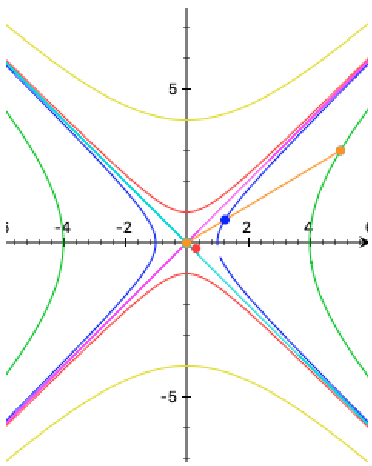
Exploring Math with EIGENMATH

Geometric Algebra *Interactive*! with Eigenmath

Complex, Hyperbolic and Geometric Algebra Numbers

$\mathbb{C} \quad \mathbb{H} \quad \mathbb{G}$



Dr. Wolfgang Lindner
`LindnerW@t-online.de`
Leichlingen, Germany
2021

Contents

Preface

This is part 5 of a series of booklets, which want to introduce the reader to some topics of (Multi)Linear Algebra and at the same time into the use of CAS Eigenmath.

About the content of the booklet

The *first* chapter constructs the well-known *complex* numbers $\mathbb{C}$ as an algebraic structure using a 2D basis and a 2×2 multiplication table for it. Alternatively we compute in $\mathbb{C}$ as a 2D Clifford algebra $c\ell(2,0)$ using the package `EVA` for the first time.
The *second* chapter introduces the not so known *hyperbolic* numbers $\mathbb{I}\!\mathrm{H}$. We realize these numbers via an algebraic structure (a 2D basis and a 2×2 multiplication table) and also model $\mathbb{I}\!\mathrm{H}$ by means of the 2D Clifford algebra $c\ell(1,1)$ using the package `EVA` for the second time. This chapter will also serve as an CAS Eigenmath companion to the article [12] and the book [13, Ch. 1] by Garret Sobczyk.
The *third* chapter deals with the quaternions $\mathbb{H}$. We construct these numbers again in tow ways: as an algebraic structure (a 4D basis and a 4×4 multiplication table) and
by means of a 4D Clifford algebra $c\ell(3)^+$ using the package `EVA`.
The *fourth* chapter abstracts the foregoing examples of number field constructions to the concept of a Geometric Algebra and demonstrates the generalizing use of it in plane and space geometry. This chapter will also serve as an CAS Eigenmath companion to the books of Sobczyk[12] and Macdonald [8]. The first book does not use any CAS and the second make use of the not so simple Python package `galgebra`.
In both cases Eigenmath should make things easier for the beginner.

Eigenmath

Eigenmath is a computer algebra system that can be used to solve problems in mathematics and in the natural and engineering sciences. It is a personal resource for students, teachers and scientists. Eigenmath is small, compact, capable and free. It runs best on MacOS or as Online tool in your browser.
The considerations in this script would be difficult to elementize without the use of a computer algebra system like Eigenmath, because heavy calculations of new kind of products occur in the conceptual constructions. Therefore, in Eigenmath sessions we explore decisive phenomena or verify or falsify hypotheses. We encourage the dialogical practice in CAS language communication with the Eigenmath assistance. If possible, all CAS dialog sequences - which are shown in `typewriter font` - should be performed live on the computer. We give therefore many lively links to invocable Eigenmath scripts that may be modified or amended by the user.

The booklet make full use of the Eigenmath package `EVA2.txt`, which was written by Bernard Eyheramendy [5]. Without his work this booklet would had been not possible. `EVA2.txt` itself is a fine opportunity to study programming in Eigenmath and the infos and tutorials to be find at his homepage deserve your interest.

The EIGENMATH routines, which are especially written for this booklet, are collected in four toolboxes `cBox, hyBox, qBox, qcBox` for the convenience of the user and are invoked e.g. by the command `run("cBox.txt")` in a running EIGENMATH Online[1] session. These CAS functions wish to train algorithmic oriented constructive thinking. The EIGENMATH commands used and the textual representation should be elementary enough to serve as a good companion while reading basic or advanced courses on Linear Algebra. They may also serve as a help system for independent individual work.

To use this booklet interactively

The social-constructivist APOS[2] learning theory was in my mind throughout the construction of this booklet. Compared to classic learning theories, the APOS theory focuses on the finding that *the mental (re)construction process of mathematical knowledge is decisively promoted by a mathematically oriented programming language as a medium* in which the knowledge constructions are represented as programming constructs (DUBINSKY). So the learning process is triggered by *actions* or manipulations on mental or virtual CAS objects. Using this booklet

... *you do not need to install any software to do the calculations.* The CAS EIGENMATH works directly out of this text, on any operating system, on every hardware (Smartphone, iPhone, tablet, PC, etc.), at any place: you only must be online and click on a link like ▷ *Click here to invoke* EIGENMATH (◁ *please click here! Really!*). From this point on you can run a given script or fork with own computations.

... *you do not need to install any software to produce quality plots interactively.* You only must be online to press a link like `CalcPlot3D` (◁ *please click here! Really!*) in this script. At this point you can make a 2D/3D–plot to visualize a concept or to make a calculation visually evident.

I thank George WEIGT for his friendly support, hints and help regarding his EIGENMATH. So it was a real pleasure to write down these notes.

Any feedback from the user is very welcome.

Wolfgang Lindner

Leichlingen, Germany
February 2021
dr.w.g.Lindner@gmail.com

[1]Running the EIGENMATH app on the iMac this command has to be substituted through `run("Downloads/cBox.txt")`. The file `cBox.txt` has therefore to be copied to the 'downloads' folder.

[2]see ARNON et. al. [1]

1 $\mathbb{C}$ – the complex numbers

It is well known that the solution set $\mathbb{L}$ of a *singular homogeneous* 3×3 linear system is often a *straight line* or a *plane through the origin*. The solution set $\mathbb{L}$ is not just a *subset* of the surrounding space $\mathbb{R}^n$, but also has a *linear structure*: with each two solution vectors $\vec{v}$ or $\vec{w}$ in $\mathbb{L}$ there are also all linear combinations $r\cdot\vec{v}+s\cdot\vec{w}$ (with $r,s\in\mathbb{R}$) solutions again.[3] Therefore, this property is particularly emphasized in a central concept of linear algebra.

1.1 $\mathbb{C}$ as vectorspace

We already know the complex numbers $\mathbb{C}$: the arithmetical playground (the 'underlying set') of $\mathbb{C}$ is the well known Euclidean plane $\mathbb{R}^2$ with the two operations of addition and forming 'multiples' of column/row vectors (a,b), i.e. $\mathbb{C}\sim\mathbb{R}^2$ or more precisely

$$\begin{aligned}
\mathbb{C} &\simeq (\mathbb{R}^2,+,\cdot) && \text{with the rules} && (1.1)\\
(a,b)+(c,d) &\overset{def}{=} (a+c,b+d) && && (1.2)\\
r\cdot(a,b) &\overset{def}{=} (r\cdot a, r\cdot b) && \text{for arbitrary } a,b,c,d,r\in\mathbb{R} && (1.3)
\end{aligned}$$

For example $(1,2)+(3,4)=(4,6)$ and $0.5\cdot(-2,2)=(-1,1)$.
Equipped with these two operations the set $\mathbb{C}$ is an "*2–dimensional vector space over the reals*", i.e. the operations $+$ and $\cdot$ respect the following rules of an abstract vector space.

Definition. Let V be a set on which there are defined two operations, one called *addition* ('+') and the other called *multiplication by scalars* ('·'). If the following 10 calculation rules ('laws', 'axioms') holds, $V\equiv(V,+,\cdot)$ is called a *vector space*:

For all $\vec{u},\vec{v},\vec{w}\in V$ and $r,s\in\mathbb{R}$ we have

($\uplus$)	$\vec{v}+\vec{w}$	$\in$	V	Closedness
(C)	$\vec{v}+\vec{w}$	$=$	$\vec{w}+\vec{v}$	Commutativity
(A)	$(\vec{u}+\vec{v})+\vec{w}$	$=$	$\vec{u}+(\vec{v}+\vec{w})$	Associativity
(N)	$\vec{v}+\vec{0}$	$=$	$\vec{v}$	there exists such an $^{+}$Neutral element $\vec{0}\in V$
(I)	$\vec{v}+(-\vec{v})$	$=$	$\vec{0}$	there exists the Invers element $-\vec{v}\in V$ for every $\vec{v}$
($\dot\cup$)	$r\cdot\vec{v}$	$\in$	V	Closedness
(1)	$r\cdot(\vec{v}+\vec{w})$	$=$	$r\cdot\vec{v}+r\cdot\vec{w}$	Distributivity I
(2)	$(r+s)\cdot\vec{v}$	$=$	$r\cdot\vec{v}+s\cdot\vec{v}$	Distributivity II
(3)	$r\cdot(s\cdot\vec{v})$	$=$	$(rs)\cdot\vec{v}$	Distributivity III
(4)	$1\cdot\vec{v}$	$=$	$\vec{v}$	there exists such an $^{\cdot}$Neutral element 1

[3]This property did not apply to inhomogeous linear system!

Exercise 1.1. **Mental model of a vector space.**

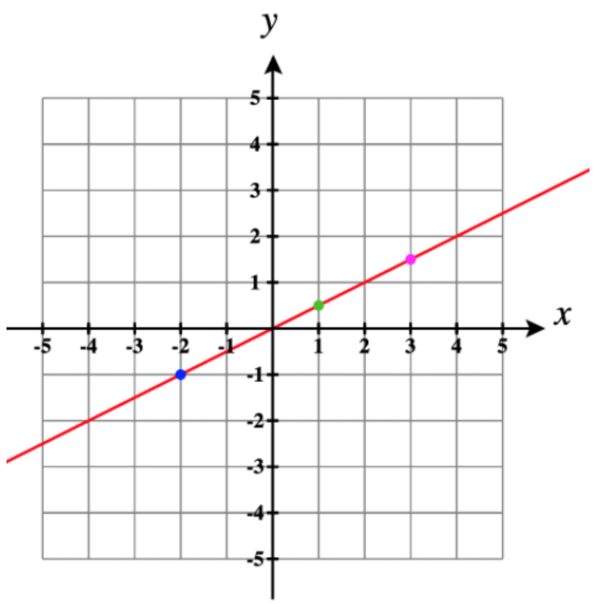


Figure 1: Red: the vectorspace $\mathbb{L}$ of solutions of $0.5x - y = 0$.
Blue: the vector $\vec{u} = [-2, -1]$.
Green: the vector $\vec{v} = [1, 0.5]$.
Magenta: the vector $w = -1 \cdot \vec{u} + 1 \cdot \vec{v} = [3, 1.5]$.

a. Verify that $\vec{u}, \vec{v}$ and $\vec{u} + \vec{v}$ from Fig.1 are solutions of $0.5x - y = 0$, i.e. $\vec{u}, \vec{v} \in \mathbb{L}$.
b. Verify that $\vec{b} = [2t, t], t \in \mathbb{R}$ is a general solution vector in $\mathbb{L}$.
c. Verify that arbitrary multiples of $\vec{b}$ are in $\mathbb{L}$, i.e. $r \cdot \vec{b} \in \mathbb{L}$ for arbitray $r \in \mathbb{R}$.
d. Verify: The set $\mathbb{L}$ is a 1–dimensional vector space over $\mathbb{R}$ with basis $\{(2, 1)\}$.
That is: the 10 vector space conditions $\uplus$**CANI**$\dot\cup$**1234** are fulfilled for $\mathbb{L}$.
If it is tidy to do these tests by paper'n pencil, please use EIGENMATH: ▷ *Click here.*
♡ Keep e.g. this model in mind when thinking at the concept of a vector space.

Remark.

1. The first group of rules **(C)**, **(A)**, **(N)**, **(I)** for the *vector addition* are the **C***ommutat ive* law, the **A***ssociative* law, the law of the existence of a **N***eutral* element and the law of the existence of **I***nvers* elements. ($\uplus$) resp. ($\dot\cup$) is the so-called *closeness* of the addition resp. multiple forming, i.e. with each pair of vectors their sum resp. multiple lies in the vector space again.

   (N) and **(I)** do not go without saying:

   ◦ **(N)** says more precisely: there is a certain element in V - which is denoted by $\vec{0}$ and called the *zero vector* - with the property that $\vec{v} + \vec{0} = \vec{v}$ applies to any $\vec{v}$.

- ∘ **(I)** says more precisely: for every arbitrary $\vec{v}$ from V there is an element - which is denoted by $-\vec{v}$ and is called *opposite vector* or *inverse element* - in V with the property: $\vec{v} + (-\vec{v}) = \vec{0}$.

2. The second group of calculation rules **(1)**, **(2)**, **(3)**, **(4)** describes the formation of multiples of vectors, i.e. the multiplication of vectors with real numbers. These rules describe the distribution of numbers on vectors under '·' and therefore are called the four *distributive laws*.

3. The 10 rules $\uplus$**CANI**$\dot{\cup}$**1234** are also called the *axioms* of a vector space.

Exercise 1.2. The arithmetic rules of build–in ℂ.
Verify the 10 vector space axioms $\uplus$**CANI**$\dot{\cup}$**1234** for ℂ, the *build–in complex number system* of Eigenmath. Here is a start: ▷*Click here.*

1.2 ℂ as algebra

To reconstruct the complex numbers inside $\mathbb{R}^2$ we enhance the arithmetical playground $\mathbb{R}^2$ of (1.1) with a third operation – a special extraordinary version of an multiplication '⋆' of column/row vectors in $\mathbb{R}^2$, called *multiplication of complex numbers* via the new rule

$$\boxed{(a, b) \star (c, d) \overset{def}{=} (a \cdot c - b \cdot d, a \cdot d + b \cdot c) \qquad (1.4)}$$

If we speak of the complex numbers we think *in this section* at the 2–dimensional number plane $\mathbb{R}^2$ equipped with the three operations $(+, \cdot, \star)$ of (1.1 ff) and (1.4) and write

$$\mathbb{C} \equiv (\mathbb{R}^2, +, \cdot, \star)$$

We will motivate this strange operation ⋆ very soon.

1.2.1 ℂ as 2D algebra over the reals ℝ

For the new ℂ–typical operation ⋆ the following rules hold for arbitrary $u, v, w \in \mathbb{R}^2$:

$$\begin{array}{llcll}
(\overset{\star}{\cup}) & v \star w & \in & \mathbb{C} & \\
(\mathbf{C}^\star) & v \star w & = & w \star v & \\
(\mathbf{A}^\star) & (u \star v) \star w & = & u \star (v \star w) & \\
(\mathbf{N}^\star) & z \star e_1 & = & z & \text{with } e_1 \overset{def}{=} (1, 0) \\
(\mathbf{I}^\star) & z \star z^{-1} & = & e_1 & \text{with } z^{-1} \overset{def}{=} \left(\frac{x}{x^2+y^2}, \frac{-y}{x^2+y^2}\right) \text{ for } z \neq (0,0)
\end{array}$$

∘ The complex number $e_1 = (1, 0)$ in rule $(\mathrm{N}^\star)$ is called the *unit* in ℂ. It is ℂ's neutral element with respect to the new multiplication ⋆.
∘ The complex number z^{-1} in rule $(\mathrm{I}^\star)$ is called the *inverse of z* in ℂ.

Exercise 1.3. **a.** Verify the the above rules for ⋆ by paper'n pencil..
b. Verify the the above rules by Eigenmath.

Solution:

```
-- C as algebra
-- define new multiplication * for 2D row vectors

star(u,v)= (u[1]*v[1]-u[2]*v[2],u[1]*v[2]+u[2]*v[1])

-- rule C*              -- after '--' comes a comment
do( u=(a,b), v=(c,d))
star(u,v)
star(v,u)
star(u,v)==star(v,u)  -- operation * is commutative

-- rule N*, existence of neutral resp. *
tty = 1                 -- now line oriented output
e1  = (1,0)
z   = (x,y)
star(z,e1)
star(z,e1) == z

-- rule I*, existence of inverse resp. *
invC(z) = (z[1]/(z[1]^2+z[2]^2), -z[2]/(z[1]^2+z[2]^2))
invC((2,1))
star((2,1), invC((2,1)))
```

$$\begin{bmatrix} ac - bd \\ ad + bc \end{bmatrix}$$

$$\begin{bmatrix} ac - bd \\ ad + bc \end{bmatrix}$$

```
1
(x,y)
1
(2/5,-1/5)
(1,0)
```

We define operation '*', check its commutativity, give the neutral element $e1$, define the inverse element `invC(z)` and test it on a special case.. ▷ *Click here to run the script.*

Exercise 1.4. Check with EIGENMATH, that the following rules also hold for arbitray $r, s \in \mathbb{R}$ and $u, v, w \in \mathbb{C} = (\mathbb{R}^2, +, \cdot, \star)$

a. $(r \cdot u + s \cdot v) \star w = r \cdot (u \star w) + s \cdot (v \star w)$
b. $u \star (r \cdot v + s \cdot w) = r \cdot (u \star v) + s \cdot (u \star w)$
c. $r \cdot (u \star v) = (r \cdot u) \star v = u \star (r \cdot v)$
▷ *Click here to invoke* EIGENMATH

Remark. With both laws a.&b. of *distribution*, the operation $\star$ is compatible with the structure of the vector space $\mathbb{C}$. A vector space together with a 3rd operation $\star$, for which the above *rules of distribution* a.&b. hold, is called an $\mathbb{R}$–*algebra*. $\star$ itself is called *the multiplication of the algebra* $\mathbb{C}$. (See e.g. KOECHER & REMMERT in [4, p. 127])

Therefore the title of this section.

Exercise 1.5. Calculate with/without EIGENMATH:

a. $(1, 2) \star (3, 4)$
b. For which $w \in \mathbb{C}$ is $(1, 2) \star z = (1, 0)$?
c. $2 \cdot (3, 4) \star (-1, 1)$

Exercise 1.6. (How to motivate the construction of $\star$?)[4] The 2D vectorspace $\mathbb{R}^2$ over the scalar field $\mathbb{R}$ has the canonical basis vectors $e_1 = (1,0)$ and $e_2 = (0,1)$. We want to have the new multiplication $\star$ to work such, that
(1) e_1 should be the unit element i.e. should fulfill rule $(N^\star)$ and
(2) e_2 should be *chosen so, that its square results in the negative unit* i.e.

$$e_2^2 = (0,1)^2 \overset{!}{=} -(0,1) = -e_1$$

Therefore for arbitrary $u = (x_1, y_1), v = (x_2, y_2) \in \mathbb{C}$ we have

$$\begin{aligned}
(x_1, y_1) \star (x_2, y_2) &= (x_1 \cdot (1,0) + y_1 \cdot (0,1)) \star (x_2 \cdot (1,0) + y_2 \cdot (0,1)) \\
&\overset{(1.2)}{=} x_1 \cdot x_2 \cdot (1,0) + (x_1 \cdot y_2 + y_1 \cdot x_2) \cdot (0,1) + y_1 \cdot y_2 \cdot (0,1)^2 \\
&\overset{!}{=} x_1 \cdot x_2 \cdot (1,0) + (x_1 \cdot y_2 + y_1 \cdot x_2) \cdot (0,1) - y_1 \cdot y_2 \cdot (0,1) \\
&= (x_1 \cdot x_2 - y_1 \cdot y_2) \cdot (1,0) + (x_1 \cdot y_2 + y_1 \cdot x_2) \cdot (0,1) \\
&= (x_1 \cdot x_2 - y_1 \cdot y_2, x_1 \cdot y_2 + y_1 \cdot x_2)
\end{aligned}$$

Explain each line for yourself.

1.2.2 Introducing the imaginary unit i.

To emphasize that we calculate in $\mathbb{R}^2$ using also the new multiplication rule $\star$ one traditionally writes

$$\mathtt{i} \overset{def}{=} (0,1) = e_2$$

and name i the *imaginary unit*. In this context the unit e_1 is identified with the number 1, i.e. we have $1 \equiv (1,0) = e_1$. Therefore, per definition we have the facts:

$$i^2 = -1 \tag{1.5}$$

$$z = (x,y) = (x,0) + (0,1) \star (y,0) \overset{(1.2)}{\equiv} x + iy \in \mathbb{C} \tag{1.6}$$

○ *Beware*: with this notation $x+iy$ the use of the new multiplication $\star$ in (1.4) is shadowed behind the symbol i and our construction $(\mathbb{R}^2, +, \cdot, \star)$ is identified with build-in $\mathbb{C}$.

○ Fact (1.5) is equivalent expressed as $\boxed{i = \sqrt{-1}}$. While $\sqrt{a}$ exists in $\mathbb{R}$ only for $a \geq 0$, we have now constructed a number system in which roots of negative real numbers exists.

○ We have following important definitions for (EIGENMATH's build–in) complex numbers:

The ℂ LEXICON I	*Math*	EIGENMATH
complex number $z \in \mathbb{C}$	$z = (x,y) = x + iy$	`z = x + iy`
the *real* part of z	$Re(z) = x$	`real(z)`
the *imaginary* part of z	$Im(z) = x$	`imag(z)`
the *magnitude* (length) of z	$\lvert z\rvert \overset{def}{=} \sqrt{x^2 + y^2}$	`mag(z)`
the *conjugate* of z	$\bar{z} \overset{def}{=} x - iy$	`conj(z)`

[4] See e.g. REMMERT in [4, p. 54]

Exercise 1.7. Let $u = 1 + 2i, v = -3 - i, w = 1 + i$. Calculate with paper'n pencil
a. real and imaginary part of u
b. the magnitudes of u, v, w
c. the conjugates of all three complex numbers
d. Draw a qualtity plot with CalcPlot3D [10] of u, $|u|$, $Re(u), Im(u), \bar{u}$.
Check the plausibility of the results using the plot.
e. Check the above calculations using Eigenmath's build–in complex numbers.
▷*Click here to invoke* Eigenmath

Exercise 1.8. (*Quotient of complex numbers*)
a. Calculate $\frac{1+i}{3-4i}$.
b. Prove: Let $z_1 = x_1 + y_1 i \in \mathbb{C}$ and $z_2 = x_2 + y_2 i \in \mathbb{C}$ with $x_2^2 + y_2^2 \neq 0_{\mathbb{R}}$. Then

$$\frac{z_1}{z_2} = \frac{x_1 x_2 + y_1 y_2}{x_2^2 + y_2^2} + i \cdot \frac{y_1 x_2 - x_1 y_2}{x_2^2 + y_2^2}$$

Exercise 1.9. (*Arithmetic with complex numbers*)
Let $u = 2 - 5i, v = 4 + i \in \mathbb{C}$.
a. Calculate $u + v, u - v, u \star v, u/v$ by paper'n pencil.
b. Determine $Re(u), Im(v), \bar{u}, |u|$.
c. Check the results of a. and b. by Eigenmath.

```
# EIGENMATH solution to a):
trace=1         -- trace=1=ON shows results in black
do( u=2-5i, v=4+i)
u + v
u - v
u-v
u*v             --  *-product of complex numbers
u/v             --  quotient resp. * of complex numbers
u*1/v           --  quotient via reciprocal of v
u*v^(-1)        --  quotient via *-inverse of v
```

▷*See the solution to a. here.*

Exercise 1.10. (*Conjugate complex numbers*)
Prove with/without Eigenmath that for arbitrary $z \in \mathbb{C}$ we have
a. $Re(z) + Im(z) \in \mathbb{R}$
b. $Re(z) \cdot Im(z) \in \mathbb{R}$

1.2.3 The complex scene : the complex number $z = 4 + 3i \in \mathbb{C}$

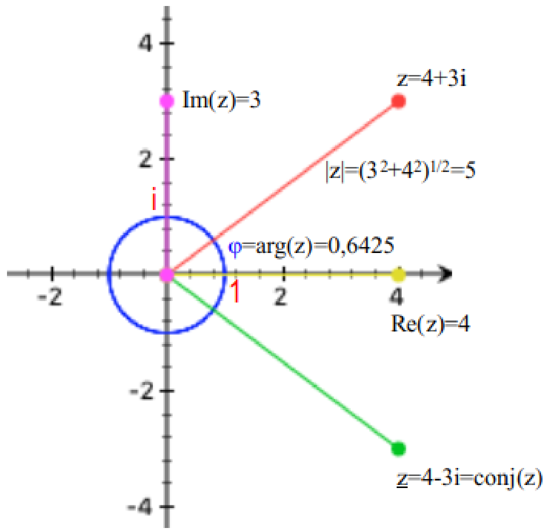


Figure 2: Red: the complex number $4+3i = (4,3) \in \mathbb{C} \equiv (\mathbb{R}^2, +, \cdot, \star)$
Green: the conjugate $\overline{4+3i} = 4 - 3i$
Yellow: the real part $Re(4+3i) = 4$
Magenta: the imaginary part $Im(4+3i) = 3$
Red: the magnitude (length) $|4+3i| = \sqrt{4^2+3^2} = 5$
Blue: the unit circle $S^1 : x^2 + y^2 = 1.$
Blue: the argument $\varphi = \measuredangle(1, z) = arctan(3/4)$='part' of S^1

Exercise 1.11.
a. Check the results in Fig.2 by a paper'n pencil calculation.
b. Check the results in Fig.2 by EIGENMATH. *Solution*:

```
# EIGENMATH
trace=1             -- trace=1=ON shows results in blue
z = 4+3i
conj(4+3i)          -- conjugate of z
real(4+3i)          -- real part of z
imag(4+3i)          -- imaginary part of z
mag(4+3i)           -- magnitude (= Euclidean length) of z
phi = arg(4+3i)     -- argument (angle) of z
phi
arg(4.+3i)          -- 4.=4.0 gives back decimal approximation
                    -- phi:(2*pi)=alpha:360  -> alpha=phi*180/pi
                    -- the argument (angle) measured in degrees:
alpha=float(phi*180/pi)
alpha               --  = 36.9 deg (phi measured in radians)
```

▷ *Click here to invoke the script.*

1.2.4 The complex exponential function

Definition. We define for arbitrary $z \in \mathbb{C}$

$$e^z \equiv exp(z) \stackrel{def}{=} \sum_{k=0}^{\infty} \frac{z^k}{k!} = 1 + z + \frac{z^2}{2!} + ...$$

- We get a function $exp\colon \mathbb{C} \to \mathbb{C}$, because the series is absolute convergent on $\mathbb{C}$.
- Analog we define $cos, sin, ...$ via convergent series, see ▷ Calculus.

Exercise 1.12. (exp, cos, sin as complex functions)
Let $z = 1 + i$. Calculate ...
a. ... the partial sum $1 + \frac{z^1}{1!} + \frac{z^2}{2!} + \frac{z^3}{3!} \sim exp(1 + i)$. Calculate $exp(1 + i)$ by EIGENMATH. How many summands must the partial sum have, such that her value coincide with the first 3 decimals of $exp(1 + i)$?

▷ *Click here to invoke* EIGENMATH *for your own calculation.*
▷ *Click here to look at my solution.*

b. ... the partial sum $\sum_{k=0}^{5} (-1)^k \frac{z^{2k+1}}{(2k+1)!}$. Compare with $sin(1 + i)$ using EIGENMATH.
c. ... the partial sum $\sum_{k=0}^{5} (-1)^k \frac{z^{2k}}{(2k)!}$. Compare with $cos(1 + i)$ using EIGENMATH.

We remind without proof to

Theorem I. *(The* EULER *formula)*
For arbitrary $z \in \mathbb{C}$*:*

$$e^{iz} = cos(z) + i\ sin(z) \tag{1.7}$$

Theorem II. *(The* DE MOIVRE *formula)*
For arbitrary $n \in \mathbb{N}$, $\phi \in \mathbb{R}$ *we have for* $z = exp(i\phi) \in \mathbb{C}$

$$z^n = exp(i\,n\phi) \tag{1.8}$$

$$(cos(\phi) + i\ sin(\phi))^n = cos(n\phi) + i\ sin(n\phi) \quad \in \mathbb{C} \tag{1.9}$$

Exercise 1.13. Calculate
a. $exp(i \star (1 + i)) \equiv e^{i\star(1+i)} =?$
b. $exp(2i)^3$
c. $(1 + 2i)^3$

1.2.5 The polar form of a complex number and their polar coordinates

Theorem III. (The polar form of a nonzero complex number)

Every $z = x + iy \in \mathbb{C} - \{0\}$ *can* uniquely *be written in the so–called* polar form

$$z \quad = \quad r \cdot (cos(\varphi) + i\ sin(\varphi)) = r \cdot e^{i\varphi} \tag{1.10}$$

for $0 \leq \varphi \leq 2\pi$, *where* $\varphi \stackrel{def}{=} arg(z) = tan^{-1}(\frac{y}{x})$ *and* $r \stackrel{def}{=} |z| = \sqrt{x^2 + y^2}$.

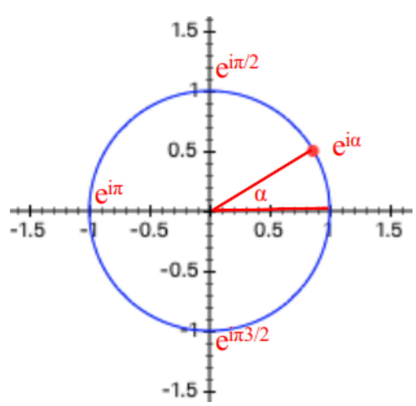
- The real numbers $(r, \varphi) \in \mathbb{R}^2$ are called the *polar coordinates* of $z \in \mathbb{C}$.
- The number $\varphi \in [0, 2\pi[$ is called the *argument* or *amplitude* of z.
- The real number r is the distance of z to the origin $O = (0, 0)$ and φ is the angle between the positive x–axis and the direction arrow to z, see Fig.2.
- Often we use the abbreviation $cis(\varphi) \stackrel{def}{=} (cos\varphi + i \cdot sin\varphi)$. We then have $z = cis(\varphi)$.
- We remind at

The ℂ LEXICON II:	*Math*	EIGENMATH
complex number z in *polar* form with ..	$z = r \cdot e^{i\varphi}$	`polar(z)`
... $r = \lvert z \rvert$ and	$= r \cdot (cos\varphi + i \cdot sin\varphi)$	
... *argument* $\varphi = arg(z) \in [0, 2\pi[$	$\varphi = \measuredangle(e_1, z) = \tan^{-1}(\frac{y}{x})$	`phi = arg(z)`
complex number z in *rectangular* form	$z = x + i \cdot y$	`rect(z)`
the *complex root* of z	$cis(\varphi) = (cos\varphi + i \cdot sin\varphi)$	`cis(phi)=..`
the *complex power* of z	$Im(z) = x$	`real(z)`
the ν^{th} *complex unit root* of $z^n = 1$	$\zeta_\nu^n = e^{\frac{2\pi i}{n}\nu}, \nu = 0, 1, ...n - 1$	`exp(2 pi i nu/n)`

Summary: we have therefore three different shapes of a complex number

rectangular	*trigonometric*	exponentially
.. alias Cartesian		.. alias polar form
$z = x + iy =$	$r \cdot (\cos\varphi + i \cdot \sin\varphi) =$	$mag(z) \cdot \exp(i \cdot arg(z)) = r \cdot e^{i \cdot \varphi}$

- We visualize some polar factors $exp(i \cdot \varphi) = e^{i \cdot \varphi}$ as points at the unit circle S^1:



Exercise 1.14. (Polar vs rectangular form of a complex number)
a. What is the argument and the magnitude of $z = 3 + 2i$?
b. Give z in polar form $e^{\cdots}$. Check the equivalence of both representations.
c. Recover the rectangular form of z in a. back from its polar form in b..
d. Check your results by EIGENMATH.
▷ *Click here to look at the solution.*

Exercise 1.15. (Visualization of the complex multiplication $\star$)
By means of the polar form of a complex number one can visualize the effect of the complex multiplication.

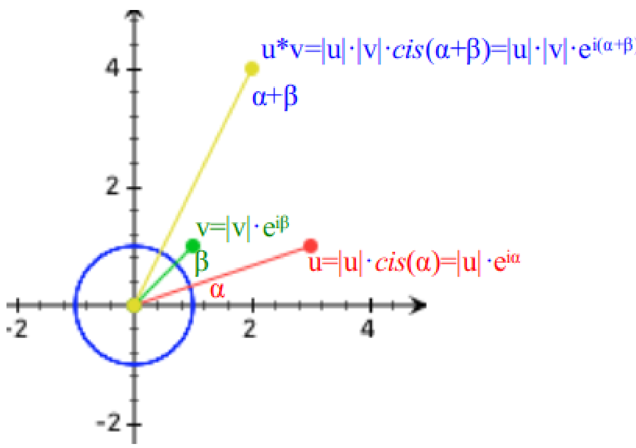


Figure 3: Red: the 1st factor u with his argument $\alpha = \measuredangle(e_1, u)$
Green: the 2nd factor v with his argument $\beta = \measuredangle(e_1, v)$
Blue: the product $u \star v$ has argument (angle) $\measuredangle(e_1, \alpha + \beta)$
The arguments (angles) are best seen as arc pieces on the unit circle $S^1 : x^2 + y^2 = 1$ starting at $e_1 = (1, 0)$.

a. In Fig.3 we have $u = 3 + i$ and $v = 1 + i$. What are the coordinates of the yellow point?
b. Transform the arguments α, β and $\alpha + \beta$ in degrees. Compare.
◦ Slogan: *you get the product of two complex numbers by multiplying their magnitudes and adding their arguments (angles)*.

Exercise 1.16. (Programming a `polar1` function for EIGENMATH)
EIGENMATH has two functions for handling polar ("$e^{\cdots}$") and rectangular ("$a + bi$") forms of complex numbers:
◦ `polar(z)` awaits as input $z = a + bi$ in rectangular form and returns its *polar* form.
◦ `rect(z)` awaits as input $z = e^{\cdots}$ in (exp=)polar form and returns its *rect*angular form.
Sometimes one has length r and angle $\varphi = arg(z)$ as inputs and needs the polar term. Therefore:

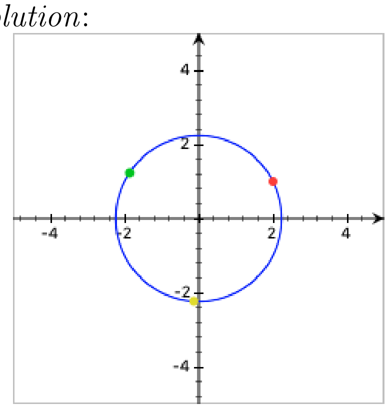
a. Write a user defined function `polar1`, which awaits (r, φ) as input and returns the polar expression ... $\cdot\, exp(..)$ as output.
b. What is `polar1(sqrt(13), arctan(2/3))` in rectangular form? In decimals?
b. Using `polar1`, what result do you respect for the expressions

```
polar1(r,p)*polar1(s,q)
1/polar1(r,p)
polar1(r,p)^3
```

Check your guess by EIGENMATH. ▷ *Click here to look at the solution.*

Exercise 1.17. (Polar form of a complex product or quotient)
Let $z, u, v \in \mathbb{C}$.
a. Determine the polar form of the complex numbers in rectangular form $1, i, -1, -i$.
b. Determine the rectangular form of $u = exp(1/3i\pi)$ and $v = \sqrt{2} \cdot exp(\frac{1}{4}i\pi)$.
c. $polar(\bar{z}) = ?$
d. $polar(z^{-1}) = ?$
e. Verify: $polar(u \star v) = |u| \cdot |v| \cdot e^{(\varphi+\psi)\cdot i} = |u| \cdot |v| \cdot cis(\varphi + \psi)$
i.e. again: *you get the product of two complex numbers by multiplying their magnitudes and adding their arguments* (angles).
f. $polar(\frac{u}{v}) = ?$
▷ *Click here to see the solution.*

Exercise 1.18. (The 3rd roots of a complex number)
We seek the complex solutions of the equation $z^3 = 2 + 11i$. We know by the so-called *Fundamental Theorem of Algebra*, that this equation must have exactly 3 solutions in $\mathbb{C}$.
a. Verify by paper'n pencil that $w1 = 2 + i$ is a solution of $z^3 = 2 + 11i$.
b. Use EIGENMATH to verify that $w2 = (2+i)\star(-\frac{1}{2} + \frac{\sqrt{3}}{2} \cdot i)$ and $w3 = (2+i)\star(-\frac{1}{2} - \frac{\sqrt{3}}{2} \cdot i)$ are solutions, too.
c. Plot the three solutions. *Solution*:



▷ *Click here to run the solution.*

Exercise 1.19. (Roots of complex numbers as edges of a regular polygon)
Let $a = r \cdot exp(i\psi) \in \mathbb{C}$.
a. Verify, that

$$root_k^n(a) \equiv \sqrt[n]{a}|_k \stackrel{def}{=} r^{1/n} \cdot \exp(i\frac{\psi + 2k\pi}{n})$$

for $k = 0, ..., n-1$ is the k^{th} of the n complex roots of a, i.e. $(\sqrt[n]{a}|_k)^n = a$.
○ The n^{th} root of a is therefore a whole *set* of complex numbers:

$$\sqrt[n]{a}| = \{\sqrt[n]{a}|_k \in \mathbb{C} \mid k \in \{0, .., n-1\}\}$$

b. Determine $\sqrt[4]{1}|$, $\sqrt[4]{i}|$.
c. Determine $\sqrt[4]{2}|$ and visualize this root set.
d. Verify by an quality plot that the root (set) $\sqrt[n]{a}|$ of a complex number a is the edge set of a regular n-gon e.g. $\sqrt[3]{a}| = \triangle$ or $\sqrt[4]{a}| = \square$ or ...

1.2.6 Inner and outer products of complex numbers

Definition. Let $u = (u_1, u_2) = u_1 + i \cdot u_2$ and $v = (v_1, v_2) = v_1 + i \cdot v_2$ be in $\mathbb{R}^2$.
○ The *scalar* alias *inner product* of u and v in the real vector space $\mathbb{C} = \mathbb{R}^2$ is defined as

$$u \bullet v \stackrel{def}{=} u_1 \cdot v_1 + u_2 \cdot v_2$$

○ The *outer* alias *wedge product* of u and v is defined as

$$u \wedge v \stackrel{def}{=} Im(\bar{u} \star v)$$

Exercise 1.20. Given $w = 2 + 3i, z = 3 - 5i$ and $u, v \in \mathbb{C}$.
a. Determine $w \bullet z, z \bullet z$ and $w \wedge z, w \wedge w$.
b. Verify: $u \bullet v = Re(u * \bar{v})$
c. Proof: $u \perp c \cdot u \Leftrightarrow c \in i \cdot \mathbb{R}$, i.e. if c is pure imaginary.
○ Check your results by EIGENMATH.
$\triangleright$ *Click here to look at the solution.*

1.2.7 Problems

P1. Complex square roots and the normed quadratic equation.
In $\mathbb{C}$ there exists always complex square roots. We have the fact [5]

Theorem IV. *(The complex square root formula)*
For arbitrary $c = a + bi \in \mathbb{C}, a, b \in \mathbb{R}$ *define*

$$\zeta := \sqrt{\frac{1}{2}(a+|c|)} + \frac{b}{|b|} \cdot \sqrt{\frac{1}{2}(-a+|c|)} \cdot i \qquad \text{if } b \neq 0. \tag{1.11}$$

$$\zeta := \sqrt{|c|)} \qquad \text{if } b = 0, a \geq 0. \tag{1.12}$$

$$\zeta := \sqrt{|c|)} \cdot i \quad \text{if } b = 0, a < 0. \tag{1.13}$$

Then $\zeta^2 = c$. *– We write:* $\zeta := \sqrt{c}$.
a. Program theorem IV in EIGENMATH.
b. Determine the complex square roots of -2, $1+i$, $exp(\pi i)$ by paper'n pencil and with EIGENMATH.
c. Using the ancient Babylonian trick of completing the square we get:
The standard quadratic equation $z^2 + az + b = 0$ has the two solutions z_1 and z_2:

$$z_{1|2} := -\frac{1}{2} \cdot a \pm \frac{1}{2}\sqrt{a^2 - 4 \cdot b} \tag{1.14}$$

Program the solution formula (1.11) in EIGENMATH.
d. Solve $x^2 - 10x + 34 = 0$.
e. Solve $z^2 + iz + 2 - 4i = 0$.
f. Solve $5z^2 + 2z + 10 = 0$.

P2. Solution of general quadratic equations.
Read more about the quadratic equation. E.g.

> ”*Solution for complex roots in polar coordinates*: If the quadratic equation $ax^2 + bx + c = 0$ with real coefficients has two complex roots – the case where $b^2 - 4ac < 0$, requiring a and c to have the same sign as each other – then the solutions for the roots can be expressed in polar form as x_1, $x_2 = r(\cos\theta \pm i \sin\theta)$, where $r = \sqrt{\frac{c}{a}}$ and $\theta = \cos^{-1}\left(\frac{-b}{2\sqrt{ac}}\right)$. [See url: `https://en.wikipedia.org/wiki/Quadratic_equation`]

a. Solve $x^2 - 10x + 34 = 0$ using the polar form.
b. Solve equation a. using the standard solution formula (1.11).
c. Solve $x^2 - 10x + 40 = 0$.[6]
d. Solve $5z^2 + 2z + 10 = 0$ using the polar form.

---

[5] See e.g. REMMERT in [4, p. 62]
[6] see HOFFMANN at `http://www.math.uni-konstanz.de/~hoffmann/Funktionentheorie/kap1.pdf`

P3. Solution of cubic equations – CARDANO's formula.
Read about the cubic equation.
E.g. [°] `https://mathshistory.st-andrews.ac.uk/HistTopics/Quadratic_etc_equations/`.

a. Program the algorithm for CARDANO's solution of the special cubic equation $x^3+mx=n$ to be found in [°] "in modern notation". Then:
Solve $x^3 = 15x + 4$.

b. Look at `https://www.mathematik.ch/anwendungenmath/Cardano/FormelCardano.html`.
Then solve $x^3 + 3x^2 + 9x + 9 = 0$ using EIGENMATH.

P4. Construction of the complex numbers via 2×2 matrices.
A well-known construction[7] represents $\mathbb{C}$ as a special set of matrices using the matrix multiplication $\star$ as complex multiplication.
Let $\hat{\mathbb{C}} \equiv (\{\begin{bmatrix} a & b \\ -b & a \end{bmatrix} \in \mathbb{R}^{2\times 2} \mid a, b \in \mathbb{R}\}, +, \star)$ be the set of skew-symmetric 2×2 matrices with equal diagonal elements. We abbreviate $C(a, b) := \begin{bmatrix} a & b \\ -b & a \end{bmatrix}$, which corresponds to the usual complex number $a + bi$.
a. Formulate this construction in EIGENMATH.
b. Calculate the sum $C(1, 2) + C(3, 4)$ and the product $C(1, 2) * C(3, 4)$.
Check the result using 'normal' complex numbers $\mathbb{C}$.
c. Verify: $C(a, 0) + C(b, 0) = C(a + b, 0)$ and $C(a, 0) \star C(b, 0) = C(a \cdot b, 0)$.
○ Therefore $\hat{\mathbb{C}}$ contains the real numbers $\mathbb{R}$ identified as the diagonal matrices. The matrix $C(a, 0)$ "is" the real number a.
d. Show: $C(0, 1)^2 \equiv -1$.
Therefore $C(0, 1)$ corresponds to $i \in \mathbb{C}$ and we have a isomorphism between $\hat{\mathbb{C}}$ and $\mathbb{C}$.

[7] See e.g. REMMERT in [4, p. 56]

1.3 ℂ as algebraic structure

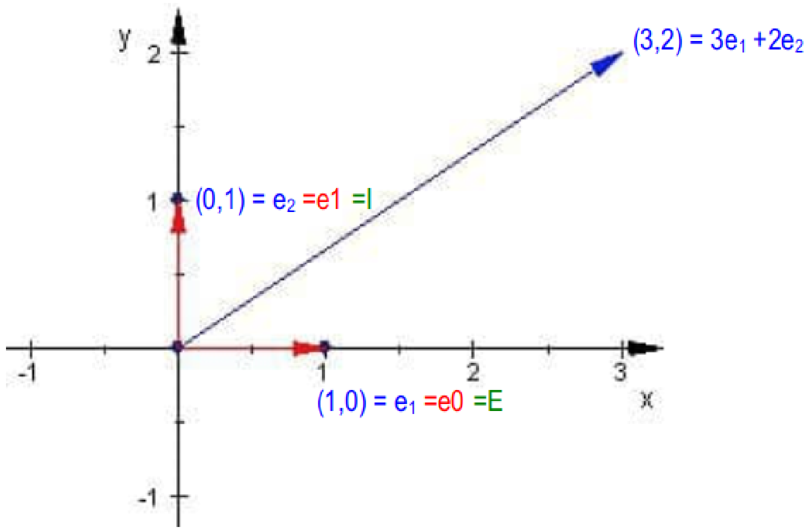


Figure 4: Blue: the point/vector $(3,2)$. The basis $\{e_1, e_2\}$ of $\mathbb{R}^2$
Red: the same basis noted $\{e_0, e_1\}$ and $(3,2) = 3 \cdot e_0 + 2 \cdot e_1$
Green: $(3,2) = 3 \cdot E + 2 \cdot I$ in basis noted $\{E, I\}$

To construct the complex numbers ℂ in an alternative way, we enhance the arithmetical playground $\mathbb{R}^2$ again with a third operation – but this time by means of a 'multiplication table' for the operation (the 'algebra multiplication'), noted '$\circledast$'. This makes $\mathbb{R}^2$ into the structure $(\mathbb{R}^2, \circledast)$ of an 'algebra'[8]. One can master the algebra multiplication of the new $\mathbb{R}$–algebra by means of its property of *bilinearity*, if one knows its effect on all possible $2^2 = 4$ [9] pairs of the elements of a basis of the underlying vector space $\mathbb{R}^2$, see e.g.[3, pp. 192–193] Here we use the fact that the algebra unit $1_{\mathbb{C}} = (1,0) = 1_{\mathbb{R}^2}$ also occurs naturally in this basis.

Therefore, to construct the new multiplication rules we describe for the 2 basis vectors $e_0 = (1,0)$ and $e_1 = (0,1)$[10] of the vector space $\mathbb{R}^2$ the following results for the operation $\circledast$:

$$e_0 \circledast e_0 = e_0 \tag{1.15}$$
$$e_0 \circledast e_1 = e_1 \tag{1.16}$$
$$e_1 \circledast e_0 = e_1 \tag{1.17}$$
$$e_1 \circledast e_1 = -e_0 \tag{1.18}$$

[8] A $\mathbb{R}$-*algebra* is a pair $(A, \circledast)$, consisting of an $\mathbb{R}$-vector space and an $\mathbb{R}$-binlinear mapping $\circledast\colon A \times A \to A$ defined through $(a,b) \mapsto a \circledast b$

[9] in general $n^2, n = dim_{\mathbb{R}} A$, A being the algebra.

[10] Why we adopt the notation e_0, e_1 alias E, I instead of the usual notation e_1, e_2 for the two basis vectors of $\mathbb{R}^2$ will become clear later on.

Remark. If we translate the 4^{th} rule $e_1 \circledast e_1 = -e_0$ using the lexicon $\stackrel{e_0,\, e_1}{\mathbf{1,\ i}}$ in die language of $\mathbb{C}$, we get the desired relation $\stackrel{e_1 \circledast e_1 = -e_0}{i \star i = i^2 = -1}$.

Noted as a *compact multiplication table* for the algebra multiplication $\circledast$, we have:

$$\begin{array}{c c} \circledast & \begin{array}{cc} e_0 & e_1 \end{array} \\ \begin{array}{c} e_0 \\ e_1 \end{array} & \begin{pmatrix} e_0 & e_1 \\ e_1 & -e_0 \end{pmatrix} \end{array}$$

If we now speak of the complex numbers we think at the 2–dimensional number plane $\mathbb{R}^2$ equipped with the three operations $(+, \cdot, \circledast)$ and write $\widetilde{\mathbb{C}} \equiv (\mathbb{R}^2, +, \cdot, \circledast)$.

Exercise 1.21. Calculate $(1+i) \star (-2+2i)$ using the multiplication table. *Solution*:

$$\begin{aligned} (1+i) \star (-2+2i) &\equiv (e_0 + e_1) \circledast (-2e_0 + 2e_1) \\ &= -2e_0 \circledast e_0 + 2e_0 \circledast e_1 - 2e_1 \circledast e_0 + 2e_1 \circledast e_1 \\ &= -2e_0 + 2e_1 - 2e_1 + 2(-e_0) \\ &= -4e_0 = (-4, 0) \equiv -4 \end{aligned}$$

1.3.1 Implementing the algebra structure $\widetilde{\mathbb{C}} \equiv (\mathbb{R}^2, +, \cdot, \circledast)$ in EIGENMATH

In order to effectively calculate in the new algebraic playground $(\mathbb{R}^2, +, \cdot, \circledast)$ we have to translate the construction above into EIGENMATH command language.

```
##### C alias R[i]
tty = 1

e0 = (1,0)              -- basis vectors
e1 = (0,1)

T = ((e0, e1),          -- (1) multiplication table
     (e1,-e0))

B = transpose(T,2,3)    -- (2) bilinear operation

mu(x,y) = dot(x,B,y)    -- (3) x^t*B*x

mu(e0,e0)               -- should be  e0=(1,0)
mu(e0,e1)               -- should be  e1=(0,1)
mu(e1,e0)               -- should be  e1=(0,1)
mu(e1,e1)               -- (4) should be -e0 = (-1,0)

mu( 2e0+3e1, 1e0-2e1 )   -- (5) mu_ltiplication

mu(a*e0+b*e1, c*e0+d*e1) -- (6)
```

```
T = (((1,0),(0,1)),((0,1),(-1,0)))
B = (((1,0),(0,1)),((0,-1),(1,0)))
(1,0)
(0,1)
(0,1)
(-1,0)
(8,-1)
(a c - b d,a d + b c)
```

Comment. In (1) we implement the multiplication table as a tensor, i.e. as a matrix consisting of two 2×2 matrices. Function $mu \stackrel{def}{=} \circledast$ defines in (2) the bilinear operation, which uses the cool possibility of EIGENMATH's `dot(.)` to allow multiple inputs. (4) verifies that the construction fulfills the desired relation $e_1^2 = -(1,0) \equiv -1$. For an arbitrary input mu returns in (6) the well known formula for the complex multiplication.

▷ *Click here to run the script.*

Exercise 1.22. Verify the calculation in (5) by only using the multiplication table. □
To convince the reader that we calculate indeed in ℂ, we spend a bit syntactic sugar and set $E \stackrel{def}{=} e_0$, $I \stackrel{def}{=} e_1$ to get the usual appearance:

```
E = (1,0)            # _E_mbedding of R in C
I = (0,1)            # _I_maginary unit
                     -- C = R[i] multiplication table:
                     -- E*E = E, E*I = I, I*E = I, I*I = -E=-1

T = ((E, I), (I,-E))
B = transpose(T,2,3)

mu(x,y) = dot(x,B,y)

mu(I,I)              -- = -E = -1 i.e.  i^2=-1

mu( 2E+3I, 1E-2I)    -- complex algebra via multiplication table T
(2*1+3i)*(1*1-2i)    -- build-in complex algebra, returning  8-i = (8,-1)
```

▷ *Click here to run the script.*

1.3.2 Reengineering of some complex functions

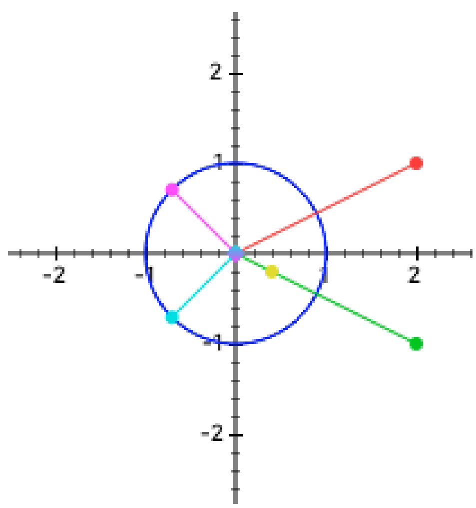


Figure 5: Blue: The unit circle $S^1 \subset \mathbb{R}^2$ with equation $x^2 + y^2 = 1$.
Red: The complex number/vector $z = 2 + 1 \cdot i = (2, 1)$
Green: ... its conjugate $\bar{z} = (2, -1) = 2 - i$
Green: ... and its inverse $1/z$.
Magenta: The complex number $w = 1_{135^o} \approx (0.707, 0.707)$
Cyan: .. and its inverse $1/w$

Exercise 1.23. Regarding Fig. 5, calculate using build–in complex functions of EIGENMATH for the complex numbers $\mathbb{C} \equiv (\mathbb{R}^2, +, \cdot, \star)$
a. the $\star$–inverse $1/z$ of $z = 2 + i$.
b. the rectangular form of $w = 1_{135^o} = r_\varphi \in S^1$.
c. the inverse of w.

> We now want to reconstruct the results of *Ex.1.23* using the new $\mathbb{R}$–algebra $\widetilde{\mathbb{C}} \stackrel{def}{=} (\mathbb{R}^2, +, \cdot, \circledast)$. Therefore we have to write functions e.g. to compute the *conjugate*, the real and *imaginary part*, the *length* (alias norm), the *reciprocal* (inverse) and the *quotient* resp. *the table based multiplication operation* $\circledast$.

Exercise 1.24. (Functions for the algebra $\widetilde{\mathbb{C}} \equiv (\mathbb{R}^2, +, \cdot, \circledast)$)
Let $w = (w[1], w[2])$ be a arbitrary 'new complex' number, i.e. $w \in \mathbb{R}^2 \equiv \widetilde{\mathbb{C}}$.
Use the EIGENMATH playground to solve the following tasks.
▷ *Click here to open the playground.*
a. Copy/Write the following functions on the playground:

```
im(w) = w[2]
re(w) = w[1]
cj(w) = (w[1],-w[2])             -- conjugate of w
iv(w) = 1/(w[1]^2+w[2]^2)*cj(w) -- inverse of w

A = 2E+3I
B = 1E-2I
```

and test these functions on the $\widetilde{\mathbb{C}}$–numbers A and B.
b. Redo Ex.1.23 using the functions and notations from a..
c. Here is a definition to compute the *quotient* u/v of two new–complex numbers:

```
qu(u,v)= 1/(v[1]^2+v[2]^2) *
            (u[1]*v[1] + u[2]*v[2],
             v[1]*u[2] - u[1]*v[2])
```

- Calculate $qu(A, B)$. Check the result using build–in functions of EIGENMATH.
- Give an alternative definition of qu using the inverse function iv.
- Give an alternative definition of iv using the quotient function qu.

d. What does `mu(A,iv(A))` test? Write this expression in math language.
e. Collect the functions of this exercise in a toolbox named `cBox.txt` for later use.
▷ *Click here to see the solution.*

1.3.3 Problems.

P5. The norm of a number $w \in \widetilde{\mathbb{C}} \equiv (\mathbb{R}^2, +, \cdot, \circledast)$.
Let w be a arbitrary 'new complex' numbers of $\widetilde{\mathbb{C}}$.
a. Define a function `no(w)` to calculate the *norm* alias the length of w.
b. Determine the norm of all 5 points in Fig. 5.
c. What is the length of $A = 2E + 3I$ and $B = 1E - 2I$?
d. Check the results by interpreting and writing A and B as 'usual' complex numbers. Use paper'n pencil and EIGENMATH.
▷ *Click here to see the solution.*

P6. The inner and outer products in $\widetilde{\mathbb{C}}$.
Let $U, V \in \widetilde{\mathbb{C}}$ be two arbitrary 'new complex' number.
a. Define the two functions `ip(U,V)` and `op(U,V)` to compute the *inner* resp. *outer product* of complex numbers through

```
ip(U,V) = inner(U,V)              -- inner product alias scalar product
op(U,V) = U[1]*V[2] - V[1]*U[2]   -- outer product
```

- Calculate the inner and outer product of $U = 3E - 4I$ and $W = -4E + 3I$.
- Calculate the inner and outer product of $A = 2E + 3I$ and $B = 1E - 2I$.
- Calculate the inner and outer product of z and $\bar{z}$ of Fig. 5.

b. Prove: `ip(U,W) = re(mu(cj(U),W))`.
Formulate this formula in mathematical language. **c.** Formulate and prove a similar formula for the outer product.
d. Verify the results of Ex.1.20 by arithmetic in $\widetilde{\mathbb{C}}$.
▷ *Click here to see the solution.*

Remark. The complete set of EIGENMATH functions for this section are bundled in the toolbox `cBox.txt` for convenience.

$\circledast$

Summary: We have constructed a new algebra $\widetilde{\mathbb{C}}$ in the Euclidean plane $\mathbb{R}^2$ by means of a multiplication table for the basis vectors $span_{\mathbb{R}}\{e_1, e_2\}$. This construction was totally independent of the 'old' complex numbers build–in in EIGENMATH. Nevertheless we get also the desired relation $I^2 = -1$ to have a root of $\sqrt{-1}$. We were able to define the crucial $\mathbb{C}$–typical functions like conjugate, imaginary part, reciprocal, norm etc. in this setting, too.

1.4 ℂ as Clifford algebra $c\ell(2,0)$

In this section we reconstruct the complex numbers ℂ again, this time using a universal recipe, which we will use later to implement the *hyperbolic numbers*, the *quaternions* and the *2D/3D geometry* with enhanced insights: the Geometric algebra 'GA'.

This time we will use the Eigenmath package `EVA2.txt`[11] for the first time. We will use it without to say e.g. what a 'graded algebra' is. Later in Chapter 4 we have to say more about this, telling the motivation behind the construction. But first we should make some easy experiences in the mere using of `EVA2` as another possibility to calculate with complex numbers ...

1.4.1 A first look at the 4D-Clifford algebra $c\ell(2,0)$

Here is our new algebraic playground:

```
run("downloads/EVA2.txt") # load package EVA
cl(2)                     # (1) specify the Clifford Algebra
tty=1                     # compact output setting

e0                     -- (2) the 4 basis vectors e0,e1,e2,e12
e1
e2
e12

U = 1e0+2e1+3e2+4e12  -- (3) a 4D vector as lin.combi.
U
V = -3e1+4e2
V
U+V                    -- (4) usual 4D addition
U-V                    -- (5) .. and subtraction
2U+3V                  -- (5) a scalar multiple
magnitude(V)           -- (6) the length of V
Vn=normalize(V)        -- (7) unit vector in direction V
Vn
inp(Vn,Vn)             -- (8) the inner/scalar product
inp(U,V)               -- feel at home

gp(e0,e0)              -- (9) the gp = GeometricProduct
gp(e1,e0)              --     as new algebra multiplication
gp(e12,e12)            -- (10) a kind of imaginary unit
```

$$\begin{bmatrix} + \\ + \end{bmatrix}$$

```
e0 = (1,0,0,0)
e1 = (0,1,0,0)
e2 = (0,0,1,0)
e12 = (0,0,0,1)
U = (1,2,3,4)
V = (0,-3,4,0)
(1,-1,7,4)
(1,5,-1,4)
(2,-5,18,8)
5.0
Vn = (0.0,-0.6,0.8,0.0)
(1.0,0,0,0)
(6.0,13.0,16.0,0)
(1.0,0.0,0.0,0.0)
(0.0,1.0,0.0,0.0)
(-1.0,0.0,0.0,0.0)
```

▷ *Click here to invoke this script* and to experiment a bit.

Comment. The call $cl(2)$ alias $cl(2,0)$ of the function $cl(..)$ of the EVA package give the output $\left[\begin{smallmatrix} + \\ + \end{smallmatrix}\right]$. This means that the norm has the *signature* $(+,+)$, i.e. $\sqrt{+x^2+y^2}$. In line (2) we list the basis vectors $span_{\mathbb{R}}\{e0, e1, e2, e12\}$, which here have other names than the usual e_1, e_2, e_3, e_4. Why? Wait.
But nevertheless we feel immediately at home in this 4D vector space $c\ell(2)$ when studying and looking at lines (3) until (8). Here *magnitude*, *normalize*, *inp*, and *gp* are functions of the package `EVA`, which are not available outside of this package.

[11] `EVA2` is an abbreviation for 'Euclidian Vector Algebra' version 2. We have to thank Bernard Eyheramendy [5] for this package. It is currently the biggest collection of user defined functions in Eigenmath.

Line (10) is crucial: it remembers at the characteristic feature $i^2 = -1$ of the imaginary unit $i \in \mathbb{C}$ of the complex numbers, i.e.

$$gp(e12, e12) = \text{"}\ (e12)^2\ \text{"} = (-1, 0, 0, 0) \equiv -1$$

∘ This observation will lead to an realization of ℂ inside the CLIFFORD algebra $c\ell(2,0)$.
∘ For the moment we may think of the geometric product gp as given through a $4{\times}4 = 16$ entry multiplication table a la ⊛ for the algebra $\widetilde{\mathbb{C}}$ in the last section.

Exercise 1.25.
a. Find two vectors $u, v \in c\ell(2)$ which are orthogonal resp. the scalar product inp.
b. Do some more free experiments in the 4D algebra $c\ell(2)$.

1.4.2 ℂ as part of the CLIFFORD algebra $c\ell(2,0)$

Here is our realization of the complex numbers ℂ as a 2D sub-algebra $\widehat{\mathbb{C}} \overset{def}{=} (\mathbb{R}^4, +, \cdot, gp)$. By sub-algebra we mean that we will only use linear combinations of the *two* basis vectors $e0, e12$, i.e. with the alias $E \overset{def}{=} e0$, $J \overset{def}{=} e12$ we define

$$\widehat{\mathbb{C}} = (span_{\mathbb{R}}\{E, J\}, +, \cdot, gp)$$

Remark. The CLIFFORD algebra multiplication, noted `gp(a,b)` in EIGENMATH package `EVA`, is often noted in mathematical texts as `ab` – *without any separating multiplication sign between the factors.* We do not recommend that use for the beginner. Instead we use a notation like $a \circledcirc b$ or $a \boxdot b$ or $a \circ b$ for `ab=gp(a,b)`.

geometric product:	*Math*	EIGENMATH
	$A \circ B$	`gp(A,B)`

```
run("EVA2.txt")
cl(2)                 -- invoke Clifford Algebra (2,0)
                      -- We give some syntactic sugar ..
E = e0                -- to Embed the real numbers R^1
J = e12               -- to have usual name for Jmaginary unit
gp(J,J)               -- output: (-1,0,0,0) i.e. J^2 = -1

a =  1e0 + 2e12
b = -2e0 + 3e12
b                     -- output: b=(-2,0,0,3) == -2+3i
-- is now noted as

a =  1*E + 2*J
b = -2*E + 3*J
b                     -- output: b=(-2,0,0,3) == -2+3i
```

▷ *Click here to invoke this script.*

1.4.3 Clifford algebra cheatsheet for EVA2

Here is a cheatsheet of the main functions of the package EVA2 for future use:

	Math	Eigenmath EVA2
geometric product	A B	`gp(A,B)`
inner/scalar product	$A \bullet B$	`inp(A,B)`
outer product	$A \wedge B$	`outp(A,B)`
Clifford conjugation	$\bar{B}$	`cj(B)`
inverse	$1/B = B^{-1}$	`inverse(B)`
magnitude	$\|B\|$	`magnitude(B)`
normalize	$\frac{B}{\|B\|}$	`normalize(B)`

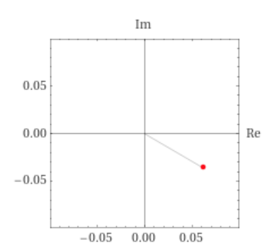
• There are also the Clifford algebra versions[12] for the regular build-in functions of the complex domain, always noted with an ending 1 to distinct it from the ℂ–functions:

`imag1, real1, polar1, rect1, exp1, log1, sqrt1, power1, sin1, cos1, tan1, sinh1, cosh1, tanh1, asin1, acos1, atan1, asinh1, acosh1, atanh1, ..`

Exercise 1.26. (Wolfram|*alpha* for complex numbers)
Wolfram|*alpha* works with complex numbers: ▷*Click here to invoke* Wolfram*'s page.*
Check their examples and results using the EVA package. E.g.
a. Calculate $1/(12+7i) \in \mathbb{C}$ inside $c\ell(2)$ using EVA. *Example* solution:

```
run("EVA2.txt")
cl(2)
inverse(12E+7J)        -- complex arithmetic in cl(2) with EVA
1/(12+7i)              -- complex arithmetic in EIGENMATH
```

Visualize the result,
loc. cit. Wolfram|*alpha*:



b. Do the other calculations from that page.
▷*Click here to invoke the script.*

c. Redo some of the exercises in section 1.5, 1.7–1.14, 1.17 and 1.20 of this booklet using Eigenmath's EVA.

[12]Most of these functions are implemented using partial Taylor sums, therefore giving 'only' approximate decimal values.

2 IH – the hyperbolic numbers

> ”The hyperbolic numbers are blood relatives of the popular complex mumbers and deserve to be taught alongside the latter. They serve not only to put Lorentzian geometry on an equal mathematical footing with Euclidean geometry, but also help students develop algebraic skills and concepts necessary in higher mathematics. Whereas the complex numbers extend the read numbers to include a new number $i = \sqrt{-1}$, the hyperbolic numbers extend the real numbers to include a *new* square root $u = \sqrt{+1}$, where $u \neq \pm 1$. [13, p. 2]

This new number u is named the *unipotent*. This u solves the equation $x^2 = 1$, but has the properties $u \neq +1$ and $u \neq -1$ and $u \notin \mathbb{R}$. Using the same pattern like the construction of the complex numbers $\mathbb{C}$ in the last section, we build the *hyperbolic numbers* $\mathbb{H}$[13] now in two different ways: first by means of a special multiplication (table) for the 2D vector space $\mathbb{R}^2$ and second using the CLIFFORD algebra $c\ell(1,1)$.

2.1 IH as algebraic structure

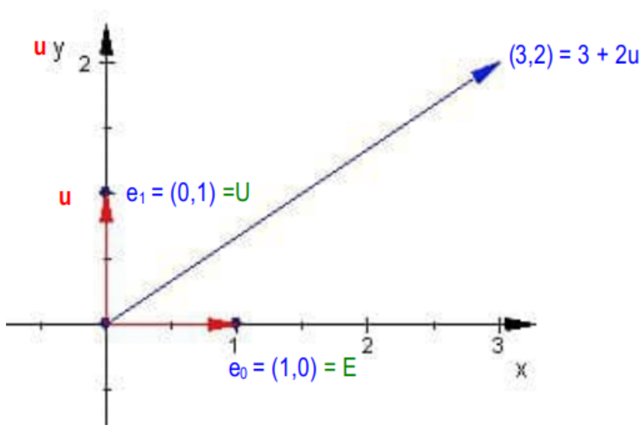


Figure 6: The hyperbolic number plane $\mathbb{H}$.
Blue: The hyperbolic number $3 + 2u$. Basis $\{e_0, e_1\}$ of $\mathbb{R}^2$.
Red: The unipotent u with $u^2 = 1$, but $u \neq \pm 1 \in \mathbb{R}$.
Green: The hyperbolic basis $\{1, u\}$ alias $\{e_0, e_1\}$ or $\{E, U\}$.

To construct the hyperbolic numbers $\mathbb{H}$ as an $\mathbb{R}$–algebra, we extend the real vector space $\mathbb{R}^2$ to include the unipotent element u together with a new third operation $\boxdot\colon \mathbb{R}^2 \times \mathbb{R}^2 \to \mathbb{R}^2$ by means of a 'hyperbolic multiplication table' for it. Analog to the reconstruction of the complex numbers $\mathbb{C}$, we will use the *bilinearity* of $\boxdot$ and describe its action on all 4 possible pairs of basis elements.

[13] This chapter is inpired by the presentations of Garret SOBCZYK in [12] and [13].

Here are the multiplication rules for the hyperbolic multiplication $\boxdot$, acting on the two basis vectors $e_0 \overset{def}{=} (1,0)$[14] and $e_1 \overset{def}{=} (0,1)$ of the vector space $\mathbb{R}^2$:

$$\begin{aligned} e_0 \boxdot e_0 &= e_0 && (2.1)\\ e_0 \boxdot e_1 &= e_1 && (2.2)\\ e_1 \boxdot e_0 &= e_1 && (2.3)\\ e_1 \boxdot e_1 &= e_0 && (2.4) \end{aligned}$$

• If we translate the 4^{th} rule using the lexicon $\overset{e_0, e_1}{\mathbf{1,\ u}}$ into the language of $\mathbb{H}$, we get the desired relation $\overset{e_1 \boxdot e_1 = e_0}{u \boxdot u = u^2 = 1}$.

• If we put the above rules in an *hyperbolic multiplication table* for this algebra multiplication $\boxdot$ for $\mathbb{H}$, we have:

$$\begin{array}{c c} \boxdot & \begin{matrix} e_0 & e_1 \end{matrix} \\ \begin{matrix} e_0 \\ e_1 \end{matrix} & \begin{pmatrix} e_0 & e_1 \\ e_1 & e_0 \end{pmatrix} \end{array}$$

Definition. The *hyperbolic numbers* $\mathbb{H}$ are the elements of the 2–dimensional number plane $\mathbb{R}^2$ equipped with the three operations $(+, \cdot, \boxdot)$ and the unipotent element $u \in \mathbb{H}$ with $u \neq \pm 1$, but $u^2 = 1$. We write: $\mathbb{H} \overset{def}{=} (\mathbb{R}^2, +, \cdot, \boxdot)$.

Exercise 2.1. Calculate $(1e_0 + 2e_1) \boxdot (-2e_0 + 3e_1)$ using the multiplication table for hyperbolic numbers. *Solution*:

$$\begin{aligned} (1e_0 + 2e_1) \boxdot (-2e_0 + 3e_1) &= -2e_0 \boxdot e_0 + 3e_0 \boxdot e_1 - 4e_1 \boxdot e_0 + 6e_1 \boxdot e_1 \\ &= -2e_0 + 3e_1 - 4e_1 + 6e_0 \\ &= 4e_0 - 1e_1 = 4 \cdot (1,0) - 1 \cdot (0,1) = (4,-1) \end{aligned}$$

2.1.1 Implementing $\mathbb{H}$ alias the binarions in Eigenmath

Remark. Using the abbreviation $E := e_0, U := e_1$ the hyperbolic multiplication table reads

$$\begin{array}{c c} \boxdot & \begin{matrix} E & U \end{matrix} \\ \begin{matrix} E \\ U \end{matrix} & \begin{pmatrix} E & U \\ U & E \end{pmatrix} \end{array}$$

and looks like a 2D analogue of the 4D table for the quaternions, which we discuss in the next chapter. Therefore the name '*bi*'narions for the hyperbolic numbers.
In order to calculate in the new algebra $\mathbb{H} = (\mathbb{R}^2, +, \cdot, \boxdot)$, we have to translate the table above into Eigenmath command language.

[14] For systematically reasons, which will become clear later on, we again do not use the usual notation $\{e_1, e_2\}$ for the two basis vectors.

```
##### HYPERBOLICS H   -- alias: the Binarions
tty=1

e0 = (1,0)            -- basis of H
e1 = (0,1)

T = ((e0,e1),
     (e1,e0))         -- Hyperbolics multiplication _T_able

M(x,y) = dot(x,T,y)   -- billinear operation on H

"Checking binarions multiplication table."
check(M(e0,e0)=e0)
check(M(e0,e1)=e1)
check(M(e1,e0)=e1)
check(M(e1,e1)=e0)
"pass"                -- (0) check ok? yes!

M(e1,e1)              -- (1) with e1=u we see: u^2=1
M(1e0+2e1,-2e0+3e1)   -- (2) checking Ex.
```

```
Checking binarions multiplication table.
pass
(1,0)
(4,-1)
```

Comment. Function $M \stackrel{def}{=} \boxdot$ realize the bilinear operation, i.e. the Multiplication of hyperbolic numbers.. The checks in lines (0) verifies, that the operation M implements the values of the $\mathbb{H}$–multiplication table and especially fulfills the desired relation $U^2 = e_1^2 = (1,0) \equiv 1$. Code line (2) verifies the result of Ex.2.1.
▷ *Click here to run the script.*

Exercise 2.2. (The algebraic characteristics of the hyperbolic number multiplication $\boxdot$) The multiplication $\boxdot$ is prescribed on its values on the finite 4 element table. Therefore it suffices to check its properties like *commutativity*, *associativity*, *distributivity* etc. on the muliplication table. E.g.

```
-- define 3 arbitrary hyperbolic numbers ('binarions')
x = (x0,x1)
y = (y0,y1)
z = (z0,z1)

"Is multiplication M of hyperbolic numbers commutative?"
test( M(x,y)=M(y,x), "yes","no")

"Is multiplication M alternative?"
test( and(M(M(x,x),y)=M(x,M(x,y)),
          M(M(y,x),x)=M(y,M(x,x))),"yes","no")
```

EIGENMATH output: commutative: yes alternative: yes
▷ *Click here to run the script.*

a. Check the *associativity* of $M \equiv \boxdot$.
b. Check the *distributivity* of M.

Exercise 2.3. (An explicit formula for the hyperbolic number multiplication $\boxdot$)
We know the explicit formula $(a+bi)\star(c+di) = (ac-bd)+(ab-cd)i$ for the complex multiplication $\star$. Derive a similar formula for the hyperbolic multiplication $\boxdot$ by EIGENMATH.

Solution. First, let's spend a bit syntactic sugar and set $E \overset{def}{=} e_0$ and $U \overset{def}{=} e_1$ to get a similar appearance of hyperbolic numbers like the complex one's, i.e. $\genfrac{}{}{0pt}{}{e_0, e_1}{\text{E, U}}$ and $\genfrac{}{}{0pt}{}{\mathbb{C}\colon z = a\cdot 1 + b\cdot i}{\mathbb{H}\colon w = a\cdot E + b\cdot U}$.

```
E = (1,0)              -- _E_mbedding of R in H
U = (0,1)              -- the _u_unipotent - the analogue to i in C
                       -- H  multiplication rules:
                       -- E*E=E, E*U=U, U*E=U, U*U=E == 1
T = ((E, U), (U,E))
M(x,y) = dot(x,T,y)    -- multiplication as bilinear operation on H

                       -- two arbitrary hyp.numbers:
x = a*E+b*U
y = c*E+d*U
y
M(x,y)                 --  explicit term for hyperbolic multiplication
```

EIGENMATH output: `y=(c,d)` `M(x,y)=(a c + b d, a d + b c)`
▷*Click here to run the script.*

Using the explicit formula for M we can forget about the construction of the algebra $\mathbb{H}$ by means of a multiplication table and think of the hyperbolic numbers as $(\mathbb{R}^2, +, \cdot, \boxdot)$ with

$$\boxdot\colon \mathbb{R}^2 \times \mathbb{R}^2 \to \mathbb{R}^2 \tag{2.5}$$

$$(a,b),(c,d) \to (a,b)\boxdot(c,d) \overset{def}{=} (a\cdot c + b\cdot d, a\cdot d + b\cdot c) \tag{2.6}$$

Exercise 2.4. (The hyperbolic multiplication)
a. Write the following function in a toolbox named `hyBox.txt` for future use:

```
# multiplication of hyperbolic numbers
hymult(x,y) = (x[1]*y[1] + x[2]*y[2], x[1]*y[2] + x[2]*y[1])
hymult(e1,e1)              -- result: (1,0) == 1
```

b. Solve Ex.2.1 using `hymult(.)`.

2.1.2 Implementing specific user functions for IH

From now on we use the following lexicon for calculation in the hyperbolic number plane $\mathbb{H}$ with $E = (1,0)$ and $U = (0,1)$:

	Math	EIGENMATH
standard basis	$(1, u)$	`(E,U)`
arbitrary hyperbolic number	$w = x + yu$	`w=x*E+y*U`

Definition. (*the hyperbolic length*)
The *hyperbolic norm* (modulus, length) of $w = x + y \cdot u \in \mathbb{H}$ is defined as the real number

$$|w|_h \stackrel{def}{=} \sqrt{|x^2 - y^2|} \tag{2.7}$$

The set $H^1 \stackrel{def}{=} \{w \in \mathbb{H} \mid x^2 - y^2 = 1\}$ is called the *unit hyperbola.*

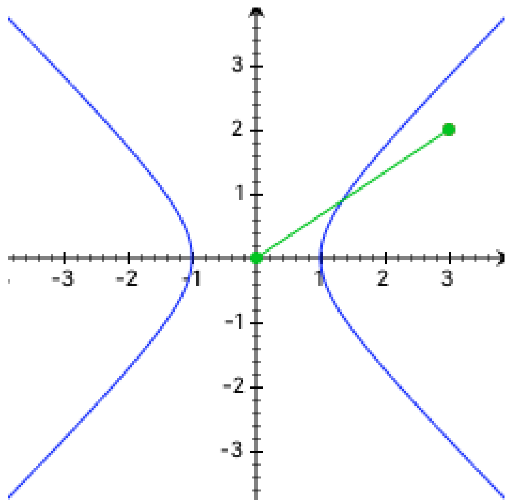


Figure 7: Blue: unit hyperbola $H^1 \subset \mathbb{R}^2$ with equation $x^2 - y^2 = 1$.
Green: the hyperbolic number/vector $w = 3 + 2 \cdot u = (3, 2)$
with hyperbolic length $|3 + 2 \cdot u|_h = \sqrt{5} \approx 2.24$.

Exercise 2.5. (The norm of a hyperbolic number)
Let $w = x + yu \in \mathbb{H}$ be a arbitrary hyperbolic number.
a. Define a function `hyno(w)` to calculate the *hy*perbolic *norm* alias the length of w as given in the definition previously.
b. Calculate the hyperbolic lenght of $w = 3 + 2u$ in Fig.7 by paper'n pencil and *hyno*(..).
c. Determine the 'hyno' of the points $P = 1 + 0u$, $Q = -1 + 0u$ and $R = -1 + u$.
d. What is the hyperbolic distance betwenn $A = 2E - 3U$ and $B = 1E - 2U$?
▷ *Click here to see the solution.*

Exercise 2.6.
a. Put the following functions for hyperbolic numbers in the toolbox `hyBox.txt`:

```
hyreal(w)   = w[1]                -- REAL part of hyperbolic number w
hyunip(w)   = w[2]                -- UNIPotent part of hyperbolic number w
hyconj(w)   = (w[1],-w[2])        -- hyperbolic CONJugate of w
hyinv(w)    = 1/(w[1]^2-w[2]^2)*hyconj(w)  -- hyperbolic INVerse of w
hyquot(v,w) = hymult( v, hyinv(w))         -- QUOTient of v and w
hynorm(w)   = sqrt(abs(w[1]^2-w[2]^2))     -- hyperbolic NORM of w
```

b. Calculate the hyperbolic length of $w = 3 + 2u$ of Fig.7.

```
-- EIGENMATH solution
do( E=(1,0), U=(0,1) )
w = 3E+2U
hynorm(w)            -- output: 5^1/2 = 2.24
abs(w)               -- the Euclidean length of w in R^2 is 13^1/2=3.6 !
```

▷ *Click here to run the script.*
c. Determine for w the hyperbolic real part, its unipotent part, its conjugate, the inverse.
d. Calculate the hyperbolic quotient of $w = 3 + 2u$ and $v = 1 + 2u$.
e. Try to calculate the hyperbolic quotients of $w = 3+2u$ and $v = 1+1u$ resp. $w/(2-2u)$.

2.1.3 Isotropic points in IH

Ex.2.6 showed, that there are hyperbolic numbers not equal $0 \cdot E + 0 \cdot U = (0, 0)$, for which the calculation of the hyperbolic quotient ejected a '*division by zero*' error message. We observed, that the denominators lie on the diagonals of the coordinate system. Those points are called *isotropic*.

Definition. A hyperbolic number $w \neq 0$ s called *isotropic*, if its hyperbolic length is zero, i.e. $|w|_h = 0$.

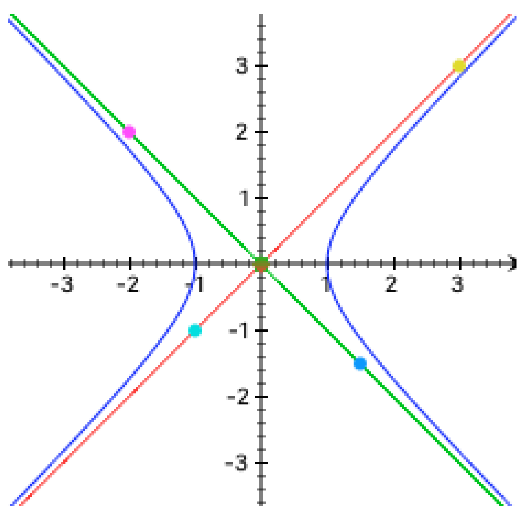


Figure 8: Blue: unit hyperbola $H^1 \subset \mathbb{R}^2$ with equation $x^2 - y^2 = 1$.
Red: the main diagonal $y = x$ as isotropic point line
Green: the second diagonal $y = -xx$ as isotropic points
Cyan ..: isotropic points $3 + 3u, -2 + 2u, -1 - u, 1.5 + 1.5u$.

Exercise 2.7. (isotropic points)
a. Verify, that the points (hyperbolic numbers) in Fig.8 are isotropic.
b. Prove: all points on the diagonals $y = \pm x$ are isotropic.

Remark. This phenomenon of the existence of isotropic subvector spaces with respect to the hyperbolic norm leads to a new non-Euclidean geometry, the LORENTZian geometry on $\mathbb{R}^2$. It plays a great role in Special Relativity, see [13, pp. 15 ff].

2.1.4 Problems.

P7. A scene for the hyperbolic number plane.

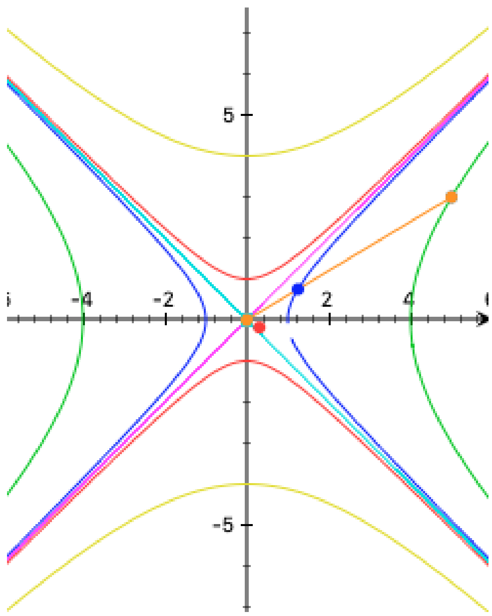


Figure 9: Blue: unit hyperbola $H_+^1 \subset \mathbb{R}^2$ with equation $x^2 - y^2 = +1$.
Red: unit hyperbola $H_-^1 \subset \mathbb{R}^2$ with equation $x^2 - y^2 = -1$.
Green: hyperbola $H_+^4 \subset \mathbb{R}^2$ with equation $x^2 - y^2 = +16$.
Yellow: hyperbola $H_-^4 \subset \mathbb{R}^2$ with equation $x^2 - y^2 = -16$.
Cyan: isotropic line $y = -x$. Magenta: isotropic line $y = x$.
$w = 5 + 3u \in \mathbb{H}$, Blue: $w/|w|_h$ Red: w_h^{-1} =`hyinv(w)`.

a. Calculate the coordinates of the blue point $w/|w|_h \in H_+^1$ by paper'n pencil. Check your result with EIGENMATH.
b. Determine the hyperbolic distance and the Euclidean distance of w from the origin.
c. Determine the coordinates of the red point $w^o := w/|w|_h \in H_+^1$ by paper'n pencil. Check your result with EIGENMATH. Determine the hyperbolic distance of w^o from the origin and from its 'father' w.
c. Determine the coordinates of the hyperbolic inverse w_h^{-1} of w by paper'n pencil. How long is the distance of this inverse to w? Check your results with EIGENMATH.
d. Verify the result of c. by calculating the hyperbolic product $w_h^{-1} \boxdot w$.

P8. Alternativ formula for the hyperbolic conjugate.
a. Argue, why the following function `Cj(.)` calculates the hyperbolic inverse.

```
-- EIGENMATH
do( E=(1,0), U=(0,1) )
w = 3E+2U
Cj(x) = 2*dot(x,E)*E - x
```

b. Calculate the hyperbolic conjugates of the three points (hyperbolic numbers) of Fig. 9. Check your results with *hyconj*(.).
▷ *Click here to run the script.*

P9. The inner and outer product in $\mathbb{H}$.
Let $U, V \in \mathbb{H}$ be two arbitrary hyperbolic numbers.
a. Define the functions `hyinp(U,V)` and `hyoutp(U,V)` to compute the *inner* resp. *outer product* of two hyperbolic numbers $U, V \in \mathbb{H}$ through

```
hyinp(U,V)  = U[1]*V[1] - U[2]*V[2]   -- inner product alias scalar product
hyoutp(U,V) = U[1]*V[2] - U[2]*V[1]   -- outer product
```

- Calculate the inner and outer product of $U = 3E - 4U$ and $W = -4E + 3U$.
- Calculate the inner and outer product of $A = 2E + 3U$ and $B = 1E - 2U$.
- Calculate the inner and outer product of w and $\bar{w}$ of Fig. 59.

b. Prove: `hyinp(U,W) = hyre( hymult( hyconj(U),W))`.
Formulate this formula in mathematical language.
c. Formulate and prove a similar formula for the outer product.
d. Find a hyperbolic number $w^{\perp}$, which is hyperbolic orthogonal to w.
▷ *Click here to see the solution.*

P10. Realization of $\mathbb{H}$ as matrix algebra.
Using the correspondence

$$\mathbb{H} \longrightarrow \mathbb{R}^{2\times 2}_{sym} \tag{2.8}$$
$$x + y \cdot u \mapsto \begin{bmatrix} x & y \\ y & x \end{bmatrix} \tag{2.9}$$

the hyperbolic numbers can be identified with the symmetric 2×2 matrices with equal diagonal entries.
a. Why is this assignment an isomorphism?
b. The hyperbolic numbers $w1 = 2 + 3u$ and $w2 = 3 - 5u$ are represented via the isomorphism (2.9) through $W1 = \begin{bmatrix} 2 & 3 \\ 3 & 2 \end{bmatrix}$ and $W2 = \begin{bmatrix} 3 & -5 \\ -5 & 3 \end{bmatrix}$.
Calculate the values of $w1 + w2, w1 - w2, 2 \cdot w1$ by paper'n pencil and EIGENMATH.

Partial *solution*:

```
W1=((2,3),(3,2))
W2=((3,-5),(-5,3))
W1+W2
W1-W2
2*W1
```

c. Verify, that the hyperbolic multiplication $\boxdot$ alias $hymult()|_{Eigenmath}$ corresponds to the usual matrix multiplication $\star$ of symmetric 2×2 matrices, i.e.

$$hymult((2,3),(3,-5)) \overset{Math}{=} W1 \star W2 \overset{Eigenmath}{=} dot(W1, W2)$$

Example:

```
W1=((2,3),(3,2))
W2=((3,-5),(-5,3))
dot(W1,W2)                  -- corresponds to (2+3u) hymult (3-5u)
```

d. Write corresponding EIGENMATH functions for the hyperbolic real part, unipotent part, the hyperbolic multiplication, quotient, conjugate, norm and the hyperbolic inverse of a hyperbolic number. Here is a start: ▷*Click here to start the start.*

```
Hconj(z)=((z[1,1],-z[1,2]),(-z[2,1],z[2,2]))
Hnorm(z)=sqrt(abs(z[1,1]^2-z[1,2]^2))
Hinv(z)=1/(z[1,1]^2-z[1,2]^2)*Hconj(z)

# TEST
W1=((2,3),(3,2))
W2=((3,-5),(-5,3))
Hconj(W1)
Hinv(W1)
```

P11. More functions for IH.
Let $U, V \in \mathbb{H}$ be two arbitrary hyperbolic numbers.
If you like it: derive functions for the hyperbolic polar form, the hyperbolic angle (argument) etc.

$\boxdot$

Summary: We have constructed a new algebra IH inside the Euclidean plane $\mathbb{R}^2$ by means of a special multiplication table for the basis vectors $span_{\mathbb{R}}\{e_0, e_1\}$ alias $span_{\mathbb{R}}\{E, U\}$. With this hyperbolic multiplication we got the desired relation $U^2 = 1$ to have a root of $\sqrt{+1}$, not being $\pm 1 \in \mathbb{R}$. We were able to define the crucial IH–typical functions like hyperbolic conjugate, hyperbolic imaginary part, hyperbolic reciprocal, hyperbolic norm etc. in this setting, too.
We did not give EIGENMATH formulas e.g. for the hyperbolic polar form, because we will show these constructs in a more general setting – viewing IH as a special CLIFFORD algebra. This is the done in the next section.

2.2 IH as split-complex numbers – the CLIFFORD algebra $c\ell(1,1)$

In this section we reconstruct the hyperbolic numbers IH using the same universal recipe, which we used to represent the algebra $\mathbb{C}$ and which is a second example of an Geometric algebra "GA".

Here we use the EIGENMATH package `EVA2.txt` for the second time. We want to broaden our experience in the use of `EVA` as another possibility to calculate with hyperbolic numbers in a more straight way.

2.2.1 A look at the 4D-CLIFFORD algebra $c\ell(1,1)$

Let's look first at $c\ell(1,1)$, a part of it will soon become an algebraic modeling of the hyperbolic numbers IH:

```
run("downloads/EVA2.txt")  -- (1)
tty=1
cl(1,1)                    -- (2)

do( print(e0),print(e1),print(e2),print(e12))  -- (3)

E = e0        -- (4) E mbedding of R in cl(1,1), i.e. first entry
E
U = e12       -- (5) U will play the role of the unipotent u
U

gp(E,E)
gp(U,U)       -- (6) U is indeed unipotent resp gp
```

```
(+,-)
e0 = (1,0,0,0)
e1 = (0,1,0,0)
e2 = (0,0,1,0)
e12 = (0,0,0,1)
E = (1,0,0,0)
U = (0,0,0,1)
(1.0,0.0,0.0,0.0)
(1.0,0.0,0.0,0.0)
```

Comment. At first glance, all looks similar to the $c\ell(2,0)$ construction of $\mathbb{C}$ in §1.4.1. That's good, because we do not have to learn a new vocabulary and may use the same notations, that we're used to. *But watch*: The call $cl(1,1)$ in code line (2) of the constructor function $cl(..)$ of the `EVA2` package give the output $(+,-)$! This means, that the norm of $cl(1,1)$ has the *signature* $(+,-)$, i.e. the norm has now the term $\sqrt{+x^2-y^2}$. Therefore the name 'split-complex'. In line (3) we list the basis vectors $span_{\mathbb{R}}\{e0, e1, e2, e12\}$ of $c\ell(1,1) \sim \mathbb{R}^4$, which have the expected canonical coordinates of the 4D vector space $\mathbb{R}^4$. In line (4) we embed the real number line $\mathbb{R}$ by means of E and his multiples into $c\ell(1,1)$.
Line (6) is crucial and shows, why we will later chose a part of $c\ell(1,1)$ as model for IH: it verifies the characteristic feature $u^2 = 1$ of the unipotent element $u \in$ IH of the hyperbolic numbers is fulfilled with respect to the geometric product of $c\ell(1,1)$, i.e.

$$gp(U,U) = U^2 = (1,0,0,0) \equiv 1$$

∘ This observation (6) leads to a realization of IH inside the CLIFFORD algebra $c\ell(1,1)$.
∘ For the moment we may think of the geometric product gp as given through the $4\times4 = 16$ entry multiplication table for the hyperbolic multiplication $\boxdot$ for the algebra IH in the last section or as a re–construction of the function $hymult(.)$ inside $c\ell(1,1)$.
▷ *Click here to invoke cl(1,1).*

Exercise 2.8.
First, do some free experiments in the 4D algebra $c\ell(1,1)$. – A possible *Solution*.

```
run("EVA2.txt") # load package EVA
cl(1,1)                    # specify the Clifford Algebra
do( E=e0, U=e12 )          # set (E,U) 2D sub-algebra

A = 1e0+2e1+3e2+4e12   -- an element in full cl(1,1), but not in H
B = 1E+2e1+3e2+4U      -- the same in other notation
A
B

a =  3E+2U             -- an element in H
b = -2E+U

a+b                    -- normal 4D addition in H
a-b                    -- normal 4D subtraction in H
2a+3b                  -- usual linear combination
magnitude(a)           -- hyperbolic length, see Fig.2
abs(a)                 -- Euclidean length

inp(U,U)               -- feel at home
```

▷ *Click here to invoke this script.*

2.2.2 IH as a 2D sub-algebra of the 4D Clifford algebra $c\ell(1,1)$

After Ex.2.8 we feel immediately at home in this 4D vector space $c\ell(1,1)$. The functions *magnitude*, *normalize*, *inp*, and *gp* (geometric product) are available by means of the package `EVA2` and work as expected. Therefore we make the

Definition: The hyperbolic number plane IH *is the the 2D sub-algebra*

$$\mathbb{H} := (span_{\mathbb{R}}\{e0, e12\}, +, \cdot, \boxdot)$$

of $c\ell(1,1)$ with algebra multiplication $\boxdot$.

Math IH	Eigenmath EVA2 $c\ell(1,1)$
$A \boxdot B$	`gp(A,B)`

.. and we can use the same EVA2–functions as for the complex numbers:

	Math	EIGENMATH EVA2
geometric product	`A B`	`gp(A,B)`
inner/scalar product	$A \bullet B$	`inp(A,B)`
outer product	$A \wedge B$	`outp(A,B)`
Clifford conjugation	$\bar{B}$	`cj(B)`
inverse	$1/B$	`inverse(B)`
magnitude	$\|B\|$	`magnitude(B)`
normalize	$\frac{B}{\|B\|}$	`normalize(B)`

• The CLIFFORD algebra functions are usable also for the hyperbolic domain, they are noted with an ending `1` to distinct them from the EIGENMATH build–in functions for the complex domain, e.g. `imag1`, `real1`, `polar1`, `rect1`, `exp1`, `log1`, `sqrt1`, `power1`, `sin1`, `cos1`, `tan1`, `sinh1`, `cosh1`, `tanh1`, `asin1`, `acos1`, `atan1`, `atanh1`, ..

Exercise 2.9. (Using $c\ell(1,1)$ for arithmetic with hyperbolic numbers)
a. Re–do *Ex.2.5* and *Ex.2.6* calculating in the CLIFFORD algebra $c\ell(1,1)$ using `EVA2`.
b. Re–do problems *P.10* and *P.12* calculating in the CLIFFORD algebra $c\ell(1,1)$ using `EVA2`.

2.2.3 The hyperbolic polar form in $\mathbb{H} \sim c\ell(1,1)$

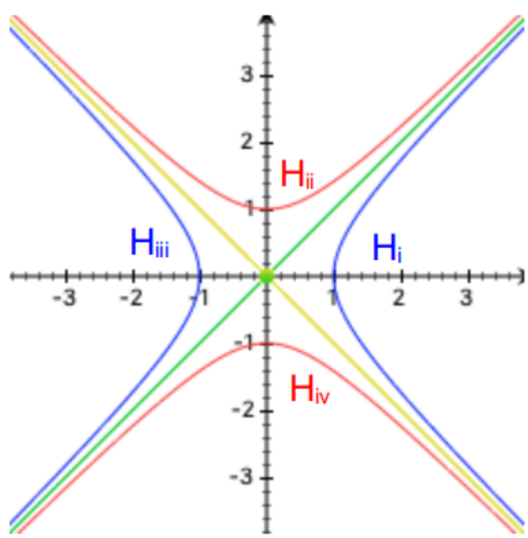


Figure 10: Blue: *unit* hyperbola $H^1_+ \subset \mathbb{R}^2$ with equation $x^2 - y^2 = +1$.
Red: *conjugate* hyperbola H^1_- with equation $x^2 - y^2 = -1$.
Green: first asymptote with equation $y = x$.
Yellow: second asymptote with equation $y = -x$.
Four hyperbolic quadrants $H_i, H_{ii}, H_{iii}, H_{iv}$, demarcated by the two asymptotes..

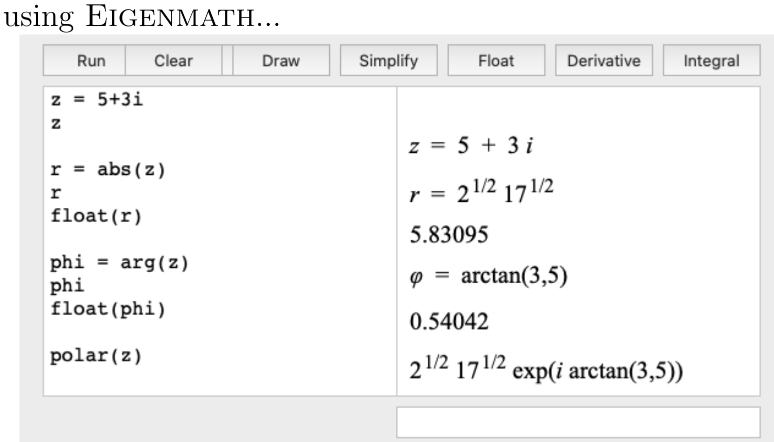
For the use of the *hyperbolic* polar form `polar1`, we divide the hyperbolic plane $\mathbb{H}$ in 4 hyperbolic quadrants $H_i, H_{ii}, H_{iii}, H_{iv}$, see Fig.10, *with the asymptotes as axes*. The set of all points $w \in \mathbb{H}$ in the hyperbolic plane that fulfill the relation $\|B\| = |w|_h = \rho$ for an hyperbolic radius $\rho > 0$ is a four branched hyperbola. For a hyperbolic number $w = x+yu$ we therefore have

$$polar1(w) = \begin{cases} +\rho \cdot exp1(\phi \cdot u) & : \text{ for } w \text{ in } H_i \quad \text{(pol1)} \\ -\rho \cdot exp1(\phi \cdot u) & : \text{ for } w \text{ in } H_{iii} \quad \text{(pol3)} \end{cases}$$

resp.

$$polar1(w) = \begin{cases} +\rho u \boxdot exp1(\phi \cdot u) & : \text{ for } w \text{ in } H_{ii} \quad \text{(pol2)} \\ -\rho u \boxdot exp1(\phi \cdot u) & : \text{ for } w \text{ in } H_{iv} \quad \text{(pol4)} \end{cases}.$$

Example. (the hyperbolic polar form of $w = 5+3u \in \mathbb{H}$, see [13, p. 6]) For analogy and contrast we look at the point $(5,3) \in \mathbb{R}^2$ of the Euclidean plane from the viewpoints of $\mathbb{C}$, i.e. $z = (5,3) = 5+3i$ and $\mathbb{H}$, i.e. $w = (5,3) = 5+3u$.
$(5,3) = z = 5+3i \in \mathbb{C}$: First we calculate the polar form of w seen as *complex* number.
The radius is $r = \sqrt{+5^2+3^2} = \sqrt{34} \approx 5.83$.
The argument (angle) is $\varphi = \arctan(3/5) =\approx 0.54042$, i.e. $\varphi \approx 31^o$.
Therefore $polar(z) = \sqrt{34} \cdot exp(0.54042 \cdot i)$.
Let's control it using EIGENMATH...



.. and by means of a plot:

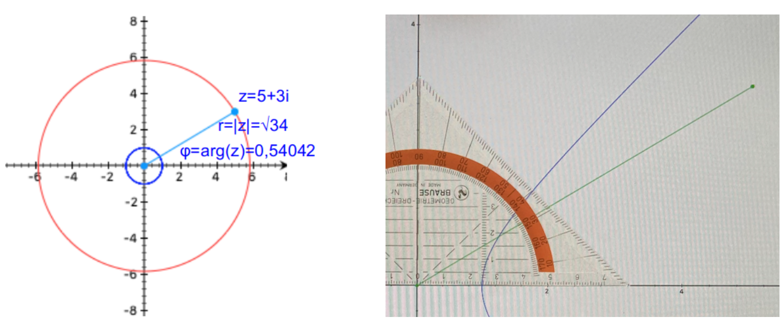
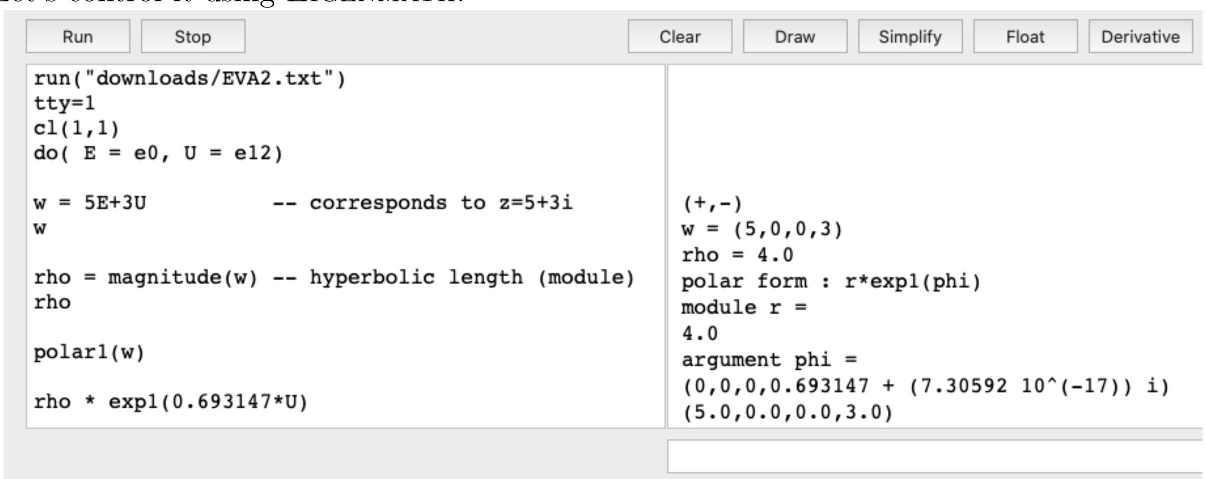


Figure 11: Blue: unit circle S^1 with equation $x^2+y^2=1$.
Red: circle S^r of radius $r = \sqrt{34} \approx 5.83$
Cyan: the complex number $z = 5+3i$ with $\varphi = \measuredangle = 31^o$.

$(5,3) = w = 5 + 3u \in \mathbb{H}$: Now we calculate the polar form of w as a *hyperbolic* number.
The hyperbolic radius is $\rho = \sqrt{+5^2 - 3^2} = \sqrt{16} = 4$.
The hyperbolic argument (angle) is $\phi \overset{H_i}{=} arctanh(3/5) \approx 0.6931$. No degree!
Therefore the hyperbolic polar form is $polar1(z) \overset{H_i}{=} 4 \cdot exp(0.6931 \cdot u)$.
Let's control it using EIGENMATH:




```
run("downloads/EVA2.txt")
tty=1
cl(1,1)
do( E = e0, U = e12)

w = 5E+3U           -- corresponds to z=5+3i
w

rho = magnitude(w) -- hyperbolic length (module)
rho

polar1(w)

rho * exp1(0.693147*U)
```

```
(+,-)
w = (5,0,0,3)
rho = 4.0
polar form : r*exp1(phi)
module r =
4.0
argument phi =
(0,0,0,0.693147 + (7.30592 10^(-17)) i)
(5.0,0.0,0.0,3.0)
```

Comment. The hyperbolic number $w = 5 + 3u$ is represented in $\mathbb{H} \sim c\ell(1,1)$ as a 4D vector, where only the 1^{st} and the 4^{th} entry is used, therefore working in a 2D sub-algebra. The `EVA` function *magnitude* returns the hyperbolic length of w alias the *hyperbolic radius*. The complete polar form of w is calculated by the `EVA` function *polar1*, which returns the hyperbolic angle (alias the *hyperbolic argument*) as the real part 0.693147 of the 4th entry. Because for $w \in H_i$ we verify the result using formula (pol1) and get back the rectangular form of w. Ok. ▷*Click here to invoke this script.*
Let's look at the geometric situation.

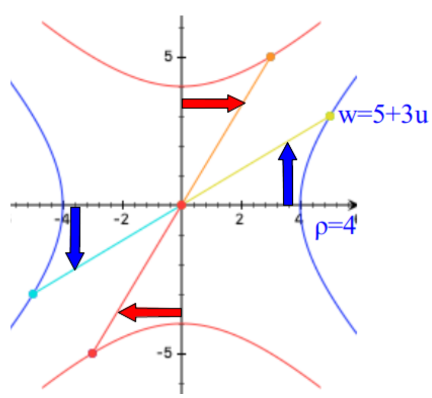


Figure 12: Blue: hyperbola H_i^4, H_{iii}^4: $x^2 - y^2 = 16$. w is a point on it. Red: conjugate hyperbola $H_i^4 i, H_{iv}^4$: $x^2 - y^2 = -16$ Cyan: the hyperbolic number $w = -5 - 3u$ and its three branch "brothers" with same hyperbolic angle (argument).

Exercise 2.10. (branch points and their hyperbolic polar forms)
a. Verify, that the hyperbolic conjugate of $w = 5+3u$ is $cj(w) = 5-3u$. Use EIGENMATH. Calculate its hyperbolic polar form.
b. There are 3 more points on the 4 branched hyperbola $H^{\rho=4}$, to be seen in Fig. 12. Give their polar1 forms and their rectangular forms. *Hint*: use symmetry. Here is a start.

```
run("EVA2.txt")
cl(1,1)
do( E = e0, U = e12)
w = 5E+3U

w3 = -magnitude(w)*exp1(0.693147 U)      -- use formula (pol3) with ..
w3                                       -- .. same hyperbolic angle !

w2 = +gp(magnitude(w)*U, exp1(0.693147 U))
w2                        -- use (pol2), because w2 on 2nd branch
```

▷*Click here to invoke this script.*

Exercise 2.11. (The hyperbolic polar form)
a. Express each of the following the hyperbolic numbers (points) in hyperbolic polar form:
$A = 2E + \sqrt{12}U, \quad B = -5E + 5U, \quad C = -\sqrt{6}E - \sqrt{6}U, \quad D = -3U$
and plot them on the hyperbolic number plane. Use EIGENMATH.
b. Interpret the points of Ex.a. as complex numbers, using their real and imaginary parts. Plot the points on the *complex* number plane. Use paper'n pencil and/or EIGENMATH.

Exercise 2.12. (An EIGENMATH function for the hyperbolic polar formulas)
Bundle the four branched separated hyperbolic polar formulas $(pol1), (pol2), (pol3), (pol4)$ in one function EIGENMATH `polH(w)`, which checks beforehand to which branch the hyperbolic number w belongs and than choses the appropriate formula (poli).
Check your function on the 4 points of Ex.2.11.a.

> In section 2.2.3 we have read of the value of the hyperbolic argument of a hyperbolic number at the output of `EVA2`–function `polar1(.)`, see the screenshot of the EIGENMATH session before Fig.12. This hyperbolic angle was 'hidden' in real part of the complex number of the 4th coordinate entry of the result. We therefore will give some possibilities of a direct calculation of the hyperbolic argument (angle). This will need a little knowledge from calculus, e.g. [9, p. 500 ff].

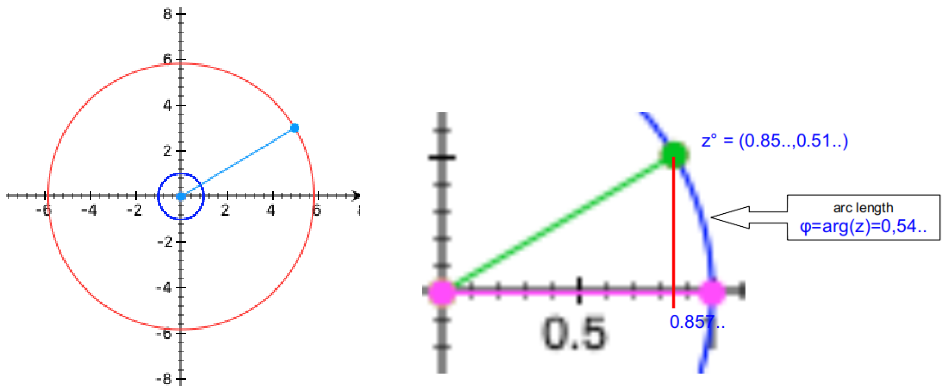


Figure 13: Blue: unit circle $S^1: x^2+y^2=1$.
Cyan: complex number $z=5+3i$.
Right: a microscopic view at the complex number angle.
Best mental image as length of the arc from • ⌒ • in rad.

2.2.4 The hyperbolic angle (argument) in $\mathbb{H} \sim c\ell(1,1)$

For analogy and contrast we look again at the point $(5,3) \in \mathbb{R}^2$ of the Euclidean plane from the viewpoints of $\mathbb{C}$, i.e. $z=(5,3)=5+3i$ and $\mathbb{H}$, i.e. $w=(5,3)=5+3u$.
$(5,3)=z=5+3i\in\mathbb{C}$: First we calculate again the `arg`(ument, angle) of w interpreted *as complex* number z in an alternative way *to gain an appropriate mental image of the concept* 'argument of z'. We calculate this angle as a proportional piece of the plane unit circle curve $S^1: x^2+y^2=1$ as seen in the microscopic view in Fig.13.right:

```
Run   Stop                                          Clear   Draw

1/abs((5,3))*(5.,3.)          -- (1) project z=5+3i on S^1

g(t) = (t, sqrt(1-t^2))       -- (2) parametrize the S^1
h = d(g(t),t)                 -- (3) velocity vector at t
h
h = sqrt(simplify(h[1]^2 + h[2]^2))
h
L = defint(h, t, 0.857493, 1) -- (4) arc length alias arg
float(L)

arg(5+3i)
float
```

$$\begin{bmatrix} 0.857493 \\ 0.514496 \end{bmatrix}$$

$$h = \begin{bmatrix} 1 \\ -\dfrac{t}{\left[-t^2+1\right]^{1/2}} \end{bmatrix}$$

$$h = \frac{1}{\left[-t^2+1\right]^{1/2}}$$

0.540419

arctan(3,5)

0.54042

▷ *Click here to invoke this script.*

Comment. In the EIGENMATH realization, we first (1) calculate the normalized point

$z^\circ = \frac{z}{|z|} \in S^1$, i.e. the coordinates of the green point in Fig.13. Second we define a parametrization $g\colon \mathbb{R} \to \mathbb{R}^2$of the unit circle, i.e. starting from the equation $x^2 + y^2 = 1$ we gain $y^2 = 1 - x^2$ and therefore $g(t) = (1, \sqrt{1-t^2})$. Third we calculate the argument of z realized as the arc length L of g between the magenta point $(1, 0)$ and the green point $z^\circ \approx (0.86, 0.51)$, i.e. we have the integral[15]

$$L \overset{(4)}{=} \int\limits_{x=0.8574}^{x=1} |g'(t)|dt \approx 0.5404 \overset{def}{=} arg(z)$$

We give a second interpretation of the complex arg (angle) as *area of the sector* $\sphericalangle = ((0, 0), (1, 0), z^\circ)$ ('trigonometric triangle'). If we express the unit circle S^1 in polar coordinates by the equation $r = f(\theta)$, together with the rays $\theta = \alpha$ to $\theta = \beta$ we enclose a region, whose area A is given by[16]

$$A = \frac{1}{2} \int\limits_{\theta=\alpha}^{\theta=\beta} f(\theta)^2 d\theta \overset{S^1}{=} \frac{1}{2} \int\limits_{\theta=0}^{\theta=0.5404} 1^2 d\theta = \frac{1}{2} \cdot arg(z)$$

We have the fact:

$$arg(z) = 2 \cdot \sphericalangle$$

i.e. *the double area of the sector of the unit circle with central angle θ equals $arg(z)$*.

```
# EIGENMATH
# Express the unit circle by equation r = f(theta) in polar coordinates.
f(theta) = r
r = 1
argZ = 2*1/2 * defint( f(theta)^2, theta, 0, 0.5404)
argZ        -- result 0.5404
```

▷ *Click here to invoke this script.*

In summa: besides the usual trigonometric definition of $arg(z) = arctanh(\frac{y}{x})$[17] we have two more possibilities to calculate the complex angle: first as length of an arc and second as area of a sector. This will help us to gain insight into the concept of the hyperbolic angle (argument) of an hyperbolic number.

Exercise 2.13. Use the arc length construction of the complex argument to program an alternative EIGENMATH– function `argC(z)` for the calculation of $arg(z)$.

[15] I thank G. WEIGT for a work around to calculate the intergral L with EIGENMATH.

[16] see e.g. [9, p. 502], where the authors also give a nice infinitesimal argument of this formula.

[17] cum grano salis, because one has to chose the correct order of nominator and denominator ..

$(5,3) = w = 5 + 3u \in \mathbb{H}$: Now we look at the argument (angle) of $w = x + yu$ *as hyperbolic number*. Because the formal definition via analogy $argH(w) = arctanh(\frac{y}{x})$[18] gives no geometric insight we try to go the alternative ways.

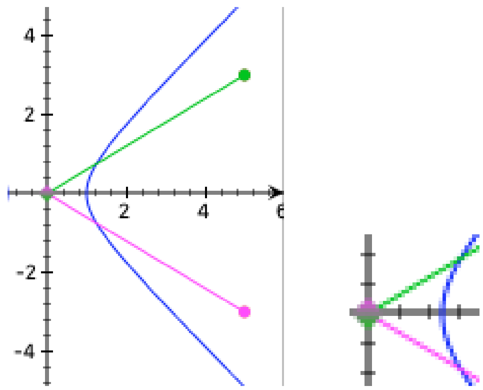


Figure 14: Blue: hyperbola $H_i^1: x^2 - y^2 = 1$.
Green: hyperbolic number $w = 5E + 3U$
and its hyperbolic conjugate $w_h^- = 5E - 3U$.
Right: a microscopic view at the hyperbolic angle of w.
Best mental image as area of the sector formed by $\circ \nearrow \searrow ($

1st: We calculate the hyperbolic arc length as length of the blue arc $\circ \nearrow \rightarrow ($ on the unit hyperbola H_i^1 using EIGENMATH.

Run Stop Clear Draw

```
run("Downloads/EVA2.txt")
cl(1,1)
do( E=e0, U=e12)
w = 5 E + 3 U
wm = magnitude(w)    -- (1) hyperbolic length of w is 4
wn = normalize(w)    -- (2) project w=5E+3U on H^1
wn                   --     =(5/4,..,3/4)

g(t) = (t, sqrt(t^2-1))   -- (3) parametrize H^1
h = d(g(t),t)                -- (4) velocity vector at
h = sqrt(simplify(h[1]^2 - h[2]^2))
h
Lh = defint(h, t, 1, 5/4) -- (4) arc length alias arg
Lh
float(Lh)
```

$$w_n = \begin{bmatrix} 1.25 \\ 0 \\ 0 \\ 0.75 \end{bmatrix}$$

$$h = \frac{i}{\left[t^2 - 1\right]^{1/2}}$$

$$L_h = i \log(2)$$

$$0.693147\, i$$

[18]cum grano salis, because one has to chose the correct order of nominator and denominator fitting to the correct hyperbolic quadrant H_i,..

▷ *Click here to invoke this script.*

Comment. Invoking `cl(1,1)` we first beam us into the hyperbolic plane. Because the hyperbolic length (magnitude) of w is $\|w\| = 4$, the normalized hyperbolic number is $w/\|w\| = 1/4 \cdot w$ and lies on H^1, i.e. the point $(\frac{5}{4}, \frac{3}{4}) \in \mathbb{R}^2$. Second we define a parametrization $g\colon \mathbb{R} \to \mathbb{R}^2$of the unit hyperbola, i.e. starting from the equation $x^2 - y^2 = 1$ (therefore the choice of `cl(1,1)`!) we gain $y^2 = x^2 - 1$ and therefore $g(t) = (1, \sqrt{t^2 - 1})$. Now we calculate the argument (hyperbolic angle) of w realized as the arc length Lh of g of the half sector ”(”, i.e. we have the integral

$$Lh \overset{(4)}{=} \int_{x=1}^{x=5/4} |g'(t)| dt = log(2) \overset{def}{=} \mathrm{argH}(z) \approx 0.6931$$

$\mathbf{2}^{nd}$: We use the fact

$$\mathrm{argH}(z) = \text{area of sector} \circ \begin{smallmatrix}\nearrow \\ \searrow\end{smallmatrix} ($$

The area of the sector of the unit hyperbola between w° and its conjugate $(w_h^-)^\circ$ equals $argH(z)$.

```
# EIGENMATH
-- 5/4 and 3/4 are the edges of a box
-- from the normalized wn on H^1. Therefore:

Lh = 5/4*3/4 - 2*defint(sqrt(x^2-1), x,1,5/4)
Lh

Lh = mag(defint( sqrt(-1/(x^2-1)), x,1,5/4))
Lh
float
```

▷ *Click here to invoke this script.*

Exercise 2.14. Use the trigonometric definition $argH(w) = arctanh(\frac{y}{x})$ for the hyperbolic argument in the quadrant H_i to program an EIGENMATH– function `arg2(z)`, which works for all four quadrants.

Exercise 2.15. Use the arc length definition via the integral to program a function `argH(w)` for the hyperbolic argument in the quadrant H_i. Try to make it work for all four hyperbolic quadrants.

Exercise 2.16. The `polar1` function of `EVA2` often gives back the argument of its input in complex number form. If you like to have only the real part, you can try the following function. Explain.

```
# EIGENMATH
run("EVA2.txt")
cl(1,1)
do( E = e0, U = e12)

phiH(c) = arctanh( mag( magnitude(imag1(c)) / magnitude(real1(c)))

w = 5E+3U
phiH(w)
```

▷ *Click here to invoke this script.*

2.2.5 Problems.

P12. The cubic equation $x^3 + 3ax + b = 0$.
The usefulness of the complex hyperbolic numbers is shown by G. SOBCZYK in [13, p. 13 ff.]. On p. 14 there is the solved example:
◦ *find the solutions of the reduced cubic equation* $x^3 - 6x + 4 = 0$.
Calculate the solutions by EIGENMATH.

P13. The Special relativity and LORENTZian Geometry.
SOBCZYK shows [13, p.15 ff.] the application of hyperbolic numbers $\mathbb{H}$ resp. $c\ell(1,1)$ in LORENTZian Geometry. There you will see e.g. the *spacetime distance* aka. the hyperbolic norm in action. Read about it. Use EIGENMATH and its package `EVA2` as companion.
In the abstract of his thesis BOROTA [2] writes:

> "The most useful aspect of spacetime [i.e. hyperbolic numbers, wL] numbers is in solving problems in the areas of special and general relativity. These areas deal with the notion of space-time, hence the name "spacetime numbers." [...] and show unusual features of spacetime arithmetic. A spacetime version of Euler's formula is then presented and then the solutions to the one-dimensional wave equation.

$\boxdot$

Summary: We have constructed the new algebra $\mathbb{H}$ of the hyperbolic numbers in the Euclidean plane $\mathbb{R}^2$ by means of a multiplication table for the basis vectors $span_{\mathbb{R}}\{e_0, e_{12}\}$. This way we get also the desired relation $u^2 = 1$ to have a root of $\sqrt{1}$, not being an element of $\mathbb{R}$. This construction is also known as the algebra of the *binarions*.
We did a second realization of the hyperbolic numbers $\mathbb{H}$ by invoking the CLIFFORD algebra $c\ell(1,1)$ of the EIGENMATH package `EVA2` and using a 2D sub-algebra of it. This package defines in this setting all crucial $\mathbb{H}$–typical functions like conjugate, imaginary part, reciprocal, norm, polar form etc.

Meanwhile the user should have gained a working knowledge of the hyperbolic numbers $\mathbb{H}$ and the use of the package `EVA2`. We now turn to a last low dimensional special example of a CLIFFORD algebra – the famous *quaternions*.

3 $\mathbb{H}$ – the quaternion numbers

Please: distinct the symbol $\mathbb{I}\mathrm{H}$ as notation of the hyperbolic numbers and the symbol $\mathbb{H}$ for the HAMILTONian quaternions.

Math concept	notation
hyperbolic numbers	$\mathbb{I}\mathrm{H}$ alias $c\ell(1,1)$
HAMILTON's quaternions	$\mathbb{H}$ alias $c\ell(3)^+$

Quaternions are a 4-dimensional number system. It is an extension of the complex number system. The (algebra) multiplication of quaternions is non-commutative, i.e. the order of the factors matters. Quaternions are used to describe and effectively do rotations of vectors in 3 dimensions. For an algebraic construction of HAMILTON's quaternions $\mathbb{H}$ in EIGENMATH by means of a multiplication table for the basis vectors, see G. WEIGT[19]. Therefore we will restrict our treatment of HAMILTON's quaternions on its realization in two other ways:
1st: as a 4D vector space enhanced with a special algebra multiplication,
2nd: as a special CLIFFORD algebra using EIGENMATH's package EVA2.

3.1 $\mathbb{H}$ as a 4D algebra with algebra multiplication $\circledcirc$

First we implement the quaternions $\mathbb{H}$ as a 4D algebra build on the vector space $\mathbb{R}^4$.

3.1.1 $\mathbb{H}$ as a 4D vector space $(\mathbb{R}^4, +, \cdot)$

```
# QUATERNIONs as vector space - NO use of EVA
tty=0                          -- compact notation OFF

E = (1,0,0,0)                  -- (1) basis
I = (0,1,0,0)
J = (0,0,1,0)
K = (0,0,0,1)

x = (x0,x1,x2,x3)              -- (2) a arbitrary quaternion as 4D vector
x
xQ = a*E + b*I + c*J + d*K     -- (3) arbitrary quaternion in basis E,I,J,K
xQ
y = (y1,y2,y3,y4)

addQ1(x,y) = (x[1]+y[1],x[2]+y[2],x[3]+y[3],x[4]+y[4])      --(3)

addQ(x,y) = (x[1]+y[1])*E + (x[2]+y[2])*I+
            (x[3]+y[3])*J + (x[4]+y[4])*K                   -- (4)
```

[19] see [18]. This demo of George was the inspiration for our construction of $\mathbb{C}$ and $\mathbb{I}\mathrm{H}$ via multiplication tables in §2.2.

```
scalQ(r,x) = r*x[1]*E + r*x[2]*I + r*x[3]*J + r*x[4]*K     -- (5)

x = (1,2,3,4)      -- example quaternion as vector in R^4
y = 5E+6I+7J+8K    -- example quaternion in basis {E,I,J,K} representation

addQ1(x,y)
addQ(x,y)
x+y                                 -- (6)

scalQ(2,x)
2*x                                 -- (7)
```

EIGENMATH output:

```
x = (x0,x1,x2,x3)
xQ = (a,b,c,d)
(6,8,10,12)
(6,8,10,12)
(6,8,10,12)
(2,4,6,8)
(2,4,6,8)
```

▷ *Click here to invoke this script.*

Comment. The abbreviation in (1) marks the connection to the usual notion for quaternions. Therefore it is allowed to note a quaternion in two ways, see (2) and (3). In (3) and (4) we formulate the operation of the addition of quaternions, which is only given to demonstrate the operation as purely 'quaternionic'. In fact, the addition is inherited from the addition '+' of vector space $\mathbb{R}^4$, see (6). The same works for the scalar multiplication of quaternions, see (7).

3.1.2 ℍ as a 4D algebra with special multplication $(\mathbb{R}^4, +, \cdot, \circledcirc)$

We now implement the algebra multiplication[20] $\circledcirc$ of quaternions in EIGENMATH. The explicit formula given here follows directly from the multiplication table in [18] in the same way as e.g. the hyperbolic multiplication $\boxdot$ in §2.2.

```
#  QUATERNION algebra multiplication
multQ(x,y)= (x[1]*y[1]-x[2]*y[2]-x[3]*y[3]-x[4]*y[4])*E +
           (x[1]*y[2]+x[2]*y[1]+x[3]*y[4]-x[4]*y[3])*I +
           (x[1]*y[3]-x[2]*y[4]+x[3]*y[1]+x[4]*y[2])*J +
           (x[1]*y[4]+x[2]*y[3]-x[3]*y[2]+x[4]*y[1])*K

tty=0                          -- pretty print output
x = (x1,x1,x2,x3)
y = (y1,y2,y3,y4)
```

[20] sometimes called the GRASSMANN multiplication or HAMILTION product.

```
multQ(x,y)                    -- (x)

a = 1E+2I+3J+4K
b = 5E+6I+7J+8K
multQ(a,b)
```

EIGENMATH output:

$$\begin{bmatrix} x_1 y_1 - x_1 y_2 - x_2 y_3 - x_3 y_4 \\ x_1 y_1 + x_1 y_2 + x_2 y_4 - x_3 y_3 \\ x_1 y_3 - x_1 y_4 + x_2 y_1 + x_3 y_2 \\ x_1 y_3 + x_1 y_4 - x_2 y_2 + x_3 y_1 \end{bmatrix} \quad \begin{bmatrix} -60 \\ 12 \\ 30 \\ 24 \end{bmatrix}$$

▷ *Click here to invoke this script.*

Exercise 3.1. (Derivation of the explicit Hamiltion product formula for $\circledcirc$)
Look back at Ex.2.3 and verify the explicit formula `multQ()` in a similar way.

Exercise 3.2. (Algebraic properties of the quaternion multiplication $\circledcirc$)
a. Following [18], quaternion multiplication is not commutative. Verify this.
Hint: use the basis quaternions, e.g. check $E \circledcirc J$ etc.
b. Check more algebraic properties e.g. associativity.

Exercise 3.3. (How to memorize the quaternion multiplication?)
To get a memorizable mental structure into the unusual and complicated quaternion multiplication $\circledcirc$ =`multQ` do the following:

1. split up: the arbitrary quaternions $x = (x1, x1, x2, x3)$ and $y = (y1, y2, y3, y4)$ into an 1D real number line part in $\mathbb{R}$ and a 3D part in $\mathbb{R}^3$, i.e. $x = (x1, x1, x2, x3) = (a, u)$ with $a = x1$ and $u = (x2, x3, x4)$.

2. verify: $(a, u) \circledcirc (b, v) = (a \cdot b - u * v, a \cdot v + u \cdot b + u \times v)$

3. memorize: "*(first's minus last's, outer's plus . inner's plus . outer's cross)*".

Implement this rule as function `multQ1` in EIGENMATH. Don't forget to check your function on examples.

> We now indicate the implementation of further functions in this setting in a series of exercises. For actual use, we recommend to use the next realization of the quaternions as a CLIFFORD algebra and then use their build-in functions, see §3.2.

Exercise 3.4. (Conjugate - length - normalize)
Put the follwing functions in a toolbox `qBox.txt`. Run the test on more quaternions.

```
#  QUATERNION conjugate, length, magnitude
x = (x1,x1,x2,x3)        -- arbitrary test inputs
y = (y1,y2,y3,y4)
a = 1E+2I+3J+4K          -- concrete test inputs
b = 5E+6I+7J+8K

conjQ(q) = q[1]*E-q[2]*I-q[3]*J-q[4]*K      -- conjugate quaternion
conjQ(x)
conjQ(a)

magQ(q) = sqrt(q[2]^2+q[1]^2+q[3]^2+q[4]^2)  -- length of quaternion
magQ(x)
magQ(a)

normalQ(q) = q/magQ(q)
normalQ(x)
normalQ(a)

unitQ(q) = normalQ(q)      -- alias
```

▷ *Click here to invoke this script.*

Exercise 3.5. (inverse quaternion and the quotient of two quaternions)

```
#  inverse QUATERNION
x = (x1,x1,x2,x3)        -- arbitrary test input
a = 1E+2I+3J+4K          -- concrete test input

invQ(x) = (x[1]*E - x[2]*I - x[3]*J -x[4]*K) /
          (x[1]^2 + x[2]^2 + x[3]^2 + x[4]^2)

invQ(x)
invQ(a)
```

▷ *Click here to invoke this script.*
a. Put the function `invQ` in your toolbox `qBox.txt`. Run the test on more quaternions.
b. Shorten the code of `invQ` by use of `magQ`.
c. Implement a division function `quotQ` of quaternions using `invQ`.

Remark. The division of two quaternions is not done with a fractional bar, but using negative exponents. The reason for this is that the multiplication of two quaternions x and y is not commutative and one therefore must distinguish between $x \star y^{-1}$ and $y^{-1} \star x$.

Exercise 3.6. (*Project*: Rotations in $\mathbb{R}^3$ by means of quaternions)
– There is a vast literature on this topic. We will give here only a very first impression. –[21]
Quaternions can be used to represent rotations in three-dimensional space $\mathbb{R}^3$. Rotations will be carried out with the help of multiplications of quaternions. Such rotations can be represented by the three vector space variables x, y, z or as 'three degrees of freedom (i.e. rotation angles)' γ, ϕ, θ. Each individual degree of freedom stand for one individual rotation around one of the axes.
A quaternion q, which should represent a rotation R, must be normalized so that we have

$$R\colon p' = q \circledcirc p \circledcirc \bar{q}$$

The rotation with the help of such a *normalized* quaternion $q \in \mathbb{H}^1$ multiplied by a point $p \in \mathbb{R}^3$ and the *conjugated* quaternion $\bar{q}$ gives the new position p' of the point p.
No matrices are required with this type of rotation.
We have the fact:

$$R\colon p' = \begin{bmatrix} q1 \\ q2 \\ q3 \\ q4 \end{bmatrix} \circledcirc \begin{bmatrix} 0 \\ x \\ y \\ z \end{bmatrix} \circledcirc \begin{bmatrix} q1 \\ -q2 \\ -q3 \\ -q4 \end{bmatrix}$$

We translate this formula into Eigenmath's script language:

```
pRq( p, q ) = multQ(q[1]*E+q[2]*I+q[3]*J+q[4]*K ,        -- q
                    multQ(0*E+p[1]*I+p[2]*J+p[3]*K,       -- p
                      q[1]*E-q[2]*I-q[3]*J-q[4]*K))       -- conjQ(q)

a = 1E+2I+3J+4K
b = 5E+6I+7J+8K
pRq( a, b)

x = (x1,x1,x2,x3)
y = (y1,y2,y3,y4)
pRq( x, y)
```

▷ *Click here to invoke this script.*

Remark. (Axis angle representation) A quaternion q_r, which represents a rotation, is normalized and is represented in the axis angle representation as follows:

$$q_r = q_1 \cdot E + q_2 \cdot I + q_3 \cdot J + q_4 \cdot K \in S_{\mathbb{H}}^{r=1} \quad \text{i.e.} \quad \|q_r\| = 1 \tag{3.1}$$
$$q_r = cos(\alpha/2) \cdot E + x \cdot sin(\alpha/2) \cdot I + y \cdot sin(\alpha/2) \cdot J + z \cdot sin(\alpha/2) \cdot K \tag{3.2}$$

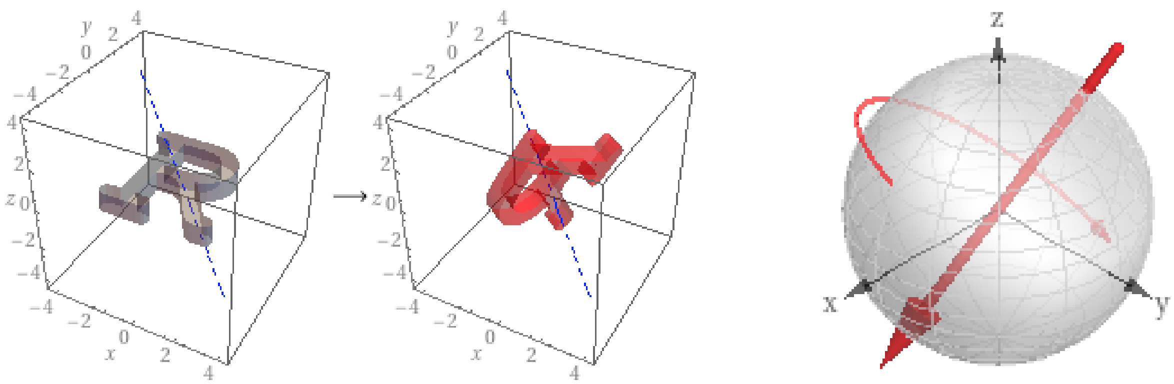
with

[21] The following short exposition is based e.g. on `https://mathepedia.de/Quaternionen.html`

- α is the angle of rotation
- $(x, y, z) \in \mathbb{R}^3$ is a normalized vector that represents the axis of rotation, e.g.
  - $X = (1, 0, 0)$ represents a rotation R_x around the x–axis and
  - $Y = (0, 1, 0)$ a rotation R_y around the y–axis.

```
#  EIGENMATH       - very preliminary implementation; to be enhanced
run("qBox.txt")
Raxis(q) = (unitQ(q)[2],unitQ(q)[3],unitQ(q)[4])
                                  -- axis  of corresponding 3D rotation
Rangle(q) = 2*arccos(q[1])        -- angle of corresponding 3D rotation

q = 0E+2I-J-3K
Raxis(q)
Rangle(q)
```

▷ *Click here to invoke this script.*
Visualisation of the above result by [19]:



↑ 3D transformation corresponding 3D rotation ↑

a. Put the function `pRq` in your toolbox `qBox.txt`. Run the test using the toolbox.
b. Verify: The quaternion I represents a rotation of 180° around the X-axis, J a rotation of 180° around the Y-axis and K a rotation of 180° around the Z-axis.
c. Verify: $I \circledcirc I = J \circledcirc J = K \circledcirc K = -1$ gives of a rotation of 360° around the axis.
b. Do a quality plot of the geometric situation using CalcPlot3D.

Exercise 3.7. (Wolfram|Alpha: quaternions.)
Verify the examples of [19] by means of the functions of our quaternion toolBox `qBox.txt`. Check norm, unit quaternion, conjugate, inverse, 3D rotation angle etc. of these examples.
▷ *Click here to invoke this script.*

Exercise 3.8. (*Project*: Polar form of a quaternion.)
– There is a vast literature on this topic. We will give here only a very first impression. –[22]
Each quaternion q can be represented in polar form. This requires the scalar amplitude $\|q\|$, the associated angle θ and a three-dimensional direction vector U:

$$q = \|q\| \cdot (cos\theta + sin\theta \cdot U) \tag{3.3}$$

$$\text{with} \qquad \theta \stackrel{def}{=} arccos(\frac{q+\bar{q}}{2 \cdot \|q\|}) \tag{3.4}$$

$$\text{and} \qquad U \stackrel{def}{=} \frac{q-\bar{q}}{\|q-\bar{q}\|} \tag{3.5}$$

The last part of formula (3.3) has to be interpreted! Therefore we do three help steps.
1° `scalarQ`: The scalar part is simply the first coordinate of the quaternion. It could be obtained by adding the conjugate value to the quaternion i.e. in (3.4) . The scalar part is in $\mathbb{R}$ and is used to determine the angle.
2° `vectorQ`: The vector part simply collects the last but first coordinates of the quaternion. It could be obtained by subtracting the conjugated quaternion from the quaternion itself, see (3.4). The vector part is our implementation in $\mathbb{R}^4$.
3° `argQ`: The quaternion argument function returns the angle between the scalar value (i.e. the real plane) and the vector represented by the quaternion.
Therefore we interpret formula (3.3) in EIGENMATH as follows:

```
run("downloads/qBox.txt")
tty=0
q = 0E+2I-J-3K

scalarQ(q) = q[1]
scalarQ(q)

vectorQ(q) = (0,q[2],q[3],q[4])
v4 = vectorQ(q)
v4

argQ(q) = arccos( scalarQ(q)/magQ(q) )
argQ(q)

polarQ(q)= magQ(q)*(cos(argQ(q))*E + sin(argQ(q))*vectorQ(q))
polarQ(q)

polar1Q(q)= do( theta = arccos( q[1]/magQ(q)),
                cth   = float(cos(theta)),
                sth   = float(sin(theta)),
                magQ(q)*(cth*q[1], sth*q[2], sth*q[3], sth*q[4]))
polar1Q(q)
```

0

$$v_4 = \begin{bmatrix} 0 \\ 2 \\ -1 \\ -3 \end{bmatrix}$$

$\frac{1}{2}\pi$

$$\begin{bmatrix} 0 \\ 2\,2^{1/2}\,7^{1/2} \\ -2^{1/2}\,7^{1/2} \\ -3\,2^{1/2}\,7^{1/2} \end{bmatrix}$$

$$\begin{bmatrix} 0 \\ 7.48331 \\ -3.74166 \\ -11.225 \end{bmatrix}$$

▷ *Click here to run the script.*

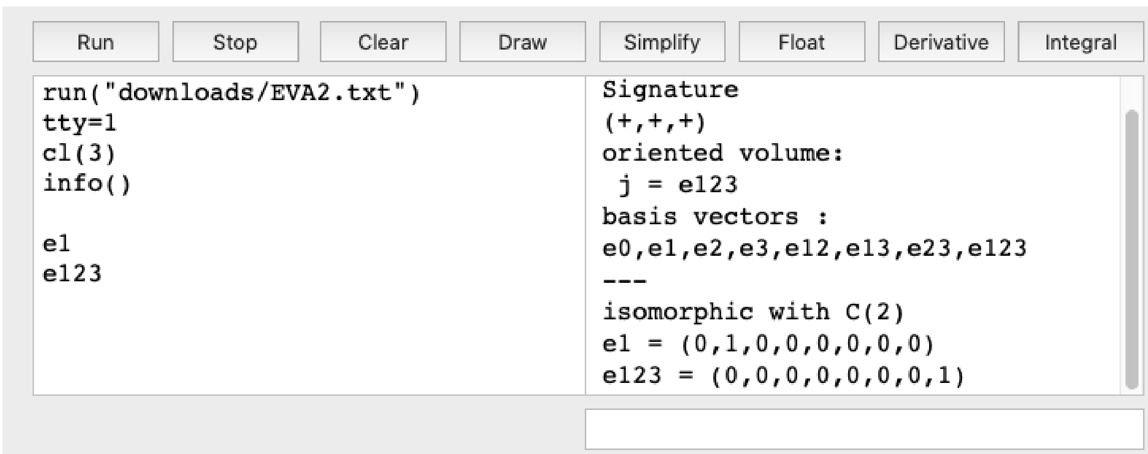
[22]The following short exposition is based e.g. on `https://mathepedia.de/Quaternionen.html`

3.2 $c\ell(3)^+$ – the CLIFFORD algebra realization of ℍ

In the previous section we worked with the quaternions only be means of bulid-in functions of EIGENMATH and a small collection of user defined functions to implement quaternion specific operations in the vector space $\mathbb{R}^4$. In this section we construct the quaternions ℍ using the same universal construction, which we used for the algebra ℂ of the complex numbers and for the hyperbolic numbers $\mathbb{H}$: an appropriate CLIFFORD algebra.

3.2.1 A look at the 4D-CLIFFORD algebra $c\ell(3)^+$

Let's look at $c\ell(3)$ and let us ask for some info about that algebra:



Comment. The call $cl(3)$ in code line 3 of the constructor function $cl(..)$ of the `EVA2` package gives the output $(+,+,+)$. This means roughly, that the norm of "the vectors in $cl(3)$"[23] has the term $\sqrt{r^2+x^2+y^2+z^2}$ with 3 plus signs. The 8 basis vectors are listed as

$$span_{\mathbb{R}}\{e0, e1, e2, e3, e12, e13, e23, e123\}$$

Obviously $c\ell(3) \sim \mathbb{R}^8$, because e.g. $e1$ and $e123$ have the expected 8 canonical coordinates in the 8D vector space $\mathbb{R}^8$.[24]
Because we need a 4D vector space to represent the 4D algebra ℍ of the quaternions, we chose 4 special basic vectors out of the 8: we take $span_{\mathbb{R}}\{e0, e12, e13, e23\}$. With $e1$ we embed the real number line $\mathbb{R}$ and his multiples into $\mathbb{H} < c\ell(3)$. The other tree vectors $\{e12, e13, e23\}$with *two* lower indices will produce the vector part of the quaternions.
We say: the quaternions are an *even subalgebra* of $c\ell(3)$, noted $c\ell(3)^+$ or sometimes $\mathbb{G}_3^+$ for a *Geometric Algebra* (GA). In summa:

$$\mathbb{H} \equiv c\ell(3)^+ \equiv \mathbb{G}_3^+ = (span_{\mathbb{R}}\{e0, e12, e13, e23\}, +, \cdot, \circledcirc)$$

▷ *Click here to invoke cl(3).*

[23]this is explained more explicit in the section on Geometric Algebra.
[24]Vector $j = e123$ would play the role of an imaginary unit, but we do not need this here.

3.2.2 Doing algebra in the 4D-Clifford subalgebra $\mathbb{H} = c\ell(3)^+$

Let's become trusted at $c\ell(3)^+$ as an realization of the quaternion algebra. The functions *magnitude*, *normalize*, *inp*, and *gp* (geometric product) are available by means of the package EVA2 and we will see, that they work as expected. Therefore we do not have to learn new symbols or new notations. To make available the usual notation $q = a+bi+cj+dk$ for a quaternion, we rename the basis vectors $\{e0, e12, e13, e23\}$ to $\{E, I, K, J\}$ and get the setting $\mathbb{H} := (span_{\mathbb{R}}\{E, I, K, J\}, +, \cdot, \circledcirc)$ inside $c\ell(3)$.

Math $\mathbb{H}$	Eigenmath EVA2 $c\ell(3)$
$A \circledcirc B$	`gp(A,B)`

We can use the same EVA2–functions as usual for the quaternions:

	Math	Eigenmath EVA2
quaternion product	A⊚B	`gp(A,B)`
inner/scalar product	$A \bullet B$	`inp(A,B)`
outer product	$A \wedge B$	`outp(A,B)`
quaternion conjugation	$\bar{B}$	`cj(B)`
inverse quaternion	$1/B$	`inverse(B)`
magnitude	$\|B\|$	`magnitude(B)`
normalize	$\frac{B}{\|B\|}$	`normalize(B)`

• The Clifford algebra functions of the EVA2 package are usable also for the quaternions. They are noted with an ending `1` to distinct them from the Eigenmath build–in functions for the complex domain, so the complex numbers $\mathbb{C}$ are also usable at the same time (e.g. to use complex quaternions): `imag1, real1, polar1, rect1, exp1, log1, sqrt1, power1, sin1, cos1, tan1, sinh1, cosh1, tanh1, asin1, acos1, atan1, asinh1, acosh1, atanh1` ...

Here is the setting to calculate with quaternions in Eigenmath's package EVA2:

```
#  QUATERNIONS in EIGENMATH
run("downloads/EVA2.txt")
tty=1
cl(3)
do(E = e0, I = e12, J = e23, K = e13 )

a = 2E + 4I - 3J + K  -- input    a = 2    +4i-3j+1k
                      --               |     |   X twist!
a                     -- goes into: (2,..,4, 1,-3,.)
b = 5E - 2I + J - 3K
b
a+b                   -- read off: 7E+2I-2K-2J
                      --              | |    X  twist!
                      -- i.e.        7+2i-2j-2k
```

```
(+,+,+)
a = (2,0,0,0,4,1,-3,0)
b = (5,0,0,0,-2,-3,1,0)
(7,0,0,0,2,-2,-2,0)
```

▷ *Click here to invoke this script.*

Remark.

- We use the alias names E, I, K, J in uppercase for the basis vectors instead of the usual $1, i, j, k$. Therefore we can also use the complex numbers noted $a + bi$ (reserved symbol i) to compute with complex quaternions.

Beware: this convention is distinct from the ordering $EIJK$ in the last section.

- We do not use the symbol e for the HAMILTONian unit, because e is a reserved symbol for $exp(1)$.
- *There is no typo in the correspondence* $\{e0, e12, e13, e23\} \mapsto \{E, I, K, J\}$: it must be $K = e13$ and $J = e23$, because of the non-commutativity of the HAMILTONian multiplication. The 8 slots (coordinates) for a quaternion using `EVA2` are therefore filled as follows, demonstrated for the quaternion a:

basis $c\ell(2)$:	e0	e1	e2	e3	e12	e13	e23	e123
basis $\mathbb{H}$:	E	-	-	-	I	K	J	-
a =	2	-	-	-	4	1	-3	-
b =	5	-	-	-	-2	-3	1	
a+b =	7	-	-	-	2	-2	-2	

All calculations with quaternions are played only at the positions $1-5-6-7$. You can watch it in the 8-tupel of the representation. To read off the correct coefficients with your eyes, you only have to remember the correct 'non-alphabetical' ordering, e.g.

basis $\mathbb{H}$:	E	-	-	-	I	K	J	-
a+b =	7	-	-	-	2	-2	-2	
	$\downarrow$				$\downarrow$	$\searrow$	$\swarrow$	
a+b =	7e				+2i	-2j	-2k	

Therefore we have: $a + b = 7E + 2I - 2K - 2J \equiv 7 + 2i - 2j - 2k$ in usual notation.[25]
Remember: the input is "twisted" saved, so read off the results also "twisted".

Exercise 3.9. (HAMILTONian rules) Verify the relations

$$i^2 = j^2 = k^2 = ijk = -1, ij = -ji = k$$

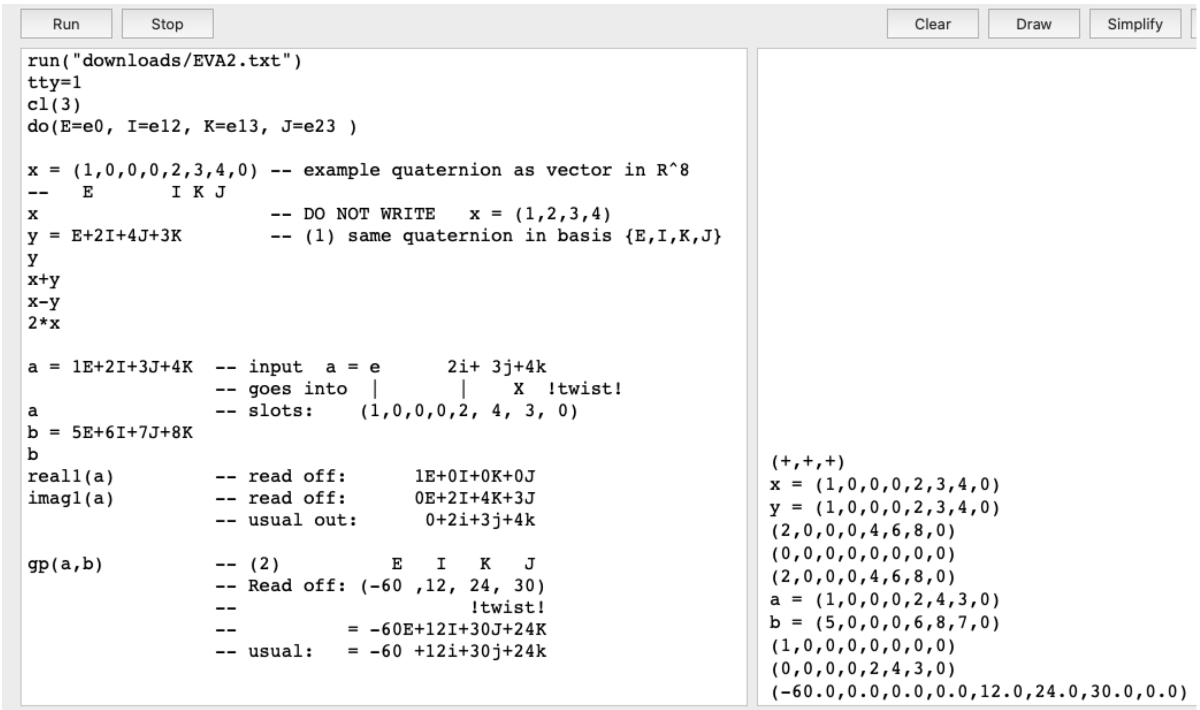
Start, but be careful: the HAMILTONian multiplication $\circledcirc$ is noted gp!

```
run("EVA2.txt")

cl(3)
do(E = e0, I = e12, K = e13, J = e23 )
gp(I,I)                  -- i^2=-1
gp(I,gp(J,K))            -- ijk=-1
gp(I,J) == - gp(J,I)     -- 1 = OK
```

▷ *Click here to invoke this script.*

[25] If you don't like this en-twisting and de-twisting you may use the ordering *EIJK*, but then the following mathematical check of the HAMILTON multiplication rules in Ex.3.9 is disturbed. But see *P.13*.

Exercise 3.10. (elementary operations with quaternions)
We continue our experiments with addition, quaternion multiplication, real and ('imaginary' alias) vector part of a quaternion.



```
run("downloads/EVA2.txt")
tty=1
cl(3)
do(E=e0, I=e12, K=e13, J=e23 )

x = (1,0,0,0,2,3,4,0) -- example quaternion as vector in R^8
--   E       I K J
x                     -- DO NOT WRITE   x = (1,2,3,4)
y = E+2I+4J+3K        -- (1) same quaternion in basis {E,I,K,J}
y
x+y
x-y
2*x

a = 1E+2I+3J+4K  -- input  a = e      2i+ 3j+4k
                 -- goes into  |        |    X  !twist!
a                -- slots:    (1,0,0,0,2, 4, 3, 0)
b = 5E+6I+7J+8K
b
real1(a)         -- read off:      1E+0I+0K+0J
imag1(a)         -- read off:      0E+2I+4K+3J
                 -- usual out:      0+2i+3j+4k

gp(a,b)          -- (2)          E   I   K   J
                 -- Read off: (-60 ,12, 24, 30)
                 --                        !twist!
                 --           = -60E+12I+30J+24K
                 -- usual:    = -60 +12i+30j+24k
```

```
(+,+,+)
x = (1,0,0,0,2,3,4,0)
y = (1,0,0,0,2,3,4,0)
(2,0,0,0,4,6,8,0)
(0,0,0,0,0,0,0,0)
(2,0,0,0,4,6,8,0)
a = (1,0,0,0,2,4,3,0)
b = (5,0,0,0,6,8,7,0)
(1,0,0,0,0,0,0,0)
(0,0,0,0,2,4,3,0)
(-60.0,0.0,0.0,0.0,12.0,24.0,30.0,0.0)
```

Comment. We show again how to read off with your eyes the result of the HAMILTONian multiplication `gp(a,b)` of the quaternions a and b:

basis $c\ell(2)$:	e0	e1	e2	e3	e12	e13	e23	e123
basis ℍ :	E	-	-	-	I	K	J	-
a =	1	-	-	-	2	4	3	-
b =	5	-	-	-	6	8	7	
$a \circledcirc b =$	-60	-	-	-	12	24	30	

.. giving the result $a \circledcirc b = -60E + 12I + 24K + 30J \overset{reorder}{=} -60 + 12i + 30j + 24k$.
▷ *Click here to invoke this script.*

Exercise 3.11. (WOLFRAM|ALPA example of quaternion multiplication)
On the internet page `https://www.wolframalpha.com/examples/mathematics/algebra/quaternions/` you find the reference example

"`quaternion -Sin[Pi]+3i+4j+3k multiplied by -1j+3.9i+4-3k`".

Reproduce it in EIGENMATH and check the result.
▷ *Click here to invoke this script.*

Exercise 3.12. (magnitude, normalization, inverse, quotient in ℍ)
We continue to calculate the length (magnitude, norm) of a quaternion, normalize a quaternion, forming their inverse resp. ⊚ and do the quotient of two quaternions.

```
run("EVA2.txt")
tty=0
cl(3)
do(E=e0, I=e12, J=e23, K=e13 )

x = x1*E+x2*I+x3*J+x4*K     -- arbitrary quaternion
a = 1E+2I+3J+4K             -- concrete quaternion

magnitude(x)                -- here we see (+,+,+)!
magnitude(a)

tty=1
normalize(x)
normalize(a)

inverse(x)
inverse(a)          -- result: 1/a = 1/30-1/15i-1/10j-2/15k

quot1(q,n) = gp(q, inverse(n)) -- division of quaternions

a = 1E+2I+3J+4K
b = 5E+6I+7J+8K
quot1(a,b)         -- read off result: ca. 0.402+0.046i-0.000j+0.091k
```

EIGENMATH output:

$$\left[x_1^2 + 1\,x_2^2 + 1\,x_3^2 + 1\,x_4^2\right]^{1/2}$$

5.47723

```
(x1 / ((x1^2.0 + 1.0 x2^2.0 + 1.0 x3^2.0 + 1.0 x4^2.0)^
(0.182574,0.0,0.0,0.0,0.365148,0.730297,0.547723,0.0)
(x1^7.0 / (x1^8.0 + 4.0 x1^6.0 x2^2.0 + 4.0 x1^6.0 x3^2
(0.0333333,0.0,0.0,0.0,-0.0666667,-0.133333,-0.1,0.0)
(0.402299,0.0,0.0,0.0,0.045977,0.091954,-(3.0 10^(-7)),
1/30.
0.0333333
1/15.
0.0666667
2/15.
0.133333
1/10.
0.1
```

Comment. The call `magnitude(x)` in line (1) shows, how the `cl(3)` info $(+,+,+)$ has to be interpreted: the norm of ℍ has a term with 3 plus signs between the 4 squares.
▷ *Click here to invoke this script.*

In section §3.2 we have verified that we can do the arithmetic and algebra of the quaternion numbers by means of a CLIFFORD algebra, in this case using a 4D subalgebra $\mathbb{H}$ of the 8D algebra $c\ell(3)$. Therefore you can forget about the construction of $\mathbb{H}$ by the multiplication table in §3.1 and use this universal construction to have the same means at hand which are usable also in other mathematical contexts.

3.2.3 Problems.

P14. Choosing the basis order $EIJK$.
The use of the basis elements $\{e0, e12, e13, e23\}$, abbreviated as $\{E, I, K, J\}$, fulfilled the HAMILTON rules in Ex.3.9 – but had the uncomfortable effect of saving the results of linear combinations in two twisted coordinate slots. If we nevertheless use the ordering $\{E, I, J, K\}$ we may avoid this and write and read the coordinates in an untwisted way, bearing in mind that *the* HAMILTON *rules had to be reflected otherwise and could not be verified in this setting.* Therefore using $\mathbb{H} := (span_{\mathbb{R}}\{E, I, J, K\}, +, \cdot, \circledcirc)$ inside $c\ell(3)$ we get a more comfortable 'usual' basis ordering. For this we have to use the preamble

```
do(E=e0, I=e12, J=e13, K=e23)
```

Example.

```
run("downloads/EVA2.txt")
tty=1
cl(3)
do(E=e0, I=e12, J=e13, K=e23 )  -- choose basis EIJK

a = 1E + 2I + 3J + 4K
a
cj(a)
```

```
(+,+,+)
a = (1,0,0,0,2,3,4,0)
(1,0,0,0,-2,-3,-4,0)
```

Now:

basis $c\ell(2)$:	e0	e1	e2	e3	e12	e13	e23	e123
basis $\mathbb{H}$:	E	-	-	-	I	J	K	-
a =	1	-	-	-	2	3	4	-
	$\downarrow$				$\downarrow$	$\downarrow$	$\downarrow$	
cj(a) =	1	-	-	-	-2	-3	-4	

The critical HAMILTON multiplication works also.We then have the correspondence

$$\begin{array}{rcl} \text{EVA2} & | & \text{Math} \\ aE + bI + cJ + dK & = & a + bi + cj + dk \\ 1E + 2I + 3J + 4K & = & 1 + 2i + 3j + 4k \end{array}$$

◦ We use and demonstrate this in the solution of the next problem.

P15. Checking the MatLAB Aerospace demo.
Here you find the MatLAB Aerospace Toolbox:
`https://de.mathworks.com/help/aerotbx/ug/quatmultiply.html`
Do all the examples with Eigenmath `EVA2` toolbox. Don't miss their examples (at the bottom) for `quatconj | quatdivide | quatinv | quatmod | quatmultiply | quatnormalize`

▷ *Click here to invoke this script.*

P16. Wolfram|*alpha* **Quaternions examples.**
Here you find the Wolfram|*alpha* Quaternions examples:
`https://www.wolframalpha.com/examples/mathematics/algebra/quaternions/`
Do all the examples with Eigenmath `EVA2` toolbox.

P17. Equivalence of the two constructions of $\mathbb{H}$.
Redo the exercises Ex.3.5 to Ex.3.8 using the Eigenmath `EVA2` toolbox i.e. using the Clifford algebra realisation of $\mathbb{H}$.

$\circledcirc$

Summary.

We *first* have constructed the well-known algebra $\mathbb{H}$ of the quaternion numbers as a 4D vector space extended by a special multiplication $\circledcirc$. This way we realized also the desired Hamilton rules.

*Second*ly, we gave also a realization of the quaternions by invoking the Clifford algebra $c\ell(3)$ of the Eigenmath package `EVA2` and choosing a 4D subalgebra. This package provides all important $\mathbb{H}$–typical functions like quaternion multiplication (`gp`), quaternion conjugate, quaternion 'imaginary' part (figuratively, i.e. the vector part of the last 3 components), quaternion reciprocal, quaternion norm and allows to enhance with user-defined functions like quaternion division etc.

Meanwhile the user should have gained a working knowledge of the quaternions $\mathbb{H}$ and the use of the package `EVA2`. We now turn to the generalization of all the our lower dimensional example constructions like the complex numbers $\mathbb{C}$, the hyperbolic numbers IH and the quaternion numbers $\mathbb{H}$ and turn to the topic of the famous Geometric Algebra (GA).

4 $\mathbb{G}$ – the 3D and 2D Geometric Algebra

We now reconstruct the well-known vector space $\mathbb{R}^3$ as a CLIFFORD algebra. This way we have all important $c\ell$–typical functions at our disposal e.g. the CLIFFORD algebra multiplication `gp`, the *geometric product*, we had used so often in the chapters before. Let us see, if we get a surplus to the usual view at $\mathbb{R}^3$!
We start with the 3D vector space $\mathbb{R}^3$, because we can describe the '*graded*' algebra construction more clear. Then we turn to the 2D vector space $\mathbb{R}^2$ to do some elementary linear algebra from this new viewpoint.
In both cases we use EIGENMATH's package `EVA2` as our working engine. Our presentation is especially inspired by the books of MACDONALD [8], SOBCZYK [13], the presentation of EYHERAMENDY in [5] and the student guide of LOUNESTO's `CLICAL` computer program, see [7].

4.1 $\mathbb{R}^3$ as Geometric Algebra $\mathbb{G}^3$

To start let's take a curious and innocent look at the implemented CLIFFORD algebra $c\ell(3)$ and call the `info()` command in `EVA` package. This gives back some information about the *Signature* (three plus sign), the *oriented volume* j and the names $e0, e1, e2, e3, e12, e13, e23, e123$ of the 8 *basis vector*s of this 8D vector space.

4.1.1 Some a priori `info()` about the CLIFFORD algebra $c\ell(3)$

Run Stop Clear Draw Simplify Float

```
run("downloads/EVA2.txt")
tty=1
cl(3)
info()

M = 0*e0+1*e1+1*e2+1*e3+2*e12+2*e13+2*e23+3*e123
M
                                  -- DIMENSION  n
M = 0*e0  +                       -- scalars    0
    1*e1  + 1*e2  + 1*e3 +        -- vectors    1
    2*e12 + 2*e13 + 2*e23 +       -- bivectors  2
    3*e123                        -- trivectors 3

dispgrd(M)
```

```
(+,+,+)
Signature
(+,+,+)
oriented volume:
 j = e123
basis vectors :
e0,e1,e2,e3,e12,e13,e23,e123
---
isomorphic with C(2)
M = (0,1,1,1,2,2,2,3)

0.0
1.0 e1 + 1.0 e2 + 1.0 e3
2.0 e12 + 2.0 e13 + 2.0 e23
3.0 e123
```

▷ *Click here to invoke this script.* – We comment on the output.

- The *oriented volume* j plays the same role as the imaginary unit i in $\mathbb{C}$:

```
j                  j = (0,0,0,0,0,0,0,1)
gp(j,j)            (-1.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0)
```

because $j^2 = gp(j,j) \equiv -1$. Why the name "or.volume"? Wait a moment. The result of "j^2" is placed in slot 1, i.e. the slot of the embedded real numbers $\mathbb{R} \subset c\ell(3)$. Therefore j is also called the *pseudoscalar*.

• As an example input we define $M = (0,1,1,1,2,2,2,3) \in c\ell(3)$ as a "full" element of $c\ell(3)$, which has components in every dimension. This is done in the form

```
M=0*e0+1*e1+1*e2+1*e3+2*e12+2*e13+2*e23+3*e123
```

Why are the basis vectors not called `e0,e1,e2,e3,e4,e5,e6,e7`? Isn't it simpler?
That would be possible, but it would hide the implicit structure of the Clifford number!
Therefore we repeat the input of M, but this time structured and sorted and spread over 4 lines of input.

- ◦ The `e0` line collects the real number parts of `M`.
- ◦ The `e1,e2,e3` line collects the 1D vector parts of `M`.
- ◦ The `e12,e12,e23` line collects the 2D number parts of `M`.
- ◦ The `e123` line collects the 3D number parts of `M`.

The `EVA` command `dispgrd` (short for 'display grade') gives an unstructured input back in a 'graded sorted structured form'.

4.1.2 A concept image of the objects in $c\ell(3)$

We elaborate a bit on the *graded output* and try to give more feeling and insight to it.

basis $c\ell(3)$:	e0	e1	e2	e3	e12	e13	e23	e123	Think of ...
M =	0	1	1	1	2	2	2	3	
point	0								•
vector		1	1	1					$\nearrow\nearrow\nearrow$
bivector					2	2	2		$\circlearrowleft\circlearrowleft\circlearrowleft$
trivector								3	

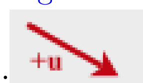
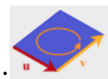
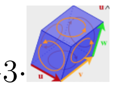
basis $c\ell(3)$:	e0	e1	e2	e3	e12	e13	e23	e123	object typ
M =	0	-1	1	-1	2	-2	2	-3	
point	0								•
vector		-1	1	-1					$\swarrow\ \nearrow\ \swarrow$
bivector					2	-2	2		$\circlearrowleft\ \circlearrowright\ \circlearrowleft$
trivector								-3	

The coefficients $0, 1, 2, 3$ are chosen to remember at the dimension.
The $\pm$ sign chose the *orientation* of every component.
$\nearrow$ means an oriented line segment.
$\circlearrowleft$ means an oriented plane segment
means an oriented space element.
This points to the "*Ausdehnungslehre*" (extension theory) of Grassmann, because the involved objects have growing dimensionality ($\overset{DE}{\sim}$"Ausdehnung").

Example. The input '*multi'vector* `M=0*e0+1*e1-2*e12+3*e123` $\in c\ell(3)$ could mentally be thought of

$$M = 0 \bullet +1 \nearrow -2 \circlearrowleft +3 \bowtie$$

or viewed as concept by invoking mental images like

M =	$0 \cdot e0$	$+1 \cdot e1$	$-2 \cdot e12$	$+3 \cdot e123$
=	$0 \cdot \bullet$	$+1 \cdot \nearrow$	$-2 \cdot \circlearrowleft$	$+3 \cdot \bowtie$
	scalar	vector	bivector	trivector
=	0s	+1v	-2B	+3T
	or.point	or.lenght	or.area	or.solid
	$0 \cdot \bullet$	$+1\cdot$ 	$-2\cdot$ 	$+3\cdot$ 

In words: the object M consists of no points, but has one positiv oriented line segment, two opposite oriented plane segments and tree positive oriented volume segments.[26]
Seems strange? Listen to MACDONALD [8, p.81]:

> " *How can we add, e.g a scalar and a vector? Are we not adding apples and oranges? Yes, but there is a sense in which we can add apples and oranges: put them together in a bag, which is analogous to M. The apples and the oranges retain their separate identities, but there are "apples + oranges" in the bag.*

In this sense we have the

> *Definition.* The *Geometric Algebra* $\mathbb{G}^3$ is the vector space $c\ell(3)$ with the additional operation gp, called the *geometric product*.

Remarks.

1. The Geometric Algebra is indeed a vector space. A proof is in [8, p.81].
2. The members of the Geometric Algebra $\mathbb{G}^3$ in $\mathbb{R}^3$ are called *multivectors* (MACDONALD) or *g-numbers* (SOBCZYK) or CLIFFORD *numbers* (if you think at $c\ell(3)$).
3. The geometric interpretation of the different elements of $\mathbb{G}^3$ for the dimensions $n = 0$ (signed point), 1 (oriented length), 2 (oriented area), 3 (oriented solid) make up its '*grades*'.
4. Thought in concepts of programming languages, a *vector* in $\mathbb{R}^8$ is an *array* of objects (real numbers) of the same kind, whereas an Geometric number in $\mathbb{G}^3$ is similar to an *record* of objects of different kinds.
5. In a nutshell: The "k-vectors" of grade k are sums of products of k vectors. When elements of different grades are multiplied, the grades add like multiplication of polynomials. It is in this sense that the Geometric Algebra is a *graded algebra*.

[26] Figures cut from `https://en.wikipedia.org/wiki/File:N_vector_positive.svg`

We do not explore or use these technical aspects in this introduction.

Example. We have predicates to decide, whether a member of $\mathbb{G}^3$ is a scalar, a vector, a bivector or trivector (or in general: a *multivector*).

```
run("EVA2.txt")

cl(3)

s = 3*e0
isScalar(s)              -- output: 1 = yes

v = 2*e1+3*e2+4*e3
isVector(v)              -- output: 1 = yes

B = 4*e23
isMvector(B)             -- output: 1 = yes

T = 3*e123
isMvector(T)             -- output: 1 = yes

M = s+v+B+T
isMvector(M)             -- output: 1 = yes
```

Exercise 4.1. Maybe you miss special predicates to check, if a g-number is a pure *bivector* or a pure *trivector*. Here is the code for `isVector`:

```
isVector(u) = test(u==grade(u,1), 1, u=0, 1)
```

Write analogous checks for `isBivector` and `isTrivector`.
Test your code on the g -numbers s, v, B, T.
▷ *Click here to invoke this script.*

Exercise 4.2. Sometimes you wish the output of EIGENMATH's `EVA` not in 8-slots coordinate form. To have the output in multivector symbolic form (but not in space consuming graded form), you may use the following helper function `disp3(.)`:

```
# display symbolic u, code from b.E.
disp3(u) = do( isMvector(u),
              print(u[1]*"e0"+u[2]*"e1"+u[3]*"e2"+u[4]*"e3"+
                    u[5]*"e12"+u[6]*"e23"+
                    u[7]*"e13"+u[8]*"e123"))
```

▷ *Click here to invoke this script.*

```
M = 3*e0+2*e1+3*e2+4*e4*e23+3*e123
M
dispgrd(M)
disp(M)
```

```
M = (3,2,3,0,0,0,4 e4,3)

3.0 e0
2.0 e1 + 3.0 e2
4.0 e23 e4
3.0 e123
3 e0 + 2 e1 + 3 e123 + 4 e13 e4 + 3 e2
```

4.1.3 Inner, outer, geometric product – `inp`, `outp`, `gp`

A. The vector space $\mathbb{R}^3$ is equipped with the standard scalar product $\bullet$, i.e. $(\mathbb{R}^3, +, \cdot, \bullet)$ is an *inner product space* aka EUCLIDean vector space. Therefore the name EVA$_{lgebra}$♡ .. and we have the geometric concepts of orthogonality, angle etc. at our disposal. For the CLIFFORD algebra $c\ell(3)$ we have an adapted version of $\bullet$=`dot(.)=inner(.)`, which is called `inp(.)` and which is per definitionem compatible with the algebra multiplication table for the 8 basis vectors.

Example.

```
# EIGENMATH
run("EVA2.txt")

cl(3)
v = 2*e1+3*e2+4*e3       -- a vector in G3
v
-- dot(v,v)              -- dot does NOT work
inp(v,v)                 -- inner product in cl(3)=G3

v8 = (0,2,3,4,0,0,0,0)   -- v as vector in R^8
dot(v8, v8)              -- here dot does work as inner product in R^8
```

▷ *Click here to invoke this script.*

B. The CLIFFORD algebra equivalent to the 3D *cross* product of $\mathbb{R}^3$ is the `outer` (alias *exterior* alias *wedge*) product of $\mathbb{G}^3 = (\mathbb{R}^8, +, \cdot, inp, outp)$.

```
# EIGENMATH
# .. preamble omitted
u = 2e1+3e2+4e3        -- (1) has 8 coordinate slots
v = 4e1+1e2+3e3
outp(u,v)              -- (2) invoke  OUTER alias WEDGE product u^v
magnitude(outp(u,v))   -- (3) output: 15

u3 = (2,3,4)           -- pendant in 3D space R^3
v3 = (4,1,3)
cross(u3,v3)           -- (3) invoke CROSS product, gives (5,10,-10)
abs(cross(u3,v3))      -- (4) output: 15 = area of parallelogram u3.v3
```

▷ *Click here to invoke this script.*

C. The CLIFFORD algebra geometric product gp has no pendant in the real vector space $\mathbb{R}^3$. It is defined as a special multiplication construct via a clever multiplication table 'Gtable' on the 8 basis vectors using `dot(.)`.[27] With it we extend the well-known inner product space $\mathbb{R}^3$ to a full blown geometric algebra

$$\mathbb{G}^3 = (\mathbb{R}^8, +, \cdot, inp, outp, gp)$$

We will give two hints as a possible motivation[28] of the geometric product. For a detailed mathematical oriented exposition see e.g. [8, pp. 93–117].
1st: We have the so-called 'fundamental identity'[29], which describes a famous connection between the tree products. In a special case for g-vectors u, v we have:

The Fundamental Identity

Math $\mathbb{G}^3$	EIGENMATH EVA2 $c\ell(3)$
$u\,v = u \bullet v + u \wedge v$	`gp(u,v)=inp(u,v)+outp(u,v)`

2nd: In Ex.3.3.2 we have see $(a, u) \circledcirc (b, v) = (.., av + ub + u \times v)$, which may shed some light on the fundamental identity.

Example.
```
# EIGENMATH
# .. preamble omitted
u = 2e1+3e2+4e3
v = 4e1+1e2+3e3

gp(u,v)
inp(u,v)+outp(u,v)
```
EIGENMATH output:
```
gp(u,v)                           (23.0,0.0,0.0,0.0,-10.0,-10.0,5.0,0.0)
inp(u,v)+outp(u,v)                (23.0,0,0,0,-10.0,-10.0,5.0,0)
gp(u,v) == inp(u,v)+outp(u,v)     1
```
▷ *Click here to invoke this script.*

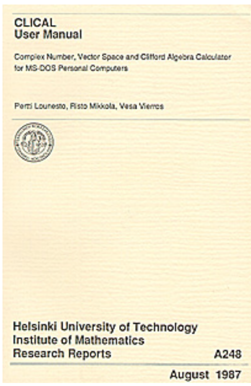
Exercise 4.3.
a. Give two g-vectors, which are orthogonal resp. *inp*. Check with *dot* and *inp*!
b. Calculate the volume of the 3D spare spanned by $A = (1, 2, 0), B = (0, 3, 4), C = (2, 0, 3)$ first using methods of $\mathbb{R}^3$ and second by interpreting A, B, C as members of $\mathbb{G}^3$.
c. Calculate the area of the triangle with edges A, B in two ways: working in $\mathbb{R}^3$ and then in $\mathbb{G}^3$.

[27] We have shown similar constructions for $\mathbb{C}$, $\mathbb{H}$, $\mathbb{H}$. Here the construction is
`gp(u,v) = do( isMvector(u), isMvector(v), dot( Gtable(u), transpose(v)))`
[28] A very convincing derivation can be found in SOBCZYK [13, pp. 24–32].
[29] see [8, p. 111]

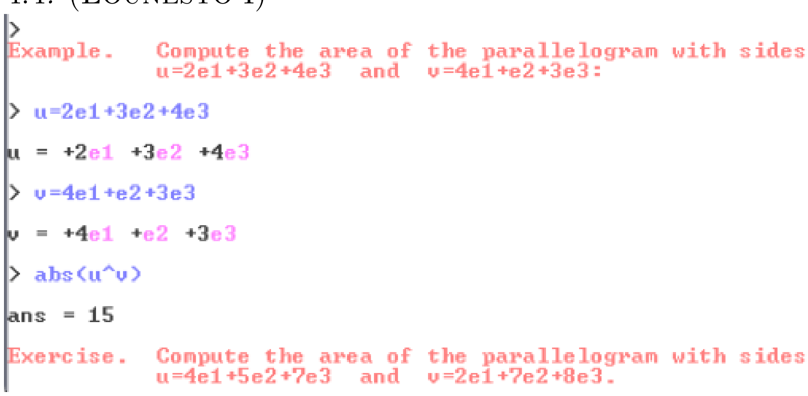
4.2 An Potpourri of applications: getting a working knowledge of $\mathbb{G}^3$

4.2.1 Doing parts of the tutorial of LOUNESTO

In 1987 Pertti LOUNESTO published an CLIFFORD algebra calculator for the MSDOS world, named CLICAL, see [7]. We follow here some steps of his tutorial using EIGENMATH's package **EVA2**. Here is an impression of CLICAL:



Exercise 4.4. (LOUNESTO I)




```
# EIGENMATH
run("EVA2.txt")

cl(3)

u = 2e1+3e2+4e3
v = 4e1+1e2+3e3
magnitude(outp(u,v))  -- wedge = outer product
```

▷ *Click here to invoke this script.*

Exercise 4.5. Do the exercise given in the screenshot.

Exercise 4.6. (Lounesto II)

```
>
Example.   Compute the component of  q=-5e1+7e2  perpendicular to the
           plane  F  spanned by  4e1+e3  and  3e1+e2:

> q=-5e1+7e2

q = -5e1 +7e2

> F=(4e1+e3)^(3e1+e2)

F = +4e12 -3e13 -e23

> (q^F)/F

ans = -e1 +3e2 +4e3

Exercise.  Compute the component of  q  parallel to the plane  F.
```

Solution. with Eigenmath. – A detailed discussion is given below in 4.3.4 for $\mathbb{G}^2$.

```
run("downloads/EVA2.txt")
tty=1
cl(3)

q=-5e1+7e2
q

F = outp(4e1+e3, 3e1+e2)         -- (1)
F
dispgrd(F)                       -- (2)

ans = gp( outp(q,F), inverse(F)) -- (3)
dispgrd(ans)                     -- (4)
```

```
(+,+,+)
q = (0,-5,7,0,0,0,0,0)
F = (0.0,0.0,0.0,0.0,4.0,-3.0,-1.0,0.0)

0.0
0
4.0 e12 - 3.0 e13 - e23
0.0

0.0
-e1 + 3.0 e2 + 4.0 e3
0
0.0
```

▷ *Click here to invoke this script.*

Exercise 4.7. (Lounesto III)

```
CliffCAL 4
>
Example.  Rotate the vector  r=2e1+e2+2e3  around the axis
          a=1.5e1+2e2  by the angle  |a|=2.5:

> a=1.5e1+2e2

a = +1.5e1 +2e2

> s=exp(j a/2)

s = 0.315
    -0.759e13 +0.569e23

> r=2e1+e2+2e3

r = +2e1 +e2 +2e3

> s r/s

ans = -0.398e1 +2.799e2 -1.004e3

Exercise.  Rotate the vector  r=3e1+4e2+2e3  around the axis
           a=4e1+e2+3e3  by the angle  2pi/7.
```

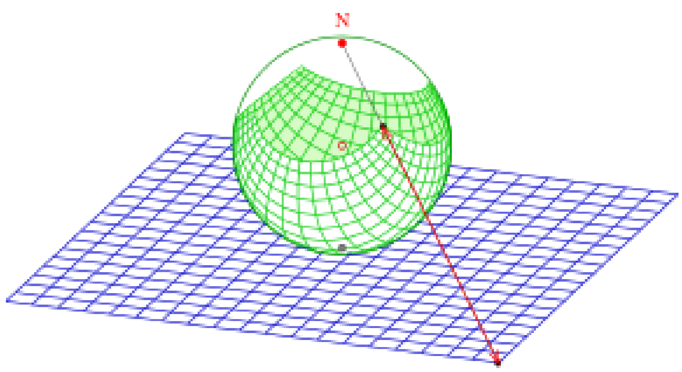
Solution. with Eigenmath

```
run("EVA2.txt")

cl(3)
```

```
a=1.5e1+2e2
a
s=exp1(gp(j,a/2))
sDownloads
dispgrd(s)

r=2e1+e2+2e3
r
quot1(a,b)= gp(a, inverse(b))      -- ad hoc definition of quotient
gp(s, quot1(r,s))
```

▷ *Click here to invoke this script.*

4.2.2 The stereographic projection



The stereographic projection[30] has the formula

$$\begin{aligned} projS\colon S^2 \subset \mathbb{R}^3 &\to \mathbb{R}^3 \\ a &\mapsto projS(a) := \frac{2}{a+e3} - e3 \end{aligned}$$

for the unit sphere centered at the origin. The following task is from [13, p.111 ff].
a. Verify that in cartesian coordinates (x, y, z) on the sphere and $(X, Y, 0)$ on the xy-plane, the projection is given by the formula

$$\left(\frac{x}{1-z}, \frac{y}{1-z}, 0\right) =: (X, Y, 0)$$

a. Verify that $a = \frac{1}{4}(\sqrt{3}, 2, 3) \in S^2$.
b. Find the corresponding point $a' \in \mathbb{R}^2$ using methods of $\mathbb{R}^3$ resp. $\mathbb{G}^3$.
c. Verify your result by a quality plot using CalcPlot3D.

[30] picture found at `https://de.m.wikipedia.org/wiki/Datei:Stereogr-proj-netz.svg`

4.2.3 Problems.

P18. Matrix representation of $\mathbb{G}^3$ via DIRAC matrices.

> The following project[31] would make no fun, if you do not use a CAS like EIGENMATH. Using square matrices to represent vectors enables us to *define a new multiplication* of vectors, which would be impossible inside $\mathbb{R}^3$.

Let $e_1, e_2, e_3 \in \mathbb{R}^{4\times 4}$ be the following matrices:

$$e_1 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}, e_2 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \\ 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix}, e_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$$

For the following tasks represent e_1, e_2, e_3 in EIGENMATH.
Then use EIGENMATH to prove the following properties.

a. Show, that every vector x in $\mathbb{R}^3$ is a linear combination of e_1, e_2, e_3, i.e. $x = x_1 \cdot e_1 + x_2 \cdot e_2 + x_3 \cdot e_3$ with $x_1, x_2, x_3 \in \mathbb{R}$.

b. Show: $e_1^2 = e_2^2 = e_3^2 = E$, E being the unity matrix `unit(4,4)`.

c. Show: $e_2 \star e_3 + e_3 \star e_2 = e_3 \star e_1 + e_1 \star e_3 = e_1 \star e_2 + e_2 \star e_1 = O$, O being the zero matrix `zero(4,4)` and $\star$ the matrix multiplication `dot(.)`.

> Therefore this set of matrices form the basis for the CLIFFORD algebra associated with the innerproduct space $(\mathbb{R}^3, +, \cdot, \bullet)$.

d. Let $y = y_1 \cdot e_1 + y_2 \cdot e_2 + y_3 \cdot e_3$ be another arbitrary vector written in the basis e_1, e_2, e_3. Define the "geometric" product $\circledcirc$ of x and y through

$$x \circledcirc y \overset{def}{=} (x_1y_1+x_2y_2+x_3y_3)E+(x_2y_3-x_3y_2)e_2\star e_3+(x_3y_1-x_1y_3)e_3\star e_1+(x_1y_2-x_2y_1)e_1\star e_2$$

and the inner product

$$x \circ y \overset{def}{=} \frac{1}{2} \cdot (x \circledcirc y + y \circledcirc x)$$

and the wedge product

$$x \wedge y \overset{def}{=} \frac{1}{2} \cdot (x \circledcirc y - y \circledcirc x)$$

- Show: $x \circledcirc y = x \circ y + x \wedge y$ (*Fundamental Identity*)
- Verify that the coefficients of the wedge product are the same coefficients like the cross product.
- Give an explicit formula for *wedge* showing the coefficients.
- Calculate $(1, 2, 3) \circledcirc (4, 5, 6)$ and $(1, 2, 3) \circ (4, 5, 6)$ and $(1, 2, 3) \wedge (4, 5, 6)$ via that definitions using EIGENMATH.

[31] This is condensed from a detailed presentation in SNYGG [11, pp. 3–6].

e. By considering all possible products of $e1, e2, e3$ one obtains an 8D vector space spanned by $\{I, e_1, e_2, e_3, e_1 \circledcirc e_2, e_2 \circledcirc e_3, e_3 \circledcirc e_1, e_1 \circledcirc e_2 \circledcirc e_3\}$.

∘ Let EIGENMATH write down all 8 basis vectors in 4×4 matrix form.

∘ Define the alias $e0 := E, e1 := e_1, e2 := e_2, e3 := e_3, e23 := e_2 \circledcirc e_3, , e31 := e_3 \circledcirc e_1, e12 := e_1 \circledcirc e_2, e123 := e_1 \circledcirc e_2 \circledcirc e_3$ for the geometric products of DIRAC vectors $e1, e2, e3$.

Verify: $(\{e0, e1, e2, e3, e12, e23, e31, e123\}, +, \cdot, \circledcirc)$ is an 8-dimensional vector space closed under $\circledcirc$, i.e. it is a realization of the CLIFFORD algebra $\mathbb{G}^3$.

Remark.
- An 0-vector (alias *scalar*) is any scalar multiple of $e0 = E$.
- An 1-vector (alias *vector*) is any linear combination of the Dirac vectors $e1, e2, e3$.
- An 2-vector (alias *bivector*) is any linear combination of vectors $e12, e23, e13$.
- An 3-vector (alias trivector alias *pseudoscalar*) is any scalar multiple of $e123$.
- An M-vector (alias *multiivector*) is any linear combination of vectors of any type, i.e. an arbitrary linear combination of the 8 basis vectors.

P19. Project: Representation of $\mathbb{G}^3$ by PAULI matrices.
Let's take another representation[32] for the three DIRAC vectors e_1, e_2, e_3.
Define the PAULI matrices $e1, e2, e3 \in \mathbb{C}^{2\times 2}$ through

$$e1 := \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, e2 := \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, e3 := \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

Redo *P.21* in this setting, i.e. show that $(\{e0, e1, e2, e3, e23, e31, e12, e123\}, +, \cdot, \circledcirc)$ is an 8-dimensional vector space closed under $\circledcirc$, i.e. it is a realization of the CLIFFORD algebra $\mathbb{G}^3$. Use EIGENMATH.

P20. Project: Representation of the quaternions $\mathbb{H}$ by PAULI matrices.
Following the setting in *P.18* realize the basis quaternions I, J, K through[33]
$I := -e23, J := -e31, K := -e12$. Let $e0 := E = \left[\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix}\right]$.
a. Verify

$$I := \begin{bmatrix} 0 & -i \\ -i & 0 \end{bmatrix}, J := \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}, K := \begin{bmatrix} -i & 0 \\ 0 & i \end{bmatrix}$$

b. Redo *P.22* in this setting, i.e. show that $(\{E, I, J, K\}, +, \cdot, \circledcirc)$ is an 4-dimensional vector space closed under $\circledcirc$, i.e. it is a realization of the CLIFFORD algebra of the quaternions $\mathbb{H}$. Use EIGENMATH.

---
[32] See SNYGG [11, p. 12, problem 2]
[33] See SNYGG [11, p. 12, problem 4] or [14, pp. 32–33]

4.3 $\mathbb{R}^2$ as Geometric Algebra $\mathbb{G}^2$

All concepts and constructions are known, so we go directly in medias res.

4.3.1 First contact with the Clifford algebra $c\ell(2)$ alias $\mathbb{G}^2$

Run Stop Clear Draw Simplify Float Derivative Integ

```
run("downloads/EVA2.txt")
tty=1
cl(2)
info()

M=1e0+2e1+3e2+4e12  -- multivector in G2=cl(2)
M
disp2(M)            -- display version for cl(2)
dispgrd(M)          -- grade display of M
tty=0
helpCL              -- HELP page for cl(n)
```

```
 j = e12
basis vectors :
e0,e1,e2,e12
---
isomorphic with R(2)
M = (1,2,3,4)
e0 + 2 e1 + 4 e12 + 3 e2

1.0 e0
2.0 e1 + 3.0 e2
4.0 e12
```

gp(u,v) :	geometic product : u v
inp(u,v) :	inner product : u.v
outp(u,v):	outer product : u^v

Invoking the `info()` we get presented: the *Signature*, which is 'two plus sign', the *oriented volume* j as abbreviation for the pseudoscalar $e12$ with the property $j^2 = -1$ and the members $e0, e1, e2, e12$ of the basis, which make up the 4D vector space $\mathbb{G}^2$. For a test we input a multivector M and display it in tree different shapes. ▷ *Click here to run the script.* Finally we invoke a small cheatsheet for Clifford algebra with the command `helpCL`. Because there is nothing new, we dive directly into some applications.

4.3.2 Determinants and the oriented volume element j

Run Stop Clear Draw Simplify Float

```
run("downloads/EVA2.txt")
tty=1
cl(2)               -- contructor for G(2)
tty=0               -- output LaTeX pretty print

A=((a11,a12),             -- (1)
   (a21,a22))
A
det(A)                    -- (2)

Det = outp( a11*e1+a12*e2,  -- (3)
            a21*e1+a22*e2)
Det                       -- (4)
Det[4]                    -- (5)
Det[4] * j                -- (6)
```

(+,+)

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

$$a_{11}\,a_{22} - a_{12}\,a_{21}$$

$$D_{et} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ a_{11}\,a_{22} - 1\,a_{12}\,a_{21} \end{bmatrix}$$

$$a_{11}\,a_{22} - 1\,a_{12}\,a_{21}$$

▷ *Click here to run the script.*

Comment. We start in (1) with a arbitrary 2×2 matrix A and calculate their determinante. We get back the well-known Leibniz formula $a_{11}a_{22}-a_{12}a_{21}$. In (3) we interpret the elements of the rows of A as coefficients of multivectors in $\mathbb{G}^2$ by defining $A1 = a11 * e1 + a12 * e2$ for the first row of matrix A and $A2 = a21 * e1 + a22 * e2$ for the second. We then call the *outer* alias wedge product and let the result showin (4) via `Det`:

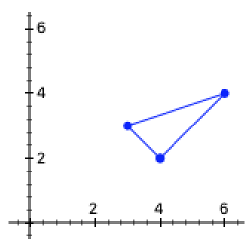
$$\begin{array}{rcl} \text{Math} & & \text{EigenMath} \\ \textit{wedge} & & \textit{outer product} \\ A1 \wedge A2 & = & \texttt{outp(A1,A2)} \\ & = & (0,0,0,det(A)) \end{array}$$

In (5) we pick off the real value $det(A)$, which resides in the 4^{th} slot, i.e. in the position of the basis vector *pseudoniverse* j. In (6) we factor out the $det(A) \in \mathbb{R}$ value of $j \in \mathbb{G}^2$ slot. Because *det* gives the area res. volume of the 2D resp. 3D parallelogram resp. spare one calls $j = 1 * j$ the *oriented (unit) volume element* of $\mathbb{G}^2$ resp. $\mathbb{G}^3$.
We memorize the fact: for arbitrary $a, b \in \mathbb{G}^2 = c\ell(2)$ and the unit bivector j

$$a \wedge b = det(a,b) \cdot j \overset{EVA}{=} \texttt{outp(a,b)}$$

Exercise 4.8.
a. Calculate the determinant of $A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$ via multivectors of $\mathbb{G}^2$.

b. Determine the area of the plane triangle $\triangle(3,3)(4,2)(6,4)$ using g-numbers.



Exercise 4.9. Verify using EIGENMATH the 3D version: for $a = a1*e1+a2*e2+a3*e3, b = b1*e1+b2*e2+b3*e3, c = c1*e1+c2*e2+c3*e3 \in \mathbb{G}^3 = c\ell(3)$ and the unit trivector $j \in \mathbb{G}^3$ we have

$$a \wedge b \wedge c = det \begin{bmatrix} a1 & a2 & a3 \\ b1 & b2 & b3 \\ c1 & c2 & c3 \end{bmatrix} \cdot j \overset{EVA}{=} \texttt{outp(a,b)}$$

Exercise 4.10. Calculate the determinant of the 4×4 matrix `B = ((1,2,5,2), (0,1,2,3), (1,0,1,0), (0,3,0,7))`$\in \mathbb{R}^{4\times 4}$ by interpreting their rows as multivectors in $\mathbb{G}^4$ and using the unit pseudoscalar alias oriented 4D volume element j.
▷*Click here to see the solution.*

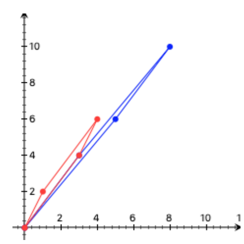
Remark.
∘ The *wedge* product ∧ (i.e. `outp`) is also called the *exterior* or GRASSMANN product in the exterior algebra $c\ell$.
∘ Calculating in the Geometric Algebra $\mathbb{G}^n$ with the outer product as operation of multiplication *one does not need a special theory of determinants anymore.* All rules and properties (e.g. orientation, multilinearity, anti-commutativity etc.) of the determinants are perfect integrated into the concept of a Geometric Algebra $c\ell(p,q)$.

4.3.3 The Geometric Algebra version of the Cramer rule

```
# EIGENMATH
A = ((1,2),(3,4))    -- (1)
B = (5,6)
det(A)
X = dot(inv(A),B)    -- (2)
X

run("EVA2.txt")
cl(2)                -- calculate in G^2

A1 = 1e1+3e2         -- (3) columnwise structure!
A2 = 2e1+4e2
B  = 5e1+6e2

Det = outp(A1,A2)    -- (4)
Det
Det[4]               -- (5)

x = outp(B,A2)[4]/outp(A1,A2)[4]  -- (6)
x
y = outp(A1,B)[4]/outp(A1,A2)[4]
y
```

Eigenmath output:

```
Det = (0.0,0.0,0.0,-2.0)
-2.0
(0.0,0.0,0.0,-2.0)
(0.0,0.0,0.0,-0.125)
x = -4.0
y = 4.5
```



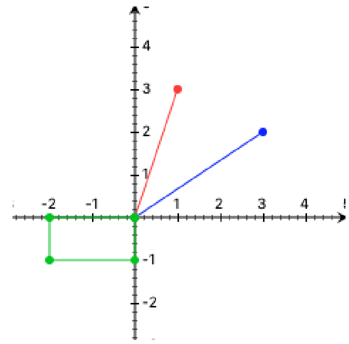
▷ *Click here to run the script..*

Comment. We are given the 2×2 linear system $\begin{bmatrix} 1x+2y=5 \\ 2x+3y=6 \end{bmatrix}$. The solution $X = \begin{bmatrix} x \\ y \end{bmatrix}$ is calculated traditionally as $X = A^{-1} \star B = \begin{bmatrix} -4 \\ 4.5 \end{bmatrix}$. This is done in (2). We alternatively invoke the Geometric Algebra $\mathbb{G}^2 = c\ell(2)$ and write the linear system in the Cramer way as 3 multivector 'column's. Then we express the solution as quotient of determinants - whereby the determinants are 'hidden' in 4th coordinate of the outer (wedge) product.
We know: The solution is geometrically interpretable as the quotient of the areas (`outp`!) of the depicted parallelograms.

4.3.4 Projections and rejections in Geometric Algebra

Vector projection is used in physics when force and work are involved. If the green box is pulled by "force" (i.e. vector) $\overrightarrow{OF}$ with $F = (3, 2)$ (blue vector), some of the force is wasted pulling up against gravity and we only use that part of the force, which is working to move the box horizontally in direction of the ground (in our model along the x–axis $e1 = (1, 0)$).

∘ Determine the portion of force F, which acts in direction of the x–axis by using high school math.



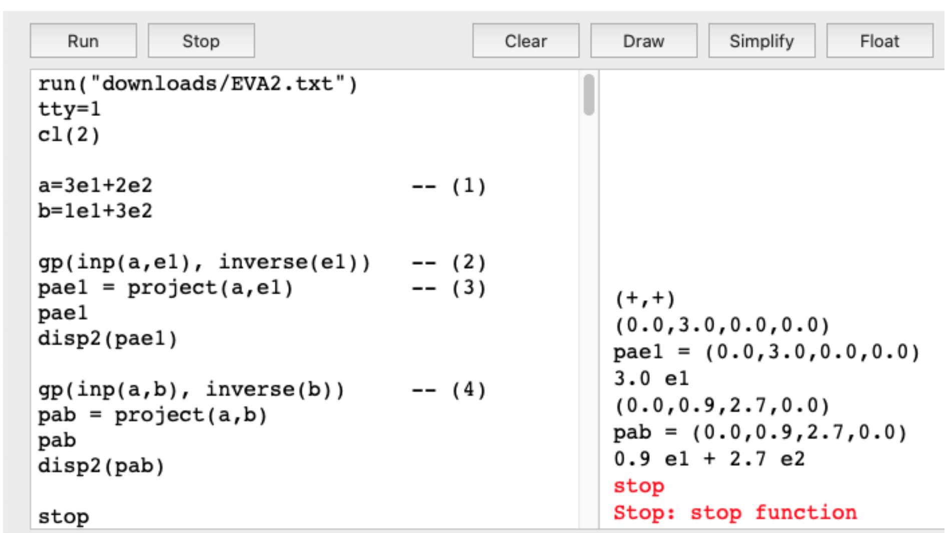
Example. Projection and rejection are important concepts of analytic geometry. We first demonstrate how to use $c\ell(2)$ concepts to calculate the *vector projection* of vector $a = (3, 2) \in \mathbb{R}^2$ onto the x-axis and onto the red vector $b = (1, 3) \in \mathbb{R}^2$. We use the $\mathbb{G}^2$ analogue to the well-known projection resp. rejection formulas[34] denoted by $a_{\|b}$ and $a_{\perp b}$:

$$\text{Math} \qquad \text{EigenMath } c\ell(2) \tag{4.1}$$

$$\mathbf{a}_{\|b} \overset{def}{=} \frac{\mathbf{a}\cdot\mathbf{b}}{\|\mathbf{b}\|}\frac{\mathbf{b}}{\|\mathbf{b}\|} = \texttt{gp(inp(a,b), inverse(b))} = \texttt{project(a,b)} \tag{4.2}$$

$$\mathbf{a}_{\perp b} \overset{def}{=} a - \frac{\mathbf{a}\cdot\mathbf{b}}{\|\mathbf{b}\|}\frac{\mathbf{b}}{\|\mathbf{b}\|} = \texttt{gp(outp(a,b), inverse(b))} = \texttt{reject(a,b)} \tag{4.3}$$

Therefore we have



```
run("downloads/EVA2.txt")
tty=1
cl(2)

a=3e1+2e2                       -- (1)
b=1e1+3e2

gp(inp(a,e1), inverse(e1))      -- (2)
pae1 = project(a,e1)            -- (3)
pae1
disp2(pae1)

gp(inp(a,b), inverse(b))        -- (4)
pab = project(a,b)
pab
disp2(pab)

stop
```

```
(+,+)
(0.0,3.0,0.0,0.0)
pae1 = (0.0,3.0,0.0,0.0)
3.0 e1
(0.0,0.9,2.7,0.0)
pab = (0.0,0.9,2.7,0.0)
0.9 e1 + 2.7 e2
stop
Stop: stop function
```

▷ *Click here to run the script..*

Comment. First (1) we put in the vectors a and b as elements of $\mathbb{G}^2$. We then use the formula (4.2) explicit and in (3) as the EVA build-in function `project`. In (4) we calculate $a_{\|b} = (3, 2)_{\|(1,3)} = (0.9, 2.7)$. Both results can be checked for plausibility in the figure.

[34] see e.g. `https://en.m.wikipedia.org/wiki/Vector_projection` or `https://www.ck12.org/book/ck-12-college-precalculus/section/9.6/`

Whereas the *projection* of a vector a onto a vector b is the component of a *parallel* to b, the *rejection* is defined as the *perpendicular* component of a resp. to b. Let's calculate the rejection with `EVA`-function `reject`:

```
# EIGENMATH
a=3e1+2e2
b=1e1+3e2

gp(outp(a,e1), inverse(e1))  -- (5)
rae1=reject(a,e1)            -- (6)
rae1
disp2(rae1)

rab=reject(a,b)              -- (7)
rab
disp2(rab)

a - project(a,e1)            -- (8) alternative formula for rejection
a - project(a,b)
```

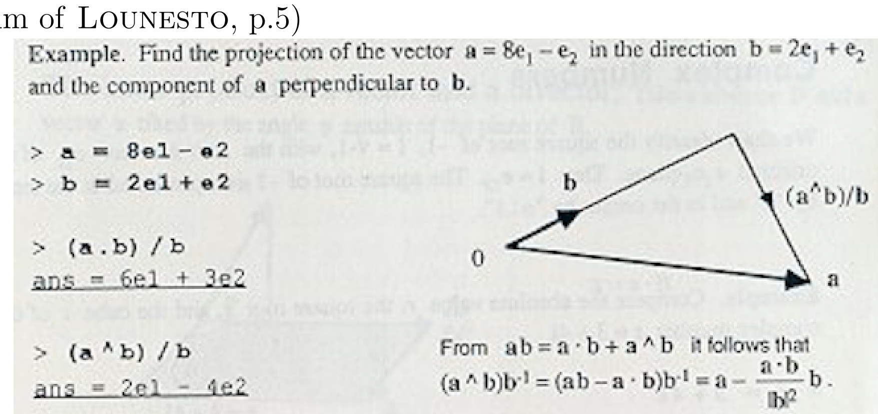
EIGENMATH output:

```
(0.0,0.0,2.0,0.0)
rae1 = (0,0,2,0)
2 e2
rab = (0,2.1,-0.7,0)
2.1 e1 - 0.7 e2
```

▷ *Click here to run the script..*

Exercise 4.11. **a.** Calculate the *scalar projection* of a onto b as length of the vector projection. Use alternatively the formula $\frac{\mathbf{a} \bullet \mathbf{b}}{\|b\|}$.
b. Calculate the scalar projection of a onto $e1$. Verify the result in the figure.
c. Determine the area of the plane triangle $\triangle(3,3)(4,2)(6,4)$ using a projection to determine its height.

Exercise 4.12. (Tutorium of LOUNESTO, p.5)



a. Redo this *CLICAL* example using EIGENMATH.

b. Under the figure in a.[35] the *Fundamental Identity*, see §4.1.3C, is used to derive the rejection formula. Explain. Elaborate on it.[36]

Exercise 4.13. (*Projection and rejection as consequence of the Fundamental Identity.*)
Looking at (4.2) and (4.3) we have for arbitrary $a \in \mathbb{G}^n, n = 2, 3$

Math		EigenMath $c\ell(2)$
		parallel and perpendicular component of a
$\mathbf{a} = a_{\|b} + \mathbf{a}_{\perp b}$	$=$	`project(a,b)+ reject(a,b)`

Show, that this decomposition is a consequence of the Fundamental Identity 4.3.1.C.
Hint: Take $b \in \mathbb{G}^n$ and normalize b to $b^\circ := \frac{b}{\|b\|}$.
Let $\circledcirc$ denote the geometric product. Then p.d. $b^\circ \circledcirc b^\circ = \|b^\circ\| = 1$.
Therefore $a = a \circledcirc 1 = a \circledcirc b^\circ \circledcirc b^\circ$. Now use the Fundamental Identity for the first two factors $a \circledcirc b^\circ$.

♡

Let's close here our short introduction to Geometric Algebra using EIGENMATH. Much more could be say about rotations, transformations, conformal geometry, spacetime geometry (MINKOWSKI space with LORENZ metric) etc. using EIGENMATH's `EVA` package. But this would be a nice topic for a another paper. Indeed, you will find some pointers and first steps on these topics in the demos of [5] and in the student guide of LOUNESTO [7, last line of the page].

4.3.5 Problems.

P21. Straigth lines and distance to a line.
Let $a = e1 + 2e2 + 3e3, b = -2e1 + 3e2 - e3, c = 2e1 + e2 - 3e3$ be multivectors in $\mathbb{G}^3$.[37]
a. Explain, that the equation of the line L through point x_0 in the direction of a is (independent of the underlying dimension)

$$L\colon (x - x_0) \wedge a = 0, \quad \text{for } x \in L$$

b. Find the equation of the line L in direction of a passing through b.
What is the distance of c to the this line L?
c. Give the equation of the plane E passing through a in "direction" of the bivector $a \wedge b$.
What is the distance of c to this plane E?

Here are some suggestions for further study.

[35] See e.g. `https://users.aalto.fi/~ppuska/mirror/Lounesto/kuvat/Pp4-5.jpg`
[36] See e.g. [13, pp. 38–39]
[37] This exercise is from [13, p. 43]

P22. Student guide of Lounesto**: plane geometry.**
Read the text and do the examples of the student guide of Lounesto [7, p. 2, 4–5] using Eigenmath's `EVA`.

P23. Student guide of Lounesto**: space geometry.**
Read the text and do the examples of the student guide of Lounesto [7, p. 2, 7–9] using Eigenmath's `EVA`.

P24. Student guide of Lounesto**: Geometric Algebra.**
Read the text and do the examples of the student guide of Lounesto [7, p. 20–26] using Eigenmath's `EVA`.

P25. Student guide of Lounesto**: selected exercises.**
Do some of the exercises No.11 to No.32 of the student guide of Lounesto [7, p. 26]. You find selected solutions on page 1.

P26. Further reading: Lorentz**ian 2-space and Special Relativity.**
Read the text of Sobczyk [13, pp. 15–20] about Clifford algebra in Lorentz plane and Special Relativity. Use Eigenmath's `EVA` along your way.

P27. Further reading: Minkowski **4-space and Special Relativity.**
Read the text of Snygg [11, pp. 27–37] about Clifford algebra in Minkowski 4-space and get a "small dose of Special Relativity". Use Eigenmath's `EVA` along your way.

References

[1] Arnon, I., Cottrill, J., Dubinsky & al. (2014): *APOS Theory. A Framework for Research and Curriculum Development in Mathematics Education.* New York: Springer-Verlag. DOI 10.1007/978-1-4614-7966-6

[2] Borota, N. A.(1999): *An analysis of spacetime numbers.* Theses and Dissertations. 1771. `https://rdw.rowan.edu/etd/1771`

[3] Dombrowski, P. (1970): *Differentialrechnung I und Abriß der linearen Algebra.* Mannheim: Bibliographisches Institut.

[4] Ebbinghaus, H.-D.. & al. (1983): *Zahlen.* Berlin: Springer-Verlag.

[5] Eyheramendy, B.(2012): *Clifford Algebra Calculator - The EVA package.* url: `http://beyhfr.free.fr/EVA2/crbst_38.html`

[6] Lindner, W. (2003): CAS-supported Multiple Representations in Elementary Linear Algebra - The Case of the Gaussian Algorithm. In: *ZDM* Vol. 35 (2), S. 36 - 42.

[7] Lounesto, P.(1987): *CLICAL, a calculator type computer program. Student Guide.* url: `https://users.aalto.fi/~ppuska/mirror/Lounesto/CLICAL.htm`

[8] Macdonald, A. (^{3}2010): *Linear and Geometric Algebra.* Amazon: Marston Gate. ISBN 9 781 453 854938

[9] Marsden, J., Weinstein, A. (1985): *Calculus II.* New York: Springer-Verlag.

[10] Seeburger, P. (2018): *CalcPlot3D.* url: `https://c3d.libretexts.org/CalcPlot3D/index.html`

[11] Snygg, J. (2012): *A New Approach to Differential Geometry using Clifford's Geometric Algebra.* New York: Birkhäuser (Springer).

[12] Sobczyk, G. (1995): "The Hyperbolic Number Plane." In: The College Mathematics Journal, 26(4), pp. 268–280

[13] Sobczyk, G. (2019): *Matrix Gateway to Geometric Algebra, Spacetime and Spinors.* Amazon: Wrockaw. ISBN 9 781 704 596624

[14] Steeb, W.-H., Lewien, D. (1992): *Algorithms and Computations with Reduce.* Mannheim: Bibliographisches Institut.

[15] Tall, D. & Vinner, S. (1991): Concept Images and Concept Definition in Mathematics with Particular reference to Limits and Continuity. In: *Educational Studies in Mathematics*, NCTM, vol. 12, pp. 49-63.

[16] WEIGT, G. (2021): EIGENMATH *Homepage.*
url: `https://georgeweigt.github.io`

[17] WEIGT, G. (2021): EIGENMATH *online.*
url: `https://georgeweigt.github.io/eigenmath-demo.html`

[18] WEIGT, G. (2021): EIGENMATH *Quaternion demo.*
url: `https://georgeweigt.github.io/quaternions.html`

[19] WOLFRAM|alpha. Quaternions.
url: `https://www.wolframalpha.com/examples/mathematics/algebra/quaternions/`

$\bowtie$

Links checked 03.03.2021, wL

Dr. Wolfgang Lindner
Leichlingen, Germany
dr.w.g.Lindner@gmail.com
2021