101=LHS101 in order to show the equality. The RHS is delivered automatically by EIGENMATHs output, to check that this term is equal to the LHS term we help EIGENMATH a little bit in the 3rd component in (7):

$$
-\frac{u^2}{(c^2 + u^2)^{3/2}} - \frac{1}{(c^2 + u^2)^{1/2}} = \frac{-u^2 + (c^2 + u^2)}{(c^2 + u^2)^{3/2}} = \frac{c^2}{(c^2 + u^2)^{3/2}} = \text{RHS101} \qquad (10.3)
$$

q.e.d.

Exercise 162. Verify the validity of the WEINGARTEN equation (10.2) for the helicoid.

Exercise 163. Verify the validity of the WEINGARTEN equations for the surface $z = uv$.

Exercise 164. Verify the validity of the WEINGARTEN equations for the monkey saddle, the cone and the hyperbolic paraboloid and ... what you like.

10.2   The GAUSS Equations

The Gauss equations are a system of three partial differential equations. They describe a connection between the tangential vectors, the unit normal vector of a parametrized surface and their coefficients of the 2nd fundamental form.

Theorem. (GAUSS Equations)
LET $\phi\colon U \to S \subset \mathbb{R}^3$ be a parameterization of a surface S in $\mathbb{R}^3$.
   Let $E = (E_1, E_2) := (\phi_u, \phi_v) = (\frac{\partial \phi}{\partial u}, \frac{\partial \phi}{\partial v})$ be the local base in $T_p S$. $\phi_{uu} := \frac{\partial^2 \phi}{\partial u \partial u}$ etc.
   Let $g = (g_{ij})$ be the surface metric tensor of ϕ.
   Let $h = (h_{ij}) = \begin{bmatrix} L & M \\ M & N \end{bmatrix}$ be the 2nd fundamental tensor of ϕ.
   Let $\vec{n} := \frac{\phi_u \times \phi_v}{|\phi_u \times \phi_v|}$ be the unit normal of S in p.[13]
   Let $\Gamma^{II} = (\Gamma^1, \Gamma^2)$ be the CHRISTOFFEL symbol coefficients of 2nd kind.

THEN we have the three GAUSS equations

$$
\begin{align}
\phi_{uu} &= \Gamma_{11}^1 \cdot \phi_u + \Gamma_{11}^2 \cdot \phi_v + L \cdot \vec{n} \tag{10.4} \\
\phi_{uv} &= \Gamma_{12}^1 \cdot \phi_u + \Gamma_{12}^2 \cdot \phi_v + M \cdot \vec{n} \tag{10.5} \\
\phi_{vv} &= \Gamma_{22}^1 \cdot \phi_u + \Gamma_{22}^2 \cdot \phi_v + N \cdot \vec{n} \tag{10.6}
\end{align}
$$

Proof: cf. OLOFF [22, p.93], PRESSLEY [23, p.240] or ▷ Derivation of classical equations. □

Remark. If we denote the coordinate basis by $E = (E_1, E_2)$ and the parameters as $(u^1, u^2) = (u, v)$ with integer indices, then the GAUSS equations may be written by

$$
\begin{align}
\frac{\partial E_1}{\partial u^1} &= \Gamma_{11}^1 \cdot E_1 + \Gamma_{11}^2 \cdot E_2 + L \cdot \vec{n} \\
\frac{\partial E_1}{\partial u^2} &= \Gamma_{12}^1 \cdot E_1 + \Gamma_{12}^2 \cdot E_2 + M \cdot \vec{n} \\
\frac{\partial E_2}{\partial u^2} &= \Gamma_{22}^1 \cdot E_1 + \Gamma_{22}^2 \cdot E_2 + N \cdot \vec{n}
\end{align}
$$

and then compactly expressed with the EINSTEIN summation convention by one tensorial equation ($i, j \in \{1, 2\}$)

$$
\frac{\partial E_i}{\partial u^j} = \Gamma_{ij}^k \cdot E_k + h_{ij} \cdot \vec{n} \tag{10.7}
$$

We now verify the validity of the Gauss Equations for some surfaces.

---

[13]To avoid name clash, we use the symbol $\vec{n}$ for the normal vector, because N is used in the formulae to denote the 4th coefficient of the sff.

Example 20. (Gauss Equations for the helicoid)
We consider the parametric surface helicoid with shape constant c. To verify the validity
of the GAUSS Equations we use the surface basis ϕ_u, ϕ_v and their derivatives $\phi_{uu}, \phi_{uv} = \phi_{vu}, \phi_{vv}$, i.e. the second partial derivatives of the tangential basis vectors, which appears
on the LHS of the GAUSS equations.

```
# EIGENMATH
phi(u,v) = (u cos(v), u sin(v), c v)  -- parametrized helicoid
U =(u,v)                              -- coordinates of surface

g = ((1,0),(0,c^2+u^2))               -- surface metric tensor
g
h=((0, -c/sqrt(c^2+u^2)), (-c/sqrt(c^2+u^2),0))
h                                     -- 2nd fund. form

do( L=h[1,1], M=h[1,2], N=h[1,1])     --(1)
do( print(L), print(M), print(N))     --(2)

n = cross(d(phi,u), d(phi,v))         --(3)
n[3]=u                                --(4)
n
an = abs(n)                           --(5)
an = sqrt(c^2+u^2)
an
n  = n/an                             --(6)
n

gu = inv(g)
gu

Dg = d(g, U)
Dg

GammaI  = 1/2 (Dg + transpose(Dg,2,3) - transpose(Dg,2,3,1,2))
GammaII = dot(gu, GammaI)
GammaII

"GAUSS equation check"

d(phi,u,u) - (GammaII[1,1,1]*d(phi,u) + GammaII[2,1,1]*d(phi,v) + L*n)  --(G1)
d(phi,u,v) - (GammaII[1,1,2]*d(phi,u) + GammaII[2,1,2]*d(phi,v) + M*n)  --(G2)
          sin(v)*(c^2+u^2)/(c^2+u^2) - sin(v)                          --(7)
          -cos(v)*(c^2+u^2)/(c^2+u^2) + cos(v)

d(phi,v,v) - (GammaII[1,2,2]*d(phi,u) + GammaII[2,2,2]*d(phi,v) + N*n)  --(G3)
```

EIGENMATH output:

GAUSS equation check

$$\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} \dfrac{c^2 \sin(v)}{c^2 + u^2} + \dfrac{u^2 \sin(v)}{c^2 + u^2} - \sin(v) \\[3ex] -\dfrac{c^2 \cos(v)}{c^2 + u^2} - \dfrac{u^2 \cos(v)}{c^2 + u^2} + \cos(v) \\[3ex] 0 \end{bmatrix}$$

0

0

$$\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

- ▷ Click here to RUN the script.

Comment. The 1st and 2nd ff g and h are known and given here directly in their matrices. In (1) we give the coefficients of the sff their usual names L, M, N and display them in (2). Line (3) constructs and simplifies (by mind) the normal vector n by means of the cross product, calculates his length in (5) and normalize n to unit length (6).
(G1), (G2),(G3) are the three GAUSS equations expressed in EIGENMATH. All terms of the three equations are collected on the LHS, so we have to verify, that the result is 0 in order to show the validity. The result $(G1) = 0$ and $(G3) = 0$ is delivered automaticly by EIGENMATHs output, to check that $(G2) = 0$ we help EIGENMATH a little bit in (7). Ok.

Exercise 165. Verify the validity of the GAUSS equations for the surface $z = uv$.

Exercise 166. Verify the validity of the GAUSS Equations for
– the monkey saddle
– the cone
– the hyperbolic paraboloid
...

10.3 The Codazzi-Mainardi Equations

The Codazzi-Mainardi equations are important formulas which link together the metric tensor g and the second fundamental form h of a parametric surface by means of the Christoffel symbol of the 2nd kind..

Theorem. (Codazzi-Mainardi Equations)
LET $\phi \colon U \to S \subset \mathbb{R}^3$ be a parameterization of a surface S in $\mathbb{R}^3$.
Let $g = (g_{ij}), h = (h_{ij}) = \begin{bmatrix} L & M \\ M & N \end{bmatrix}$ be the 1st and 2nd fundamental tensors of ϕ .
Let $\Gamma^{II} = (\Gamma^1, \Gamma^2)$ be the Christoffel symbol coefficients of 2nd kind as above.

THEN we have the two Codazzi-Mainardi equations

$$
\begin{aligned}
L_v - M_u &= L \cdot \Gamma^1{}_{12} + M \cdot \left(\Gamma^2{}_{12} - \Gamma^1{}_{11} \right) - N \cdot \Gamma^2{}_{11} && (10.8) \\
M_v - N_u &= L \cdot \Gamma^1{}_{22} + M \cdot \left(\Gamma^2{}_{22} - \Gamma^1{}_{12} \right) - N \cdot \Gamma^2{}_{12} && (10.9)
\end{aligned}
$$

or by compacting both equations via the Einstein summation convention into one tensorial equation for $i, j, l \in \{1, 2\}$

$$
\frac{\partial h_{ij}}{\partial u^l} - \Gamma^k{}_{il} \cdot h_{kj} = \frac{\partial h_{il}}{\partial u^j} - \Gamma^k{}_{ij} \cdot h_{kl} \tag{10.10}
$$

Proof: cf. Pressley [23, p.241], Banchoff [2, p.229] or ▷ Derivation of classical equations. □

Example 21. (Codazzi-Mainardi equations for the helicoid)
We consider the parametric surface helicoid with shape constant c. To verify the validity of the Codazzi-Mainardi equations we need the coefficients L, M, N and the partial derivatives of these coefficients of the sff h w.r.t. u and v, i.e. L_v, M_u, M_v, N_u and the coefficients of Γ^{II}. Again we collect all terms of the equation on the LHS and show, that LHS $= 0$.

```
# EIGENMATH
phi(u,v) = (u cos(v), u sin(v), c v)    -- HELICOID
U =(u,v)                                -- coordinates of surface

g = ((1,0),(0,c^2+u^2))                 -- surface metric tensor
h = ((0, -c/sqrt(c^2+u^2)), (-c/sqrt(c^2+u^2),0))  -- 2nd FF

do( L=h[1,1], M=h[1,2], N=h[1,1])       -- name hij
do( print(L), print(M), print(N))       -- display L, M, N

gu = inv(g)
Dg = d(g, U)
```

```
GammaI  = 1/2 (Dg + transpose(Dg,2,3) - transpose(Dg,2,3,1,2))
GammaII = dot(gu, GammaI)
GammaII

"CODAZZI equation check"

d(L,v)-d(M,u) -                             --(CM1)
     (L*GammaII[1,1,2] +
      M*(GammaII[2,1,2]-GammaII[1,1,1])  -
      N*GammaII[2,1,1] )

d(M,v)-d(N,u) -                             --(CM2)
     (L*GammaII[1,2,2] -
      M*(GammaII[2,2,2]-GammaII[1,1,2])  -
      N*GammaII[2,1,2])
```

● ▷ Click here to RUN the script.
EIGENMATH output:

CODAZZI equation check

0

0

Comment. We implement the CODAZZI-MAINARDI equations 1:1 from equations (10.9) and (10.10) in lines (CM1) and (CM2). EIGENMATHs output displays that $(CM1) = (CM2) = 0$. Ok.

Exercise 167. Verify the validity of the CODAZZI-MAINARDI equations for the surface $z = uv$.

Exercise 168. Verify the validity of the CODAZZI-MAINARDI equations for
– the monkey saddle
– the cone
– the hyperbolic paraboloid
– a function graph
...

Exercise 169. Verify the validity of the CODAZZI-MAINARDI equations for the surface of revolution with the parametrization $\phi\colon U \to \mathbb{R}^3$ defined by $\phi(u,v) = (u\cos(v), u\sin(v), f(u))$ with $f\colon \mathbb{R} \to \mathbb{R}_+$ arbitrary.

   Why is it important to test if the CODAZZI-MAINARDI equations are fulfilled for the 1st and 2nd fundamental forms of a surface? The answer is given in the so-called *Fundamental Theorem of Surfaces Theory.*

10.4   The Fundamental Theorem of Surfaces Theory

> The question arises to what extent a parametric surface is (uniquely) determined by specifying the first and second fundamental forms. If one calculates the second derivatives of the basis vectors of the tangent space of the surface, one finds that the coefficients of the first and second fundamental forms cannot be chosen completely independently of each other. The necessary conditions work in the form of the CODAZZI-MAINARDI equations and BRIOSCHI's formula. It is a fact that these necessary conditions are also sufficient. Looking back, this motivates our short treatment of the CODAZZI-MAINARDI equation.

With the CODAZZI-MAINARDI equations we are now ready to formulate the so-called *fundamental theorem of the surface theory*, which is analogous to the fundamental theorem of curve theory. It was published by O. BONNET in 1867.

Theorem. (The Fundamental Theorem of Surfaces Theory)
LET the coefficients of the matrix $g, h\colon U \to \mathbb{R}$ of the first and second fundamental forms
    (1) be C^2-functions of the parameters u and v and
    (2) satisfy the CODAZZI-MAINARDI equations and
    (3) satisfy BRIOSCHI's formula for the calculation of the GAUSS curvature K,

THEN there exists a uniquely determined (up to translations and rotations) surface S, *which owns these given first and second fundamental forms, i.e. $I_S = g$ and $II_S = h$.*

Proof: Cf. BANCHOFF [2, p.238], LIPSCHUTZ [19, p.203] or ▷ FuTheorem of surfaces, GLUCK [10, p.34] □

Remark. The conditions (2) and (3) are called the *compatibility equations*. They answer the important existence question, if it is possible to construct a 2D surface with prescribed 1st and 2nd fundamental tensor coefficients.

10.5   Exercises

Exercise 170. Verify that the functions $E = 1 + 4u^2, F = -4uv, G = 1 + 4v^2, L = 2(4u^2 + 4v^2 + 1)^{-1/2}, M = 0, N = -2(4u^2 + 4v^2 + 1)^{-1/2}$ satisfy the compatibility equations.

Exercise 171. (existence of a surface with precribed fundamental coefficients)
Let $(E, F, G) = (1, 0, \sin^2 u)$ and $(L, M, N) = (1, 0, \sin^2 u)$ be candidates for the coefficients for the 1st and 2nd fundamental tensors of a surface S, cf. [19, p.204].
a. Check, that $E, .., N$ satisfy the compatibility equations, i.e. that the CODAZZI-MAINARDI equations and BRIOSCHI's formula are fulfilled.
b. Show, that the surface S with this set of given fundamental coefficients is the sphere of radius 1.
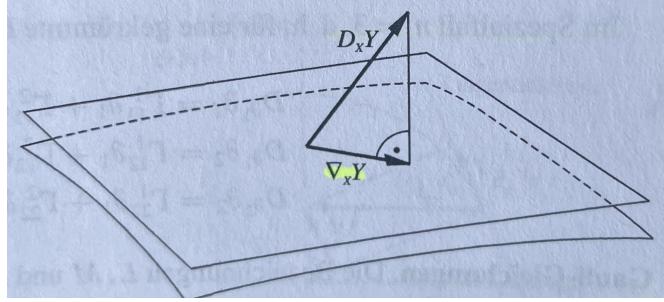
**Exercise 172.** Study the example 10.2 in [23, p.242] giving a cylinder of radius 1.

Exercise 173. Verify that there is no surface S parametrized by $\phi(u, v)$ with 1st and 2nd fundamental forms $I = du^2 + \cos^2 u \, dv^2$ and $II = \cos^2 u \, du^2 + dv^2$, cf. [23, p.244].

Exercise 174. Do problem 9 in GLUCK [10, p.37].

Exercise 175. Do problem 10 in GLUCK [10, p.38].

11   Covariant Derivative

Figure 14: $D_X Y$: the *directional* derivative in $\mathbb{R}^3$, e.g. grad.
The result is element of the *ambient* vector space.
$\nabla_X Y$: the *covariant* derivative in TpS. The result is in the
tangential vector space, i.e. $\nabla_X Y$ is the orthogonal projec-
tion of the directional derivative $D_X Y$ onto TpM.

The CHRISTOFFEL symbol Γ will now be used to formulate the concept of the *covariant derivative* of two vector fields X and Y on the tangent space $T_p S$ of a parametrized surface S in ambient space $\mathbb{R}^3$.

11.1   Definition

Let $\phi \colon U \to S \subset \mathbb{R}^3$ be a parameterization of a surface S in $\mathbb{R}^3$.
Let $E = (E_1, E_2) := (\phi_u, \phi_v)$ be the local base in $T_p S$.
Let $g = (g_{ij})$ be the surface metric tensor of ϕ.
Let $X, Y \colon U \to T_p S$ be two vector fields defined by their coefficients as $X = \sum_{i=1}^{2} X^i \cdot E_i$
    and $Y = \sum_{i=1}^{2} Y^i \cdot E_i$ in the local base E.
Let $\Gamma^{II} = (\Gamma^1, \Gamma^2)$ be the CHRISTOFFEL tensor coefficients of 2nd kind.

Then we define – using EINSTEIN summation convention! – the *covariant derivation of Y w.r.t. X*, noted $\nabla_X Y$ by

$$\nabla_X Y := X^i \frac{\partial Y^j}{\partial x^i} E_j + X^i Y^j \Gamma_{ij}^k E_k \tag{11.1}$$

or more explicit by decoding the EINSTEIN summation convention

$$\nabla_X Y := \left(X_u \frac{\partial Y_u}{\partial u} + X_v \frac{\partial Y_u}{\partial v} + \sum_{i,j=1}^{2} X_i Y_j \Gamma_{ij}^u \right) E_u + \left(X_u \frac{\partial Y_v}{\partial u} + X_v \frac{\partial Y_v}{\partial v} + \sum_{i,j=1}^{2} X_i Y_j \Gamma_{ij}^v \right) E_v \tag{11.2}$$

Remark. 1. Formula (11.1) is given e.g. in OLOFF [22, p.91] or THORPE [36, p.229].
2. Formula (11.2) is given in ▷ Connexion de Levi-Civita.
3. In (11.2) the symbol $\sum_{i,j=1}^{2} \cdots \equiv \sum_{i=1}^{2} \sum_{j=1}^{2} \ldots$ means a double summation.

11.2 Implementation and Examples

In the following examples we calculate the two coefficients of the covariant derivative in (10.2) and then associate the results to E_1 resp. E_2. We demonstrate this with three different surfaces and different vector fields.

Example 22. (covariant derivative of two vector fields on $z = uv$)
We consider the function surface $f(u,v) = u \cdot v$ and its parametrization ϕ.

```
# EIGENMATH
-- HYPERBOLIC PARABOLOID
phi(u,v) = (u , v, u*v)          -- S=phi(U)

U = (u,v)

g(i,j) = dot(d(phi, U[i]), d(phi,U[j]))
g = ((g(1,1),g(1,2)),
     (g(2,1),g(2,2)) )
g

-- base in TpS
Eu = d(phi,u)
Eu
Ev = d(phi,v)
Ev

gu = inv(g)
gu

Dg = d(g, U)    -- derivative of metric tensor
Dg

GammaI = 1/2 (Dg + transpose(Dg,2,3) - transpose(Dg,2,3,1,2))
GammaI

GammaII = dot(gu, GammaI)
GammaII

-- DEFINITION of covariant derivative via (10.2)
---------------------------------------------  (0)
covD(Y,X)= ( X[1]*d(Y[1],u) + X[2]*d(Y[1],v) +
             sum(i,1,2, sum(j,1,2, X[i]*Y[j]*GammaII[1,i,j])),  -- *Eu
             X[1]*d(Y[2],u)+X[2]*d(Y[2],v) +
             sum(i,1,2, sum(j,1,2, X[i]*Y[j]*GammaII[2,i,j])) ) -- *Ev
---------------------------------------------

"four example vector fields on the hyperbolic paraboloid"  --(1)
```

```
X  = (u,v^3)        -- = Xu*Eu +Xv*Ev =  u*Eu +   v^3*Ev
Y  = (u v, u+v)     -- = Yu*Eu +Yv*Ev = uv*Eu + (u+v)*Ev
D1 = (1,0)          -- i.e. D1 = 1*Eu + 0*Ev = Eu in coordinates
D2 = (0,1)

covD(Y,X)           --(2)
covD(D1,D1)
covD(D1,D2)         --(3)
covD(D2,D2)
```

EIGENMATH output:

$$\begin{bmatrix} \dfrac{u^2 v}{u^2 + v^2 + 1} + \dfrac{u v^5}{u^2 + v^2 + 1} + u v^3 + \dfrac{u v^2}{u^2 + v^2 + 1} + u v \\[3mm] \dfrac{u^3}{u^2 + v^2 + 1} + \dfrac{u^2 v^4}{u^2 + v^2 + 1} + \dfrac{u^2 v}{u^2 + v^2 + 1} + u + v^3 \end{bmatrix} \qquad \begin{bmatrix} \dfrac{v}{u^2 + v^2 + 1} \\[3mm] \dfrac{u}{u^2 + v^2 + 1} \end{bmatrix}$$

Comment. The definition of the covariant derivative follows 1:1 the mathematical definition in (10.2), but we limit us to only calculate the pair of coefficients. In (1) we define four different vector fields and calculate their covariant derivatives. The output displays on the LHS the result of $\nabla_X Y$ and on the RHS the result of $\nabla_{E_u} E_v$.

● ▷ Click here to RUN the example.

Example 23. (covariant derivative of two vector fields on the helicoid)
Consider the parametric surface of the helicoid.

```
# EIGENMATH
-- HELICOID

phi(u,v) = (u cos(v), u sin(v), c v)
U =(u,v)   -- coordinates of surface

"metric tensor g"
g = zero(2,2)
    g[1,1] = dot(d(phi,u), d(phi,u))
    g[1,1]   -- c^2+s^2=1                      (1)
    g[1,1] = 1
    g[1,2] = dot(d(phi,u), d(phi,v))
    g[2,1] = g[1,2]
    g[2,2] = dot(d(phi,v), d(phi,v))
    g[2,2]   -- (c^2+u^2).(cos(v)^2+sin(v)^2)     (2)
    g[2,2] = c^2 + u^2
g

"base in TpS"
Eu = d(phi,u)
```

```
Eu                     -- display Eu
Ev = d(phi,v)
Ev

gu = inv(g)
gu

Dg = d(g, U)    -- derivative of metric tensor
Dg

GammaI = 1/2 (Dg + transpose(Dg,2,3) - transpose(Dg,2,3,1,2))
GammaI          -- display GammaI

GammaII = dot(gu, GammaI)
GammaII         -- display GammaII

-- DEFINITION of covariant derivative via (10.2)
-- using GammaII
-----------------------------------------------
covD(Y,X)= ( X[1]*d(Y[1],u) + X[2]*d(Y[1],v) +
              sum(i,1,2, sum(j,1,2, X[i]*Y[j]*GammaII[1,i,j])),
              X[1]*d(Y[2],u)+X[2]*d(Y[2],v) +
              sum(i,1,2, sum(j,1,2, X[i]*Y[j]*GammaII[2,i,j])) )
-----------------------------------------------

X = (u^2,v^2)    --i.e.   .. =  u^2*Eu +  v^2*Ev
Y = (u-v, u+v)  --i.e.   .. = (u-v)*Eu +(u+v)*Ev

covD(Y,X)
```

EIGENMATH output:

$$\left[\begin{array}{c} u^2 - u^2 v^2 - u v^3 - v^2 \\ \dfrac{u^4}{c^2 + u^2} + \dfrac{u^3 v}{c^2 + u^2} + u^2 + \dfrac{u^2 v^2}{c^2 + u^2} - \dfrac{u v^3}{c^2 + u^2} + v^2 \end{array} \right]$$

Comment. To have a simplified output we help EIGENMATH in (1) and (2) to simplify the coefficients g_{11} and g_{22}. We use the well-known trigonometric Pythagoras formula. The output displays the result of $\nabla_X Y$.

● ▷ Click here to RUN the example.

Example 24. (covariant derivative of two vector fields on the sphere)
We consider the parametric sphere of radius R. The aim of this example is to verify the covariant derivatives of the coordinate vector fields $E_1 = E_u \equiv \partial_\theta$ and $E_2 = E_v \equiv \partial_\varphi$, whose results are given in OLOFF [22, p.92; p.95].

```
# EIGENMATH
-- A SPHERE OF RADIUS R
phi(u,v) = (R sin(u) cos(v), R sin(u) sin(v), R cos(u))
--          x                 y                z


U = (u,v)

Eu = d(phi,u)
Eu
Ev = d(phi,v)
Ev

"metric tensor g"
g = zero(2,2)
      g[1,1] = dot(d(phi,u), d(phi,u))
      g[1,1] = R^2            -- simplified by mind
      g[1,2] = dot(d(phi,u), d(phi,v))
      g[2,1] = g[1,2]
      g[2,2] = dot(d(phi,v), d(phi,v))
      g[2,2] = R^2 sin(u)^2  -- simplified by mind
g

gu = inv(g)
gu

Dg = d(g, U)
Dg

GammaI = 1/2 (Dg + transpose(Dg,2,3) - transpose(Dg,2,3,1,2))
GammaI

GammaII = dot(gu, GammaI)
GammaII

------------------------------------------------- COVARIANT DERIVATIVE
covD(Y,X)= ( X[1]*d(Y[1],u) + X[2]*d(Y[1],v) +
               sum(i,1,2, sum(j,1,2, X[i]*Y[j]*GammaII[1,i,j])),
               X[1]*d(Y[2],u)+X[2]*d(Y[2],v) +
               sum(i,1,2, sum(j,1,2, X[i]*Y[j]*GammaII[2,i,j])) )
-------------------------------------------------
```

```
D1 = (1,0)        -- Eu
D2 = (0,1)        -- Ev

covD(D1,D1)       --(1)
covD(D1,D2)       --(2)
covD(D2,D2)       --(3)
```

● ▷ Click here to RUN the example.
EIGENMATH output:

$$\begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 \\ \dfrac{\cos(u)}{\sin(u)} \end{bmatrix}$$

$$\begin{bmatrix} -\cos(u)\sin(u) \\ 0 \end{bmatrix}$$

Comment. We interpret the EIGENMATH output.
The output associated with line (1) means

$$\texttt{covD(D1,D1)} \ = \ \nabla_{D1}D1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \equiv 0 \cdot E_1 + 0 \cdot E_2$$

$$\equiv \ 0 \cdot \partial_\vartheta + 0 \cdot \partial_\varphi = \nabla_{\partial_\vartheta} \partial_\vartheta \big\|^{\text{OLOFF}} = 0$$

The output associated with line (2) means

$$\texttt{covD(D1,D2)} \ = \ \nabla_{D1}D2 = \begin{bmatrix} 0 \\ \cos(u)/\sin(u) \end{bmatrix}$$

$$\equiv \ 0 \cdot E_1 + \frac{\cos(u)}{\sin(u)} \cdot E_2 \equiv 0 \cdot \partial_\vartheta + \cot(\varphi) \cdot \partial_\varphi$$

$$= \ \nabla_{\partial_\vartheta} \partial_\varphi \big\|^{\text{OLOFF}} = \cot(\varphi) \cdot \partial_\varphi$$

The output associated with line (3) means

$$\texttt{covD(D2,D2)} \ = \ \nabla_{D2}D2 = \begin{bmatrix} -\cos(u)\sin(u) \\ 0 \end{bmatrix} \equiv -\cos(u)\sin(u) \cdot E_1 + 0 \cdot E_2$$

$$\equiv \ -\cos(\vartheta)\sin(\vartheta) \cdot \partial_\vartheta + 0 \cdot \partial_\varphi = \nabla_{\partial_\varphi} \partial_\varphi \big\|^{\text{OLOFF}} = -\cos(\vartheta)\sin(\vartheta) \cdot \partial_\vartheta$$

These results for the covariant derivatives of the coordinate vector fields on the sphere agree with the results given by OLOFF [22, p.92; p.95].

11.3   Exercise

Exercise 176. (the directional derivative)

In the figure 14 we gave a first visual impression of what the difference of the covariant derivative of two vector fields and their directional derivative is.

OLOFF [22, p.88] gives the following definition of the *directional derivative $D_X Y$* of two vector fields on a parametrized surface $\phi \colon U \to \mathbb{R}^3$ with $\partial_1 \equiv \phi_u \equiv E_u, \partial_2 \equiv \phi_v \equiv E_v$:

$$D_X Y \overset{\text{OLOFF}}{:=} X^i \partial_i Y^k \partial_k \tag{11.3}$$

a. Interpret formula (10.3) using our E_u, E_v instead of ∂_1, ∂_2.
Be warned: (10.3) is decoded by means of the EINSTEIN summation convention.
b. Implement $D_X Y$ as function `dirD(Y,X)` in EIGENMATH and calculate the directional derivatives of the vector fields of the forgoing examples.
c. THORPE [36, p.55] gives an alternative geometric definition of the covariant derivative as the tangential component of the directional derivative, i.e. as orthogonal projection of the directional derivative onto the tangential space $T_p S$ of the surface:

$$\nabla_X Y := D_X Y - (D_X Y \bullet N) \cdot N \tag{11.4}$$

where N is the unit normal vector (field) of the surface, see also HITCHIN [14, p.64] or SHIFRIN [32, p.67]
Implement definition (11.4) in EIGENMATH and reproduce some results of the examples of this chapter.

Remark. Definition (11.4) uses the normal vector and is therefore an extrinsic concept. The definition (11.2) uses only the 1st fundamental form and the CHRISTOFFEL tensor and belongs to intrinsic geometry.

● *Consider the following parametric surfaces. Task: Calculate the covariant derivatives $\nabla_X Y$ of the vector fields $X = (u^2, v^2) \equiv u^2 * E_u + v^2 * E_v$ and $Y = (u - v, u + v) \equiv (u - v) * E_u + (u + v) * E_v$ on the respective surface ϕ using EIGENMATH ...*

Exercise 177. ... the cylinder $\phi(u, v) = (a \cos(v), a \sin(v), u)$. Choose $a = 1$.

Exercise 178. the cone $\phi(u, v) = (u \cos(v), u \sin(v), u)$.

Exercise 179. the hyperbolic paraboloid $\phi(u, v) = (u, v, u^2 - v^2)$.

Exercise 180. the monkey saddle $\phi(u, v) = (u, v, u^3 - 3uv^2)$.

Exercise 181. the function graph $\phi(u, v) = (u, v, f(u, v))$.

Exercise 182. the surface of revolution $\phi(u, v) = (f(u) \cos(v), f(u) \sin(v), g(u))$.

Exercise 183. the PLÜCKER conoid $\phi(u, v) = (v \cdot \cos(u), v \cdot \sin(u), 2 \cos(u) \cdot \sin(u))$.

Exercise 184. the SCHERK surface $\phi(u, v) = (u, v, \ln(\frac{\cos v}{\cos u}))$.

Exercise 185. the hexenhut $\phi(u, v) = (\alpha/\sqrt{u} \cdot \cos v, \alpha/\sqrt{u} \cdot \sin v, u)$ where $\alpha^2 = \frac{2}{3\sqrt{3}}$.

Exercise 186. the 'hexe' surface $\phi(u, v) = (\sqrt{u} \cdot \cos v, \sqrt{u} \cdot \sin v, u)$.

Exercise 187. Do some more covariant derivatives of vector fields of your choice on surfaces you are interested in.

$\heartsuit$

This ends our first steps into the Elementary Differential Geometry using EIGENMATH as our companion.

12   Appendix: source code of diffgeoBox

This is a collection of the relevant definitions from the booklet. This file could be loaded to do calculations with tensors using EIGENMATH.

```
############### (2023) Dr.W.G. Lindner,  Leichlingen DE
###   diffgeoBox       Differential Geometry box
###############

-- ======================== Pattern to calculate
--          K,H,kappa,Christoffel,Riemann,Ricci,R
-- ------------------------ of parametization phi

phi(u,v) = (u,v, u+v)      -- parametrization, please adjust 3rd component
phiu = d(phi,u)            -- tangential space basis
phiv = d(phi,v)
U    = (u,v)               -- coordinates on TpS

                           -- surface metric tensor
g(i,j) = dot(d(phi, U[i]), d(phi,U[j]))
g      = ((g(1,1),g(1,2)), (g(2,1),g(2,2)))
g
do( E=g[1,1], F=g[1,2], G=g[2,2])   -- classic names for g

gu = inv(g)                -- inverse of g
V  = 1/sqrt(det(g))        -- volume element
N = cross(phiu, phiv)      -- surface normal vector field
N
                           -- 2nd fund. form
h(i,j) = 1/V*dot( cross(d(phi,u), d(phi,v)), d(phi,U[i],U[j]))
h      = ( (h(1,1),h(1,2)), (h(2,1),h(2,2)))
h
do( L=h[1,1], M=h[1,2], N=h[2,2])   -- classic names for h

A = dot(gu, h)             -- shape operator
A
K = det(A)                 -- GAUSS curvature
K
H = contract(A)            -- mean curvature
H

Dg = d(g, U)               -- partial derivatives

-- ------------------------ BEGIN of code by G. Weigt
```

```
GammaI = 1/2 (Dg + transpose(Dg,2,3) - transpose(Dg,2,3,1,2))
GammaII = dot(gu, GammaI)
GammaII

T1 = d(GammaII, U)
T2 = dot(transpose(GammaII,2,3), GammaII)
RII = transpose(T1,3,4) -T1+ transpose(T2,2,3) - transpose(T2,2,3,3,4)
RII                      -- RIEMANN tensor Ri^jkl 2nd kind

RI = dot(g, RII)         -- RIEMANN tensor Rijkl 1st kind
RI

Ric = contract(RII,1,3)  -- RICCI tensor
Ric
                         -- RICCI curvature scalar
R = contract( dot( gu, transpose(Ric)))
R

-- ----------------------- END of code by G. Weigt

-- CHRISTOFFEL symbols coefficients
gammaI(i,j,k) = 1/2 * ( d(g[j,k],U[i]) +
                        d(g[i,k],U[j]) - d(g[i,j],U[k]) )
--gammaI(1,2,2)

gammaII(k,l,m) = sum(i,1,2, gu[k,i]*1/2 *
                ( d(g[m,i],U[l]) + d(g[l,i],U[m]) - d(g[m,l],U[i]) ))
--gammaII(1,2,2)

Kintrinsic(u,v)= do(      -- intrinsic curvature
    Ch111 = gammaII(1,1,1),
    Ch121 = gammaII(1,2,1),
    Ch211 = gammaII(2,1,1),
    Ch221 = gammaII(2,2,1),
    Ch222 = gammaII(2,2,2),
    Kintrinsic1 = 1/g[1,1]*(
                    d(Ch211,v) - d(Ch221,u) +
                    Ch221*(Ch111-Ch221) + Ch211*(Ch222-Ch121)),
    Kintrinsic1)


-- RIEMANN curvature Rijkl coefficients
RiemI(b,m,s,q) = d(gammaI(b,m,q),U[s]) - d(gammaI(b,m,s),U[q]) +
                    sum(n,1,2, gammaI(b,n,s)*gammaII(n,m,q) ) -
                    sum(n,1,2, gammaI(b,n,q)*gammaII(n,m,s) )
--RiemI(1,2,1,2)
```

```
-- RIEMANN curvature Ri^jkl coefficients
RiemII(b,m,s,q) = d(gammaII(b,m,q),U[s]) - d(gammaII(b,m,s),U[q]) +
                       sum(n,1,2, gammaII(b,n,s)*gammaII(n,m,q) )    -
                       sum(n,1,2, gammaII(b,n,q)*gammaII(n,m,s) )
--RiemII(1,2,1,2)

-- RICCI tensor coefficients
Ricci(m,q) = sum( s,1,2, RiemII(s,m,s,q))
--Ricci(1,2)

--RICCI curvature scalar
Rcs  = sum( i,1,2, sum(j,1,2, gu[i,j]*Ricci(i,j) ))
--Rcs

---- DEFINITION of covariant derivative
covD(Y,X)= ( X[1]*d(Y[1],u) + X[2]*d(Y[1],v) +
             sum(i,1,2, sum(j,1,2, X[i]*Y[j]*GammaII[1,i,j])),
             X[1]*d(Y[2],u)+X[2]*d(Y[2],v) +
             sum(i,1,2, sum(j,1,2, X[i]*Y[j]*GammaII[2,i,j])) )
```

• ▷ Click here to GET the code.

13   Bibliography

References

[1] BÄR, C. (2013): *Differential Geometry.*
url: https://www.math.uni-potsdam.de/fileadmin/user_upload/Prof-Geometrie/Dokumente/Lehre/Lehrmaterialien/skript-DiffGeo-engl.pdf

[2] BANCHOFF, T. F. & LOVETT, S. (^{3}2022): *Differential Geometry of Curves and Surfaces.* London: Chapman and Hall.

[3] CARMESIN, H.-O. (2012): *Schüler entdecken die Einstein-Geometrie mit dem Beschleunigungssensor.* url: http://hans-otto.carmesin.org/images/Physikdid/2012Einsteingeometrie_In_PhyDid_B.pdf

[4] CASEY, J. (1996): *Exploring Curvature.* Braunschweig: Vieweg.

[5] COXETER, H. S. M. (1963): *Unvergängliche Geometrie.* Basel: Birkhäuser.

[6] DESERNO, M. (2004): *Notes on Differential Geometry.*
url: https://www.cmu.edu/biolphys/deserno/pdf/diff_geom.pdf

[7] DINEEN, S. (^{2}2001): *Multivariate Calculus and Geometry.* London: Springer.

[8] EARL, M. (2004): *The Local Theory of Curves and Surfaces.*
url: https://courses-archive.maths.ox.ac.uk/node/view_material/48844

[9] GALLOWAY, ?. (2022): *Introduction to Differential Geometry.*
url: https://www.math.miami.edu/~galloway/IntroDGnotes.pdf

[10] GLUCK, H. (2022): *Intrinsic Geometry of Surfaces.*
url: https://www2.math.upenn.edu/~shiydong/Math501X-4-IntrinsicGeometry.pdf

[11] GRINFELD, P. (2013): *Introduction to Tensor Analysis and the Calculus of Moving Surfaces.* New York: Springer.

[12] HERTRICH-JEROMIN, U. (2019): *Differential Geometry.*
url: https://www.geometrie.tuwien.ac.at/hertrich-jeromin/pub/dg.pdf

[13] HECKMANN, G. (2019): *Classical Differential Geometry.*
url: https://www.math.ru.nl/~heckman/CDG.pdf

[14] HITCHIN, N. (2019): *Geometry of Surfaces.*
url: https://people.maths.ox.ac.uk/~joyce/Nairobi2019/Hitchin-GeometryOfSurfaces.pdf

[15] KREYSZIG, E. (1957): *Differentialgeometrie.* Leipzig: Akademische Verlagsgesellschaft Geest & Portig.

[16] LINDNER, W. (2021): *Some Tapas of Tensors.*
url: https://lindnerdrwg.github.io/Some-Tapas-of-Tensors.pdf

[17] LINDNER, W. (2023): *Elementary Differential Geometry using* EIGENMATH *– solutions manual.* url: https://lindnerdrwg.github.io/Elementary-Differential-Geometry-of-Surfaces-solutions.pdf

[18] LINDNER, W. (2021): *Selected Solutions to Bachman's Differential Forms.*
url: https://lindnerdrwg.github.io/Bachman-Differential-Forms-Solutions-selected.pdf

[19] LIPSCHUTZ, M. M. (1969): *Differential Geometry.* New York: McGraw-Hill

[20] MALKOWSKY, E. & NICKEL, W. (1993): *Computergrafik in der Differentialgeometrie.* Braunschweig: Vieweg.

[21] MARSDEN, J. & WEINSTEIN, A. (^{2}1985): *Calculus III.* New York: Springer.

[22] OLOFF, R. (^{6}2018): *Geometrie der Raumzeit.* [engl.: Geometry of Spacetime.]
Berlin: Springer Spectrum.

[23] PRESSLEY, A. (2001): *Elementary Differential Geometry.* London: Springer.

[24] RECKZIEGEL, H. ET AL. (1998): *Elementare Differentialgeometrie mit Maple.*
Braunschweig: Vieweg.

[25] REJBRAND, A. (2023): *The Rejbrand Encyclopaedia of Curves and Surfaces.*
url:https://trecs.se/surfaces.php

[26] SEEBURGER, P. (2018): *CalcPlot3D app.*
url:https://c3d.libretexts.org/CalcPlot3D/index.html

[27] SEEBURGER, P. (2018): *CalcPlot3D Manual.*
url: https://c3d.libretexts.org/CalcPlot3D/CalcPlot3D-Help/front.html

[28] SEEBURGER, P. (2018): *CalcPlot3D - Introduction Video.*
url:https://c3d.libretexts.org/CalcPlot3D/CalcPlot3D-Help/section-1.html

[29] SHIFRIN, T. (2018): *Differential Geometry: A First Course in Curves and Surfaces.*
url:https://math.franklin.uga.edu/sites/default/files/inline-files/ShifrinDiffGeo.pdf

[30] SNYGG, J. (2001): *A New Approach to Differential Geometry using Clifford's Geometric Algebra.* New York: Springer-Birkhäuser.

[31] SOCHI, T. (2017): *Introduction to Differential Geometry of Space Curves and Surfaces.* London: CreativeSpace. ISBN 9781546735892.

[32] SOFI, G. M. (2018): *Fundamental Theorem of Surfaces in* $\mathbb{R}^3$.
url:https://www.cukashmir.ac.in/cukashmir/User_Files/image/module203_MMT_C204.pdf

[33] SPIEGEL, M. R. (1963): *Advanced Calculus.*
New York: McGraw-Hill (Schaums' Outline Series)

[34] STEEB, W.-H & LEWIEN, D. (1991): *Algorithms and Computation with* REDUCE. Mannheim: Bibliographisches Institut.

[35] STEEB, W.-H & SHI, T. K. (1998): *Symbolic C++: An Introduction to Computer Algebra Using Object-Oriented Programming.* Singapore: Springer.

[36] THORPE, J. A. (1979): *Elementary Topics in Differential Geometry.* New York: Springer.

[37] VAS, L. (2021): *Surfaces I.* url: `https://liavas.net/courses/math430/files/Surfaces.pdf`

[38] VAS, L. (2021): *Surfaces II.* url: `https://liavas.net/courses/math430/files/Surfaces_part2.pdf`

[39] VAS, L. (2021): *Surfaces III.* url: `https://liavas.net/courses/math430/files/Surfaces_part3.pdf`

[40] WEIGT, G. (2021): EIGENMATH *Homepage.* url: `https://georgeweigt.github.io`

[41] WEIGT, G. (2021): EIGENMATH *online.* url: `https://georgeweigt.github.io/eigenmath-demo.html`

[42] WEIGT, G. (2021): *The Schwarzschild Metric.* url: `https://georgeweigt.github.io/schwarzschild-metric.html`

[43] WHEELER, N. (2015): *Differential Geometry of Some SURFACES IN 3-SPACE.* url: `https://www.reed.edu/physics/faculty/wheeler/documents/Miscellaneous%20Math/Differential%20Geometry/A.%20Surfaces%20in%203-Space.pdf`

[44] WOLFRAM mathworld: Catenoid. url: `https://mathworld.wolfram.com/Catenoid.html`

[45] WOLFRAM mathworld: Cone. url: `https://mathworld.wolfram.com/Cone.html`

[46] WOLFRAM mathworld: Cylinder. url: `https://mathworld.wolfram.com/Cylinder.html`

[47] WOLFRAM mathworld: ENNEPER's Surface. url: `https://mathworld.wolfram.com/EnnepersMinimalSurface.html`

[48] WOLFRAM mathworld: Helicoid. url: `https://mathworld.wolfram.com/Helicoid.html`

[49] WOLFRAM mathworld: MOEBIUS Strip. url: `https://mathworld.wolfram.com/Catenoid.html`

[50] WOLFRAM mathworld: PLÜCKER's conoid.
url: https://mathworld.wolfram.com/PlueckersConoid.html

[51] WOLFRAM mathworld: SCHERK's Surface.
url: https://mathworld.wolfram.com/ScherksMinimalSurfaces.html

[52] WOLFRAM mathworld: Sphere.
url: https://mathworld.wolfram.com/Sphere.html