2 Projektionen, Transformationen und Bewegungen

2.1 Projektive Erweiterung der euklidischen Ebene und des euklidischen Raumes

Für viele Anwendungen ist das Arbeiten in der euklidischen Ebene E_2, bezogen auf kartesische Koordinaten $\{x, y\}$ nicht ausreichend (vgl. Abb.10). Eine Erweiterung des Koordinatenbegriffs und damit eine Erweiterung von E_2 ist angebracht. Dies wird im folgenden durchgeführt.

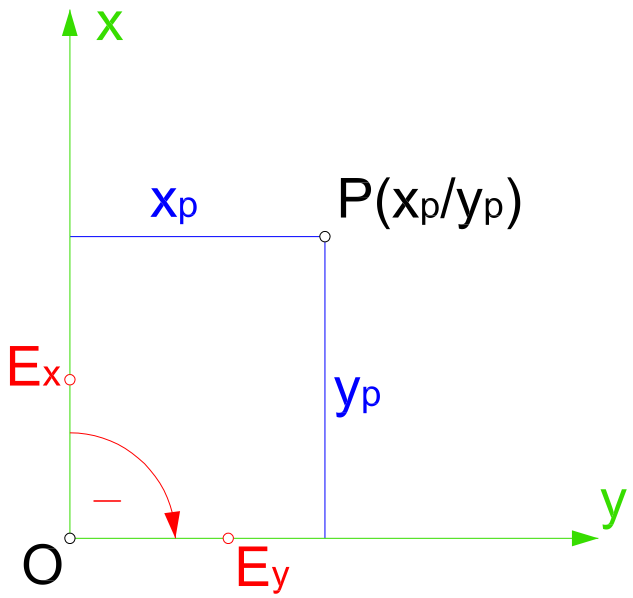


Abbildung 10: Kartesisches Koordinatensystem in der Ebene

Aus der Elementargeometrie ist wohlbekannt, wie man nach Zugrundelegung eines kartesischen Koordinatensystems $\{U;\ x,\ y\}$ jeden Punkt $X \in E_2$ durch kartesische Koordinaten $\{x, y\}$ beschreiben kann. Umgekehrt legt jedes Zahlenpaar (x, y) bezüglich $\{U;\ x,\ y\}$ eindeutig einen Punkt $X \in E_2$ fest.
Sei nun Q ein weiterer Punkt auf der Verbindungsgeraden UP, dann kann Q nach dem Strahlensatz durch $Q(\mu x, \mu y)$ beschrieben werden; umgekehrt stellt $(\mu x, \mu y)$ bei festem $X \neq U$ für ein bestimmtes μ einen Punkt Q auf UX dar. Wollen wir den Fernpunkt von UX ins Spiel bringen (natürlich rein heuristisch), dann ist der Grenzübergang $\mu \to \pm\infty$ durchzuführen. Dies liefert das sinnlose Paar (∞, ∞). Wir wollen im folgenden durch einen Kunstgriff diesen Grenzübergang möglich machen:

1. Ist $X = (x, y) \in E_2$ und (x_0, x_1, x_2) ein geordnetes Tripel mit $x_0 \neq 0$, dann soll dieses Tripel genau dann dem Punkt X zugewiesen werden, wenn gilt:

$$x = \frac{x_1}{x_0} \quad , \qquad y = \frac{x_2}{x_0} \tag{38}$$

   Dann ist $(\lambda x_0, \lambda x_1, \lambda x_2)$ mit $\lambda \neq 0$ demselben Punkt X zugewiesen, denn $\frac{\lambda x_1}{\lambda x_0} = \frac{x_1}{x_0} = x$, $\frac{\lambda x_2}{\lambda x_0} = y$

   Sind umgekehrt zwei Zahlentripel (x_0, x_1, x_2), (x_0', x_1', x_2') mit $x_0, x_0'| \neq 0$ demselben Punkt X zugewiesen, dann existiert stets ein $\lambda \neq 0$ so, dass $x_i' = \lambda x_i$ $(i = 0, 1, 2)$. Denn aus

$\frac{x_i'}{x_0'} = \frac{x_i}{x_0}$ $(i = 1, 2) \Rightarrow x_i' = \frac{x_0'}{x_0} x_i$. Wird $\frac{x_0'}{x_0} = \lambda$ gesetzt, dann ist $\lambda \neq 0$ und man erhält $x_i' = \lambda x_i$ $(i = 0, 1, 2)$.

Es gilt also: Jedem Punkt $X \in E_2$ ist umkehrbar eindeutig ein Zahlentripel $(x_0 : x_1 : x_2)$ mit $x_0 \neq 0$ zugeordnet. Die Koordinaten $(x_0 : x_1 : x_2)$ heißen *homogene* Koordinaten. x_0 heißt *homogenisierende* Koordinate.

2. Für die Koordinaten $(\mu x, \mu y)$ $(\mu \neq 0)$ von Q gilt also:

$$(\mu x, \mu y) = (x_0 : \mu x_1 : \mu x_2) = (\frac{x_0}{\mu} : x_1 : x_2)$$

Führt man jetzt auf der rechten Seite den Grenzübergang $\mu \to \pm\infty$ durch, so ergibt sich $(0 : x_1 : x_2)$. Diesem Tripel entspricht kein Punkt der Geraden UX in E_2, aber es ist naheliegend rein formal auch dieses Tripel zuzulassen und die Sprechweise einzuführen: $(0 : x_1 : x_2)$ beschreibt den *Fernpunkt (uneigentlichen Punkt)* auf UX. Da man zum selben Tripel gelangt, egal ob $\mu \to +\infty$ oder $\mu \to -\infty$ strebt, so ist hierdurch der Geraden UX in E_2, ein *einziger* Fernpunkt zugewiesen worden. Die Gerade, erweitert durch einen einzigen Fernpunkt, heißt *projektive Gerade.*

3. Das Tripel $(0 : 0 : 0)$ beschreibt weder einen eigentlichen noch einen uneigentlichen Punkt auf UX. Wegen $x_0 = 0$ beschreibt nämlich $(0 : 0 : 0)$ keinen eigentlichen Punkt auf UX; $(0 : x : y) = (0 : 0 : 0)$ ist aber auch kein uneigentlicher Punkt, sonst wäre $X = U$ im Widerspruch zur Konstruktion von UX. Das *triviale* Tripel $(0 : 0 : 0)$ ist also in Hinkunft auszuscheiden.

4. Wird obige Erweiterung für alle Geraden der Ebene E_2 durchgeführt, so ist die Gesamtheit der auftretenden Fernpunkte durch $x_0 = 0$ gekennzeichnet. Die durch Hinzunahme der Fernpunkte ihrer Geraden erweiterte euklidische Ebene heißt *projektive Ebene* P_2. $x_0 = 0$ ist eine lineare Gleichung. Dies legt die Sprechweise nahe: Alle Fernpunkte einer projektiven Ebene liegen auf der *Ferngeraden* der Ebene.

5. Zwei parallele Geraden schneiden sich in einem Fernpunkt. Das, was zwei parallele Geraden gemeinsam haben, nennt man ihre *Richtung.* Hiermit erhält man: Jeder Richtung $\overrightarrow{UX} = (x, y)$ der Ebene entspricht ein Fernpunkt $(0 : x : y)$ dieser Ebene.

Völlig analoge Überlegungen kann man für den dreidimensionalen euklidischen Raum E_3 (Anschauungsraum) durchführen. Wir beziehen diesen Raum auf ein *kartesisches Koordinatensystem* $\{O; x, y, z\}$ mit dem Ursprung U und den Achsen x, y, z (vgl. Abbildung 11). Die Achsen stehen *paarweise aufeinander orthogonal* (normal).
Wie die Abbildung 11 zeigt, kann jedem Punkt $\in E_3$ *umkehrbar eindeutig* ein *Zahlentripel* (x_p, y_p, z_p) - seine Koordinaten - zugeordnet werden. Dies sieht man z.B. so, dass man P *normal* auf die $[xy]$-Ebene projiziert. Der Projektionspunkt P' *(Grundriss von p)* wird in der euklidischen $[xy]$-Ebene durch die Koordinaten (x_p, y_p) erfaßt. Über P' wird die z-Koordinate zur Festlegung von P verwendet. Genauso könnte man über die *Normalprojektion* P'' von P auf die $[yz]$-Ebene vorgehen. P'' heißt der *Aufriss* von P. Man erkennt damit ein *wichtiges Resultat* für die *Darstellende Geometrie*: *Durch die beiden Normalprojektionen P' und P'' ist ein Punkt $P \in E_3$ eindeutig bestimmt.*
An der Abbildung 11 kann man nun eine analoge Überlegung anstellen wie an der Abbildung 10. Ein Punkt $Q \in g = OX$ hat die Koordinaten $Q(\mu x, \mu y, \mu z)$ mit $\mu \in \mathcal{R}$. Für $\mu \to \pm\infty$ erhält man den *Fernpunkt* G_u, der sich koordinatenmäßig so nicht beschreiben läßt.

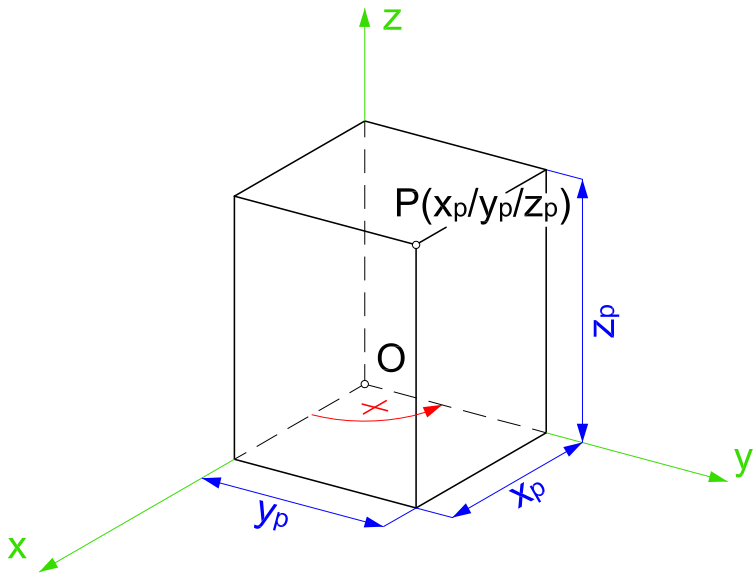


Abbildung 11: Kartesisches Koordinatensystem im Raum

Definition 2.1 *Die projektiven (homogenen) Koordinaten* $(x_0 : x_1 : x_2 : x_3)$ *des Punktes* $X \in E_3$ *werden definiert durch*

$$x = \frac{x_1}{x_0}, \quad y = \frac{x_2}{x_0}, \quad z = \frac{x_3}{x_0}; \quad x_0 \neq 0. \tag{39}$$

x_0 *heißt homogenisierende Koordinate. Für* $x_0 = 1$ *erhält man wieder die kartesischen Koordinaten* x, y, z.

1. Jedem Punkt $X \in E_3$ ist umkehrbar eindeutig ein Quadrupel $(x_0 : x_1 : x_2 : x_3)$ mit $x_0 \neq 0$ zugeordnet.

2. Es ist $Q(\mu x, \mu y, \mu z) = Q(x_0 : \mu x_1 : \mu x_2 : \mu x_3) = Q(\frac{x_0}{\mu} : x_1 : x_2 : x_3)$ und man findet $\lim\limits_{\mu \to \pm\infty} Q(\frac{x_0}{\mu} : x_1 : x_2 : x_3) = G_u(0 : x_1 : x_2 : x_3)$ Damit ist es gelungen den *Fernpunkt* G_u von g *koordinatenmäßig zu beschreiben.* Wird g durch den Fernpunkt G_u erweitert, so erhält man die *projektive Gerade* g^0.

3. $(0 : 0 : 0 : 0)$ beschreibt keinen Punkt.

4. Der euklidische Raum E_3, erweitert durch die Fernpunkte aller Geraden von E_3, heißt *projektiver Raum* P_3.

5. Alle parallelen Geraden in E_3 besitzen denselben Fernpunkt, d.h. *dieselbe Richtung.* Jeder Richtung, festgelegt durch einen Vektor $\overrightarrow{UX} = \{x, y, z\}$ entspricht ein *Fernpunkt* $(0 : x : y : z)$.

6. *Die Menge der Fernpunkte* in E_2 oder E_3 ist durch $x_0 = 0$, d.h. eine lineare Gleichung festgelegt.

   Da eine lineare Gleichung in E_2 (E_3) eine Gerade (Ebene) bestimmt, definiert man: *Die Menge der Fernpunkte von* E_2 (E_3) *bestimmen die Ferngerade (Fernebene) von* E_2 (E_3).

Satz 2.1 *Bezüglich eines kartesischen Koordinatensystems ist jedem Punkt der euklidischen Ebene bzw. des euklidischen Raumes ein homogenes Tripel* $(x_0 : x_1 : x_2)$ *bzw. ein homogenes Quadrupel* $(x_0 : x_1 : x_2 : x_3)$ *mit* $x_0 \neq 0$ *zugewiesen. Ein nichttriviales Tripel (Quadrupel) mit* $x_0 = 0$ *heißt ein Fernpunkt der Ebene (des Raumes) und legt eine Richtung fest. Zwei eigentliche oder uneigentliche Punkte sind gleich, wenn ihre homogenen Koordinaten proportional sind.*

Ganz ähnlich wie man einem Punkt X in E_3 die homogenen Koordinaten $(x_0 : x_1 : x_2 : x_3)$ zuordnet, so kann man auch einer Ebene Koordinaten zuordnen. Um dies abzuleiten gehen wir von der Gleichung einer Ebene in Vektorform aus. Eine Ebene in E_3 ist bestimmt, wenn man 3 Punkte A, B, C von ihr kennt. Seien die zu den Punkten führenden Ortsvektoren $\overrightarrow{OA} = \vec{a}, \overrightarrow{OB} = \vec{b}, \overrightarrow{OC} = \vec{c}$ (Abb. 12).

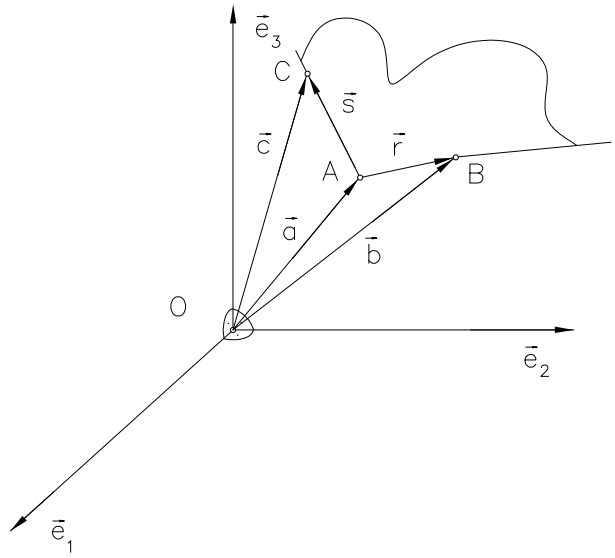


Abbildung 12: Ebenengleichung

Dann erhält man die Ebenengleichung der Verbindungsebene $\varepsilon = (A, B, C)$

$$\varepsilon : \vec{x} = \vec{a} + \lambda(\vec{b} - \vec{c}) + \mu(\vec{c} - \vec{a}) = \vec{a} + \lambda\ \vec{r} + \mu\ \vec{s}, \qquad \lambda, \mu \in \mathbb{R} \tag{40}$$

Die Parameter $\lambda,\ \mu$ kann man eliminieren durch skalare (innere) Multiplikation mit dem Vektor $\vec{n} = \vec{r} \times \vec{s}$

$$\vec{n}^T \vec{x} = \vec{n}^T(\vec{a} + \lambda(\vec{b} - \vec{c}) + \mu(\vec{c} - \vec{a})) = \vec{n}^T \vec{a}$$

wegen $\vec{n}^T \vec{r} = \vec{n}^T \vec{s} = 0$. Damit erhalten wir

$$\vec{n}^T \vec{x} - \vec{n}^T \vec{a} = n_1 x_1 + n_2 x_2 + n_3 x_3 + n_0 = 0 \tag{41}$$

Der Vektor $\vec{n} = (n_1, n_2, n_3)^T$ ist ein Normalvektor der Ebene ε und $n_0 = -\vec{n}^T \vec{a} = \det(\vec{a}, \vec{b}, \vec{c})$. Wir bezeichnen $(n_0 : n_1 : n_2 : n_3)$ als die homogenen Koordinaten der Ebene ε.
Damit haben wir jetzt das Rüstzeug um die für das CAD und die Darstellende Geometrie (Darstellungsmethoden) so wichtigen Projektionen aus dem 3-dim Raum auf eine Bildebene (Zeichenblatt, Computerbildschirm) herzuleiten.

2.2 Projektionen

Gegeben sei eine Bildebene π durch die homogenen Koordinaten $(u_0 : u_1 : u_2 : u_3)$ und ein Projektionszentrum $Z(z_0 : z_1 : z_2 : z_3)$. Es soll gelten $u_0 z_0 + u_1 z_1 + u_2 z_2 + u_3 z_3 \neq 0$ (diese Bedingung verhindert, dass das Projektionszentrum in der Bildebene liegt!) und $\sqrt{u_1^2 + u_2^2 + u_3^2} = 1$. Für die Projektion $\varphi : E_3 \to \pi$ haben wir nun folgenden Fälle zu betrachten:

1. $z_0 \neq 0$, **Zentralprojektion**, wir können setzen $Z(1 : z_1 : z_2 : z_3)$.
2. $z_0 = 0$, **Parallelprojektion** mit der Projektionsrichtung $\vec{r} = -\vec{z} = (z_1, z_2, z_3)^T$
3. **Normalprojektion** $\vec{z} = \vec{u} = (u_1, u_2, u_3)^T$.

Um das Folgende übersichtlich gestalten zu können werden wir die vier homogenen Koordinaten von Bildebene und Projektionszentrum zu einem Vierervektor zusammenfassen und schreiben $\underline{\vec{u}} = (u_0, u_1, u_2, u_3)$ usw.
Jede Projektionsgerade p_X durch X (Verbindungsgerade des Punktes mit dem Zentrum) läßt sich darstellen

$$\underline{\vec{y}} = t_0 \underline{\vec{z}} + t_1 \underline{\vec{x}}$$

Für ihren Schnittpunkt mit der Bildebene gilt $X^\varphi = \underline{\vec{x}}^\varphi = s_0 \underline{\vec{z}} + s_1 \underline{\vec{x}}$. Und aus $\underline{\vec{u}}^T \underline{\vec{x}}^\varphi = 0$ folgt durch Einsetzen

$$s_0 \ \underline{\vec{u}}^T \underline{\vec{z}} + s_1 \ \underline{\vec{u}}^T \underline{\vec{x}} = 0 \quad \Rightarrow$$

$$s_0 \ : \ s_1 = -\underline{\vec{u}}^T \ \underline{\vec{x}} : \underline{\vec{u}}^T \ \underline{\vec{z}} \quad \Rightarrow$$

$$\underline{\vec{x}}^\varphi = -(\underline{\vec{u}}^T \ \underline{\vec{x}})\underline{\vec{z}} + (\underline{\vec{u}}^T \ \underline{\vec{z}})\underline{\vec{x}}$$

Diese Gleichung läßt sich nach dem Ausmultiplizieren als Matrizengleichung schreiben

$$\underline{\vec{x}}^\varphi = \mathbf{P} \underline{\vec{x}}$$

mit

$$\mathbf{P} = \begin{bmatrix} u_1 z_1 + u_2 z_2 + u_3 z_3 & -u_1 z_0 & -u_2 z_0 & -u_3 z_0 \\ -u_0 z_1 & u_0 z_0 + u_2 z_2 + u_3 z_3 & -u_2 z_1 & -u_3 z_1 \\ -u_0 z_2 & -u_1 z_2 & u_1 z_1 + u_0 z_0 + u_3 z_3 & -u_3 z_2 \\ -u_0 z_3 & -u_1 z_3 & -u_2 z_3 & u_1 z_1 + u_2 z_2 + u_0 z_0 \end{bmatrix}$$

$\mathbf{P}$ heißt Projektionsmatrix. Sie vermittelt die Projektion aus dem Zentrum Z auf die Bildebene π. Da die Bildebene im Koordinatensystem Σ (U, x, y, z) noch ganz allgemein liegt, ist noch eine Koordinatentransformation durchzuführen, die eine Koordinatenebene in die Bildebene (Bildkoordinatensystem Σ_B $(O, y_1, y_2$ bringt. Wir vereinbaren dazu:

- Der Ursprung des Bildkoordinatensystems soll im Bildpunkt des Ursprung des räumlichen Koordinatensystems liegen
- das Bild der z-Achse soll mit der y_2-Achse des Bildkoordinatensystems übereinstimmen

Diese Koordinatentransformation wird durch

$$\underline{\vec{x}} = \mathbf{T} \underline{\vec{y}}$$

mit der Matrix $\mathbf{T} = \begin{pmatrix} 1 & \vec{o}^T \\ \vec{t} & \mathbf{D} \end{pmatrix}$ und $\mathbf{D} = (\vec{t}_1, \vec{t}_2, \vec{t}_3)$. Die Einträge in der Matrix beschreiben: $\vec{t}$ stellt den Ursprung O des Bildkoordinatensystems im Raumkoordinatensystem Σ dar. Die Spalten sind die Richtungen der Achsen y_i des Bildkoordinatensystems in Σ gemessen. Da wir aber nicht die Koordinaten in Σ berechnen wollen, sondern in Σ_B benötigen wir die inverse Transformation $\mathbf{T}^{-1}$ und erhalten zusammen mit der Projektionsmatrix $\mathbf{P}$:

$$\underline{\vec{y}}^{\varphi} = \mathbf{T}^{-1}\mathbf{P}\underline{\vec{x}}, \qquad \mathbf{T}^{-1} = \begin{pmatrix} 1 & \vec{o}^T \\ -\mathbf{D}^T\vec{t} & \mathbf{D}^T \end{pmatrix} \tag{42}$$

$$\mathbf{M} := \mathbf{T}^{-1}\mathbf{P}$$

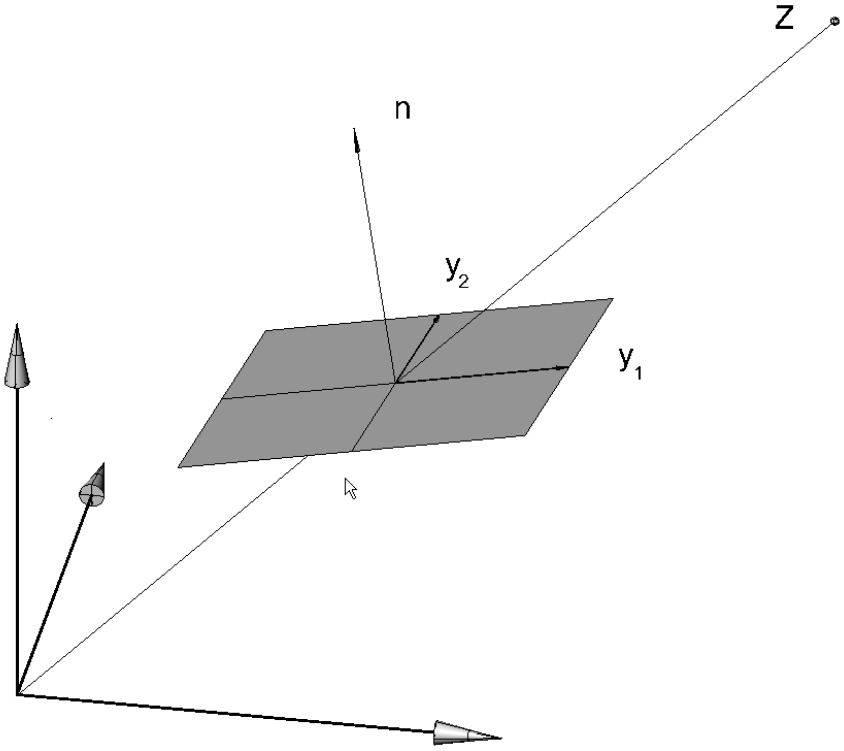


Abbildung 13: Bildkoordinatensystem

Die Einträge der Matrix $\mathbf{T}$ können nun wie folgt berechnet werden:

1. Da das Bildkoordinatensystem in der Projektionsebene liegt gilt: $y_3 = \vec{t}_3 = \dfrac{\vec{u}}{|\vec{u}|}$

2. Zur Berechnung von $\vec{t}_2$ muss die Forderung nach der Richtung der Projektion der z-Achse parallel zum linken Bildrand erfüllt werden. Dies erfordert eine Fallunterscheidung

   - wenn gilt $z_1 = z_2 = 0$, dann definieren wir einen Vektor $\vec{v} := (0, u_3z_3, -u_2z_3)^T$ sonst $\vec{v} := (-u_3z_1, u_3z_2, u_0z_0 + u_1z_1 + u_2z_2)^T$
   - wenn $u_3z_0 = 0$ dann ist $\vec{t}_2 = \dfrac{\vec{v}}{|\vec{v}|}$ sonst:
   - wir definieren $\vec{v}' := \dfrac{u_0}{-\underline{\vec{u}}^T\ \underline{\vec{z}}}\vec{z} - \dfrac{1}{u_3z_0}\underline{\vec{v}}$ und $\vec{t}_2 = \dfrac{\vec{v}'}{|\vec{v}'|}$

3. $\vec{t}_1 = \vec{t}_2 \times \vec{t}_2$

4. $\vec{t} = \dfrac{u_0}{-\underline{\vec{u}}^T \ \underline{\vec{z}}} \vec{z}$, $(O = U^\varphi)$.

Mit diesem Algorithmus kann nun die Matrix der Transformation in das Bildkoordinatensystem beschrieben werden:

$$\mathbf{T^{-1}} := \begin{pmatrix} 1 & \vec{0}^t \\ -\vec{t}_1^T \vec{t} & \vec{t}_1^T \\ -\vec{t}_2^T \vec{t} & \vec{t}_2^T \\ -\vec{t}_2^T \vec{t} & \vec{t}_2^T \end{pmatrix}$$

Damit können sämtliche Projektionen aus dem dreidimensionalen Raum auf eine zweidimensionale Bildebene beschrieben werden. Für die konkrete Berechnung verschiedenener Projektionen wird auf die Maple-Notebooks Projektionsmatrix1-3, die von der homepage des Institutes heruntergeladen werden können. Wir zeigen hier eines der Notebooks:

```
> restart;with(linalg):with(plots):
```

```
Allgemeiner Fall Dreipunktperspektive (Horizont waagrecht u1:u2=z1:z2)
```

```
> Z:=vector([z0,z1,z2,z3]);
```

$$Z := [z0,\ z1,\ z2,\ z3]$$

```
> Eb:=vector([u0,u1,u2,u3]);
```

$$Eb := [u0,\ u1,\ u2,\ u3]$$

```
> X:=vector([x0,x1,x2,x3]);
```

$$X := [x0,\ x1,\ x2,\ x3]$$

```
> P:=matrix(4,4,[u1*z1+u2*z2+u3*z3,-u1*z0,-u2*z0,-u3*z0,-u0*z1,u0*z0+u2
> *z2+u3*z3,-u2*z1,-u3*z1,-u0*z2,-u1*z2,u1*z1+u0*z0+u3*z3,-u3*z2,-u0*z3,
> -u1*z3,-u2*z3,u1*z1+u2*z2+u0*z0]);
```

$$P := \begin{bmatrix} u1\ z1 + u2\ z2 + u3\ z3\ ,\ -u1\ z0\ ,\ -u2\ z0\ ,\ -u3\ z0 \\ -u0\ z1\ ,\ u0\ z0 + u2\ z2 + u3\ z3\ ,\ -u2\ z1\ ,\ -u3\ z1 \\ -u0\ z2\ ,\ -u1\ z2\ ,\ u1\ z1 + u0\ z0 + u3\ z3\ ,\ -u3\ z2 \\ -u0\ z3\ ,\ -u1\ z3\ ,\ -u2\ z3\ ,\ u1\ z1 + u2\ z2 + u0\ z0 \end{bmatrix}$$

```
> rank(P);det(P);
```

$$3$$
$$0$$

```
Berechnung der beiden Transformationsmatrizen, die die Bildtrafo
bringen (auf das Koordinatensystem der Zeichenebene bzw. des
Computerbildschirmes).
```

```
> t[3]:=vector([u1,u2,u3]);
```

$$t_3 := [u1\ ,\ u2,\ u3]$$

```
> nt3:=sqrt(u1^2+u2^2+u3^2);
```

$$nt3 := \sqrt{u1^2 + u2^2 + u3^2}$$

```
> tn[3]:=evalm(t[3]*1/nt3);
```

$$tn_3 := \left[\frac{u1}{\sqrt{u1^2 + u2^2 + u3^2}}, \frac{u2}{\sqrt{u1^2 + u2^2 + u3^2}}, \frac{u3}{\sqrt{u1^2 + u2^2 + u3^2}}\right]$$

```
> vw:=col(P,4);
```

$$vw := [-u3\ z0,\ -u3\ z1\ ,\ -u3\ z2,\ u1\ z1 + u2\ z2 + u0\ z0]$$

```
> v:=vector([vw[2],vw[3],vw[4]]);
```

$$v := [-u3\ z1,\ -u3\ z2,\ u1\ z1 + u2\ z2 + u0\ z0]$$

```
> uTz:=evalm(transpose(Eb)&*Z);
```

$$uTz := u0\ z0 + u1\ z1 + u2\ z2 + u3\ z3$$

```
> t:=scalarmul(-vector([Z[2],Z[3],Z[4]]),u0/(u1*z1+u2*z2+u3*z3));
```

$$t := \left[-\frac{u0\ z1}{u1\ z1 + u2\ z2 + u3\ z3}, -\frac{u0\ z2}{u1\ z1 + u2\ z2 + u3\ z3}, -\frac{u0\ z3}{u1\ z1 + u2\ z2 + u3\ z3}\right]$$

```
> vs:=evalm(scalarmul(-v,1/(u3*z0))-t);
```

$$vs := \left[\frac{z1}{z0} + \frac{u0\ z1}{u1\ z1 + u2\ z2 + u3\ z3}, \frac{z2}{z0} + \frac{u0\ z2}{u1\ z1 + u2\ z2 + u3\ z3}, \frac{-u1\ z1 - u2\ z2 - u0\ z0}{u3\ z0} + \frac{u0\ z3}{u1\ z1 + u2\ z2 + u3\ z3}\right]$$

```
> tn[2]:=map(simplify,evalm(vs*1/(sqrt(vs[1]^2+vs[2]^2+vs[3]^2)))):
> tn[1]:=map(simplify,crossprod(tn[2],tn[3])):
> TI:=matrix(4,4,[1,0,0,0,evalm(-transpose(tn[1])&*t),tn[1][1],tn[1][2]
> ,tn[1][3],evalm(-transpose(tn[2])&*t),tn[2][1],tn[2][2],tn[2][3],evalm
> (-transpose(tn[3])&*t),tn[3][1],tn[3][2],tn[3][3]]):
> map(simplify,TI):
> M:=map(simplify,evalm(TI&*P)):
```

Beispiel:

```
> u0:=-5;u1:=3;u2:=-2;u3:=-2;z0:=1;z1:=3;z2:=-2;z3:=1;
```

$$u0 := -5$$
$$u1 := 3$$
$$u2 := -2$$
$$u3 := -2$$
$$z0 := 1$$
$$z1 := 3$$
$$z2 := -2$$
$$z3 := 1$$

```
> M:=map(simplify,evalm(TI&*P)):MB:=map(simplify,op(M)):
```

Wir berechnen und plotten das Bild des Einheitswürfels (A,B,D,C,E,F,G,H)

```
> A:=vector([1,1,1,0]);B:=vector([1,-1,1,0]);C:=vector([1,-1,-1,0]);Dd:
> =vector([1,1,-1,0]);E:=vector([1,1,1,2]);F:=vector([1,-1,1,2]);G:=vect
> or([1,-1,-1,2]);H:=vector([1,1,-1,2]);Fx:=vector([0,1,0,0]);Fy:=vector
> ([0,0,1,0]);Fz:=vector([0,0,0,1]);
```

$$A := [1,\ 1,\ 1,\ 0]$$
$$B := [1,\ -1,\ 1,\ 0]$$
$$C := [1,\ -1,\ -1,\ 0]$$
$$Dd := [1,\ 1,\ -1,\ 0]$$
$$E := [1,\ 1,\ 1,\ 2]$$

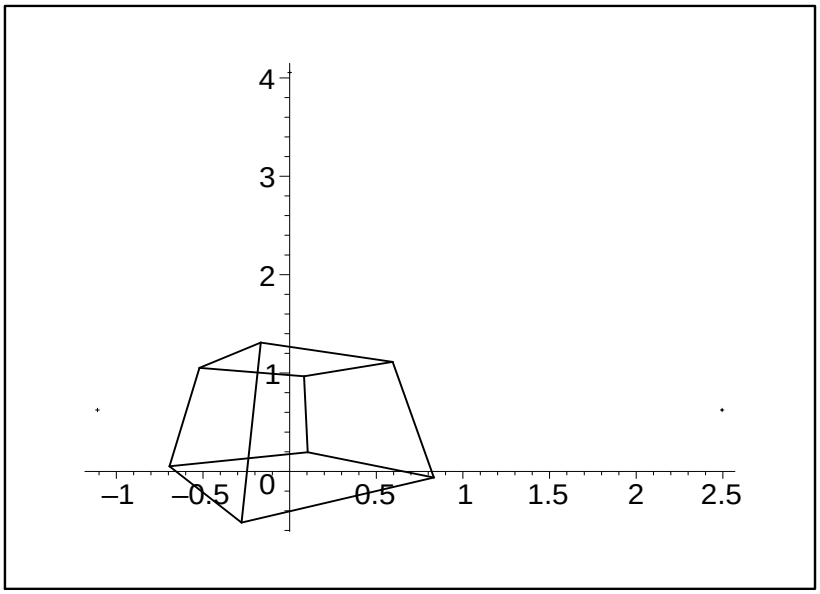
$$F := [1, -1, 1, 2]$$
$$G := [1, -1, -1, 2]$$
$$H := [1, 1, -1, 2]$$
$$Fx := [0, 1, 0, 0]$$
$$Fy := [0, 0, 1, 0]$$
$$Fz := [0, 0, 0, 1]$$

```
> Ap:=evalm(MB&*A):Bp:=evalm(MB&*B);Cp:=evalm(MB&*C);Dp:=evalm(MB&*Dd);
> Ep:=evalm(MB&*E);Fp:=evalm(MB&*F);Gp:=evalm(MB&*G);Hp:=evalm(MB&*H);Fx
> p:=evalm(MB&*Fx);Fyp:=evalm(MB&*Fy);Fzp:=evalm(MB&*Fz);
```

$$Bp := \left[16, \frac{6\sqrt{13}}{13}, \frac{30\sqrt{221}}{143}, 0\right]$$
$$Cp := \left[12, -\frac{30\sqrt{13}}{13}, \frac{6\sqrt{221}}{143}, 0\right]$$
$$Dp := \left[6, -\frac{6\sqrt{13}}{13}, -\frac{30\sqrt{221}}{143}, 0\right]$$
$$Ep := \left[14, \frac{30\sqrt{13}}{13}, \frac{150\sqrt{221}}{143}, 0\right]$$
$$Fp := \left[20, \frac{6\sqrt{13}}{13}, \frac{186\sqrt{221}}{143}, 0\right]$$
$$Gp := \left[16, -\frac{30\sqrt{13}}{13}, \frac{162\sqrt{221}}{143}, 0\right]$$
$$Hp := \left[10, -\frac{6\sqrt{13}}{13}, \frac{126\sqrt{221}}{143}, 0\right]$$
$$Fxp := \left[-3, \frac{12\sqrt{13}}{13}, -\frac{18\sqrt{221}}{143}, 0\right]$$
$$Fyp := \left[2, \frac{18\sqrt{13}}{13}, \frac{12\sqrt{221}}{143}, 0\right]$$
$$Fzp := \left[2, 0, \frac{6\sqrt{221}}{11}, 0\right]$$

```
> Wuerfel:=plot([[Ap[2]/Ap[1],Ap[3]/Ap[1]],[Bp[2]/Bp[1],Bp[3]/Bp[1]],[C
> p[2]/Cp[1],Cp[3]/Cp[1]],[Dp[2]/Dp[1],Dp[3]/Dp[1]],[Ap[2]/Ap[1],Ap[3]/A
> p[1]],[Ep[2]/Ep[1],Ep[3]/Ep[1]],[Fp[2]/Fp[1],Fp[3]/Fp[1]],[Gp[2]/Gp[1]
> ,Gp[3]/Gp[1]],[Hp[2]/Hp[1],Hp[3]/Hp[1]],[Ep[2]/Ep[1],Ep[3]/Ep[1]]],sty
> le=line,color=black,thickness=3):
> Wuerfel1:=plot([[Bp[2]/Bp[1],Bp[3]/Bp[1]],[Fp[2]/Fp[1],Fp[3]/Fp[1]]],
> style=line,color=black,thickness=3):
> Wuerfel2:=plot([[Cp[2]/Cp[1],Cp[3]/Cp[1]],[Gp[2]/Gp[1],Gp[3]/Gp[1]]],
> style=line,color=black,thickness=3):
> Wuerfel3:=plot([[Dp[2]/Dp[1],Dp[3]/Dp[1]],[Hp[2]/Hp[1],Hp[3]/Hp[1]]],
> style=line,color=black,thickness=3):
```

```
> Fernp:=plot([[Fxp[2]/Fxp[1],Fxp[3]/Fxp[1]],[Fyp[2]/Fyp[1],Fyp[3]/Fyp[
> 1]],[Fzp[2]/Fzp[1],Fzp[3]/Fzp[1]]],style=point,color=black):
> display(Wuerfel,Wuerfel1,Wuerfel2,Wuerfel3,Fernp);
```



Neben den am Anfang dieses Abschnittes genannten Bedingungen für spezielle Projektionen, sollen noch zwei spezielle Axonometrien erwähnt werden, die von keinem der bekannten CAD-Systeme geliefert werden, dennoch aber in der Architekturdarstellung eine wichtige Rolle spielen:

1. Horizontale Axonometrien (Militärriss) ergeben sich für: $z_0 = 0, u_1 = u_2 = 0$
2. Frontale Axonometrien (Schrägriss, Kavalierriss) ergeben sich für: $z_0 = 0, u_2 = u_3 = 0$

2.3 Transformationen

Für das Operieren in CAD-Systemen sind neben den Projektionen die Transformationen, die auf Punktmengen des Raumes oder auch nur einer Ebene ausgeübt werden können extrem wichtig. Wir beginn en mit Transformationen in der Ebene und werden die Betrachtungen dann auf Objekte des 3-dim Raumes ausdehnen.
Wir werden sämtliche Transformationen mit Hilfe von Transformationsmatrizen darstellen. Die Bauart der Matrizen wird durch die Art der Transformation bestimmt.

2.3.1 Translation (Schiebung) in der Ebene

Eine Translation der Punkte $X(x_0 : x_1 : x_2 : x_3)$ einer Ebene ε ist durch die Angabe eines Schiebvektors $\vec{t} = (t_1, t_2)^T$ bestimmt. Da wir mit homogenen Koordinaten arbeiten ist die Transformationsmatrix (der Matrizenoperator) eine 3×3-Matrix. Eine Translation ist gegeben durch

$$X' := \begin{pmatrix} x'_0 \\ x'_1 \\ x'_2 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ t_1 & 1 & 0 \\ t_2 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_0 \\ x_1 \\ x_2 \end{pmatrix} \tag{43}$$

Eine Translation ist in CAD-Systemen durch die Befehle `Schieben, Kopieren, Move, Copy` implementiert.

Aufgabe 2.1 *Eine Translation ist durch den Schiebvektor* $\vec{t} = (2, -4)^T$ *gegeben. Man ermittle die Transformationsmatrix und die Bildkoordinaten des Dreiecks* $A(0,0)$, $B(2,0)$, $C(1,3)$.
Lösung: nach Gl. (43) erhalten wir für die Transformationsmatrix $\mathbf{T}$

$$\mathbf{T} = \begin{pmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -4 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_0 \\ x_1 \\ x_2 \end{pmatrix}$$

Da wir die Matrix mit den homogenen Koordinaten der Punkte multiplizieren müssen erhalten wir z.B. für den Punkt C*:*

$$\mathbf{C}' = \begin{pmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -4 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 3 \end{pmatrix} = \begin{pmatrix} 1 \\ 3 \\ -1 \end{pmatrix}.$$

Durch eine analoge Rechnung erhält man für die Punkte B *und* A *die Bildkoordinaten* $A' = (1 : 2 : -4) = (2, -4)$, $B' = (1 : 3 : -4) = (3, -4)$. *Wir vermerken, dass Ausgangsobjekt (Dreieck) und Bildobjekt (Bilddreieck) kongruent sind; die Seiten sind gleich lang, die Winkel sind gleich groß, die Fläche ist gleich groß usw. ....*

2.3.2 Rotation (Drehung) in der Ebene

Eine Drehung in der Ebene ist durch Angabe des Drehzentrums und des Drehwinkels φ gegeben. Wir geben zuerst die Drehung um den Ursprung des Koordinatensystems an

$$\mathbf{D} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\varphi & -\sin\varphi \\ 0 & \sin\varphi & \cos\varphi \end{pmatrix} \tag{44}$$

Bei der Drehung um einen allgemeinen Punkt $P(p_1, p_2)$ kann man folgendermaßen vorgehen: Man schiebt das Koordinatensystem so, dass der Punkt P Ursprung wird, dreht und schiebt wieder zurück. Die zusammengesetzte Transformation ergibt die Drehung um P:

$$\begin{aligned} \mathbf{D}' = \mathbf{TDT^{-1}} &= \begin{pmatrix} 1 & 0 & 0 \\ p_1 & 1 & 0 \\ p_2 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\varphi & -\sin\varphi \\ 0 & \sin\varphi & \cos\varphi \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ -p_1 & 1 & 0 \\ -p_2 & 0 & 1 \end{pmatrix} = \\ &= \begin{pmatrix} 1 & 0 & 0 \\ p_1 - p_1\cos\varphi + p_2\sin\varphi & \cos\varphi & -\sin\varphi \\ p_2 - p_1\sin\varphi - p_2\cos\varphi & \sin\varphi & \cos\varphi \end{pmatrix} \end{aligned} \tag{45}$$

Bemerkung 2.3.1 *Eine Drehung um* $\pm 180°$ *ist eine Punktspiegelung am Zentrum* Z.

Aufgabe 2.2 *Man ermittele die Matrix der Drehung um den Punkt* $P(2,3)$ *mit dem Drehwinkel* $\varphi = \frac{\Pi}{3}$.
Lösung: Einsetzen in Gl.(45) ergibt

$$\begin{pmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 3 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & -\frac{\sqrt{3}}{2} \\ 0 & \frac{\sqrt{3}}{2} & \frac{1}{2} \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ -3 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 1 + \frac{3\sqrt{3}}{2} & \frac{1}{2} & -\frac{\sqrt{3}}{2} \\ \frac{3}{2}\sqrt{3} & \frac{\sqrt{3}}{2} & \frac{1}{2} \end{pmatrix}$$

2.3.3 Affinitäten:

Affinitäten sind die allgemeinsten Transformationen, die die Fernpunkte nicht transformieren, d.h. die die Parallelität erhalten. Wir geben eine Normalform an, die die x-Achse als Affinitätsachse besitzt.

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & p \\ 0 & 0 & q \end{pmatrix} \tag{46}$$

Wir erhalten durch spezielle Wahl der beiden Parameter p, q eine Reihe von bekannten Transformationen:

1. Für $p = 0, q \neq 0$: orthogonale perspektive Affinitäten.
2. Für $q = -1$ Affinspiegelungen.
3. Für $p = 0, q = -1$ (axiale) Spiegelungen.
4. Für $q = 1$ Scherungen.

Die Determinante der Transformationsmatrix von Spiegelung, Scherung und Affinspiegelung ist gleich eins. Daher sind diese Transformationen inhaltstreu.

Axiale Spiegelung: Diese Transformation ist eine ungleichsinnige Transformation weil sie den Umlaufsinn eines Dreieckes umkehrt. Wir zeigen die Normalform dieser Transformation als Spiegelung an der x-Achse. Sie besitzt die Transformationsmatrix.

$$\mathbf{S} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix} \tag{47}$$

Wie man leicht ausrechnet ist die Determinante von $\mathbf{S}$ gleich -1, was ein Kennzeichen aller Transformationen ist, die den Richtungssinn umkehren. Auf die Darstellung einer Spiegelung um eine allgemeine Achse wollen wir hier nicht eingehen.

Alle bisherigen Transformationen (mit Ausnahme der allgemeinen Affinitäten) haben die Punktmengen in kongruente Punktmengen übergeführt. Wenn die Determinante der Transformationsmatrix $+1$ ist, so spricht man von gleichsinnigen Kongruenztransformationen, wenn sie gleich -1 ist, so heißen die Transformationen ungleichsinnig.

2.3.4 Skalierung

Mit der Skalierung eines Objektes verlassen wir den Bereich der Kongruenztransformationen. Hier geht es um die Vergrößerung und Verkleinerung von Punktmengen. Eine Skalierung ist in CAD Systemen so implementiert, dass man ein Objekt entweder in eine oder zwei Richtungen skalieren kann. Wir erlauben beide Möglichkeiten mit der Transformationsmatrix:

$$\mathbf{Sk} := \begin{pmatrix} 1 & 0 & 0 \\ 0 & s_1 & 0 \\ 0 & 0 & s_2 \end{pmatrix}, \tag{48}$$

dabei ist s_1 die Skalierung in der x-Richtung, s_2 ist die Skalierung in der y-Richtung. $s_1 = s_2$ bedeutet, dass der Skalierungsfaktor in beiden Richtungen gleich ist. Dies liefert eine zentrische Ähnlichkeit vom Ursprung des Koordinatensystems aus.

Wir wenden uns nun räumlichen Transformationen zu. Wir werden uns dabei auf Translationen, Drehungen und Schraubungen beschränken.

2.3.5 Translationen im Raum

Wie in der Ebene ist eine Translation durch Angabe eines Translationsvektors bestimmt:

$$X' := \begin{pmatrix} x'_0 \\ x'_1 \\ x'_2 \\ x'_3 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ t_1 & 1 & 0 & 0 \\ t_2 & 0 & 1 & 0 \\ t_3 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{pmatrix} \tag{49}$$

2.3.6 Drehungen im Raum

Eine Drehung im Raum ist durch die Angabe einer Drehachse und einen Drehwinkel bestimmt. Wir geben der Übersichtlichkeit halber die Normalform einer solchen Drehung als Drehung um die z-Achse des Koordinatensystems an:

$$\mathbf{D} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\varphi & -\sin\varphi & 0 \\ 0 & \sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{50}$$

Die Drehung um einen allgemeine Achse erhält man wenn durch eine Transformation die allgemeine Drehachse zur z-Achse eine neuen Koordinatensystems gemacht wird und dann wieder zurücktransformiert wird. Diese Transformation besteht aber aus einer Drehung und einer Schiebung; daher soll die Zusammensetzung einer Drehung mit einer Schiebung besprochen werden.

2.3.7 Schraubung

Die Zusammensetzung einer Drehung um eine Achse mit einer Schiebung längs dieser Achse heißt *Schraubung*. Wir zeigen die Transformation anhand einer Drehung um die z-Achse und einer Schiebung längs der z-Achse.

$$\mathbf{SCH} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\varphi & -\sin\varphi & 0 \\ 0 & \sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ t_3 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & \\ 0 & \cos\varphi & -\sin\varphi & 0 \\ 0 & \sin\varphi & \cos\varphi & 0 \\ t_3 & 0 & 0 & 1 \end{pmatrix} \tag{51}$$

Eine Schraubung stellt die allgemeinste gleichsinnige Kongruenztransformation im 3-dim Raum dar.

2.4 Bewegung

Die bisher behandelten Transformationen stellen diskrete Transformationen dar. Drehwinkel, Schiebvektoren, Skalierungsfaktoren haben in diesen Transformationen feste Werte. Wenn nun diese Größen von einem (Zeit)Parameter abhängen, dann entstehen Bewegungen. Diese Bewegungen sind euklidische Bewegungen wenn die Transformationen Schiebungen, Drehungen oder Schraubungen sind (oder: gleichsinnige Kongruenztransformationen sind).

Bei dieser Parameterabängigkeit der Transformationen beschreibt nun jeder Punkt eine *Bahnkurve*, jede Kurve eine *Bahnfläche*, jedes Gebiet einen *Körper* (ein solid). Wir zeigen die Auswirkungen dieser Parameterabhängigkeit anhand eines Maple-worksheets:

Worksheet: Schraubung, Drehung, Schiebung

```
> with(plots):with(linalg):
> A:=matrix(4,4,[1,0,0,0,0,cos(t),-sin(t),0,0,sin(t),cos(t),0,p*t,0,0,1]
> );
```

$$A := \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(t) & -\sin(t) & 0 \\ 0 & \sin(t) & \cos(t) & 0 \\ p\,t & 0 & 0 & 1 \end{bmatrix}$$

Die Bahnkurve eines Punktes ist eine Schraublinie. Definition der Schraublinien mit der z-Achse als Schraubachse und dem Parameter p

```
> Schraublinie:=evalm(A&*vector([1,x,y,z]));
```

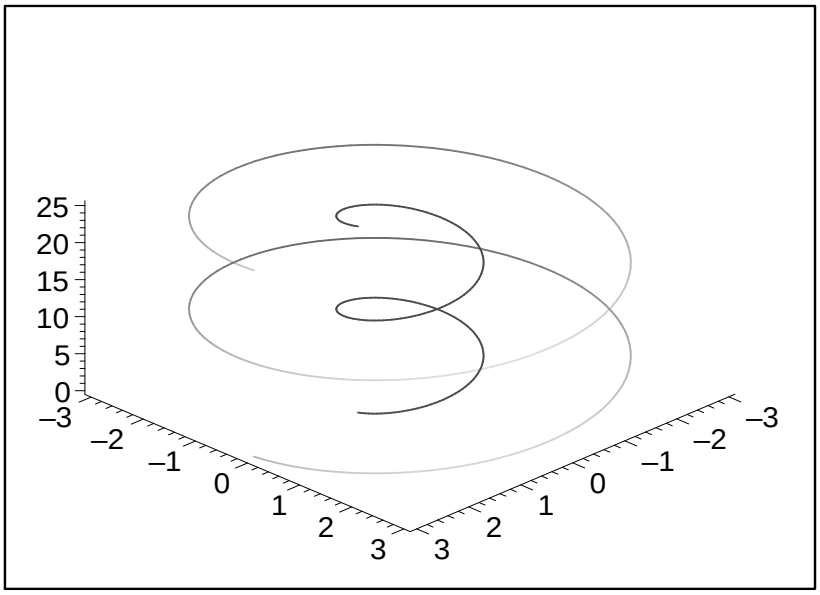
$$\mathit{Schraublinie} := [1,\ \cos(t)\,x - \sin(t)\,y,\ \sin(t)\,x + \cos(t)\,y,\ p\,t + z]$$

```
> Schraublinie1:=subs(x=1,y=0,z=0,p=2,op(Schraublinie));
> Schraublinie2:=subs(x=3,y=0,z=0,p=2,op(Schraublinie));
```

$$\mathit{Schraublinie1} := [1,\ \cos(t),\ \sin(t),\ 2\,t]$$

$$\mathit{Schraublinie2} := [1,\ 3\cos(t),\ 3\sin(t),\ 2\,t]$$

```
> s1:=spacecurve([Schraublinie1[2],Schraublinie1[3],Schraublinie1[4]],
> t=0..4*Pi,thickness=3,numpoints=200,color=red):
> s2:=spacecurve([Schraublinie2[2],Schraublinie2[3],Schraublinie2[4]],
> t=0..4*Pi,thickness=3,numpoints=200,axes=frame):
> display3d({s1,s2});
```




```
> Drehmatrix:=subs(p=0,op(A));
```

$$\mathit{Drehmatrix} := \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(t) & -\sin(t) & 0 \\ 0 & \sin(t) & \cos(t) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

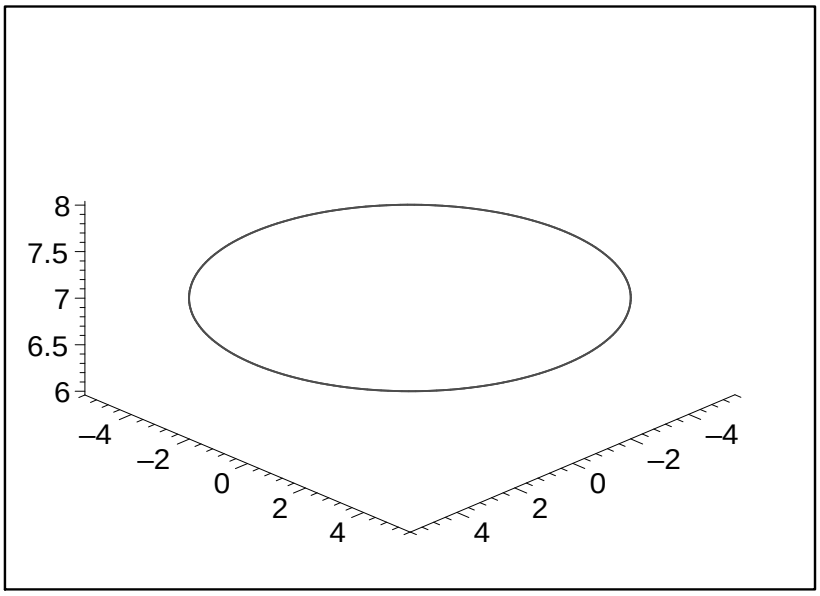
```
> Bahn:=evalm(Drehmatrix&*vector([1,x,y,z]));
```

$$Bahn := [1,\ \cos(t)\,x - \sin(t)\,y,\ \sin(t)\,x + \cos(t)\,y,\ z]$$

```
> Bahn1:=subs(x=2,y=5,z=7,op(Bahn));
```

$$Bahn1 := [1,\ 2\cos(t) - 5\sin(t),\ 2\sin(t) + 5\cos(t),\ 7]$$

```
> spacecurve([Bahn1[2],Bahn1[3],Bahn1[4]],
> t=0..4*Pi,thickness=3,numpoints=200,color=red,axes=FRAME);
```



Wenn der Drehanteil der Bewegung die Einheitsmatrix ist, dann ist die zugehörige Bewegung eine Schiebung :

```
> Schiebmatrix:=matrix(4,4,[1,0,0,0,r*t,1,0,0,q*t,0,1,0,p*t,0,0,1]);
```

$$Schiebmatrix := \begin{bmatrix} 1 & 0 & 0 & 0 \\ r\,t & 1 & 0 & 0 \\ q\,t & 0 & 1 & 0 \\ p\,t & 0 & 0 & 1 \end{bmatrix}$$

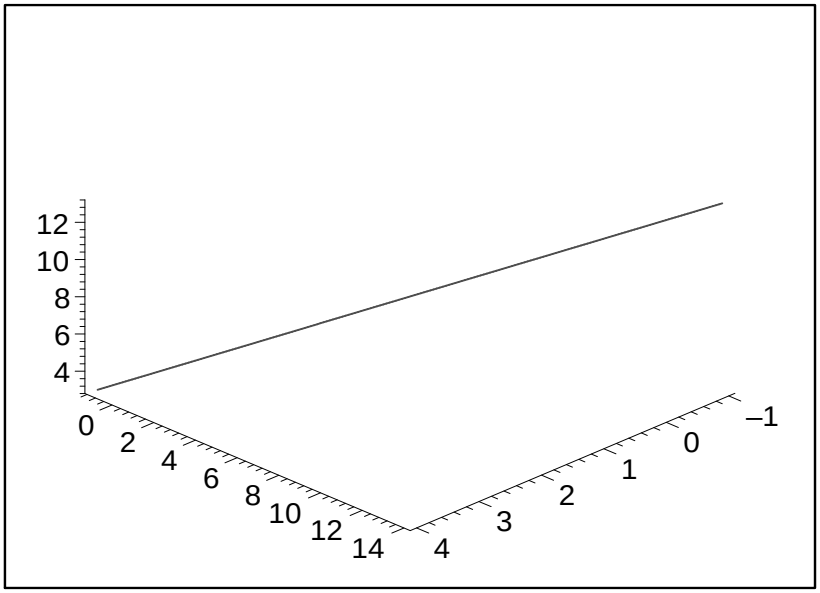
```
> Bahnkurve:=evalm(Schiebmatrix&*vector([1,x,y,z]));
```

$$Bahnkurve := [1,\ r\,t + x,\ q\,t + y,\ p\,t + z]$$

```
> Bahnkurve1:=subs(x=2,y=5,z=7,p=2,q=3,r=-1,op(Bahnkurve));
```

$$Bahnkurve1 := [1,\ -t + 2,\ 3\,t + 5,\ 2\,t + 7]$$

```
> spacecurve([Bahnkurve1[2],Bahnkurve1[3],Bahnkurve1[4]],
> t=-2..3,thickness=3,numpoints=200,color=red,axes=FRAME);
```



Im Schiebanteil der Bewegung können auch nichtlineare Funktionen des Bewegungsparameters t vorkommen.Man spricht dann von einer krummen Schiebung:

```
> KSchiebmatrix:=matrix(4,4,[1,0,0,0,t,1,0,0,3+t,0,1,0,1/2*t^2,0,0,1]);
```

$$KSchiebmatrix := \begin{bmatrix} 1 & 0 & 0 & 0 \\ t & 1 & 0 & 0 \\ 3+t & 0 & 1 & 0 \\ \frac{t^2}{2} & 0 & 0 & 1 \end{bmatrix}$$

```
> Bahnku:=evalm(KSchiebmatrix&*vector([1,x,y,z]));
```

$$Bahnku := \left[1,\, t+x,\, 3+t+y,\, \frac{t^2}{2}+z\right]$$

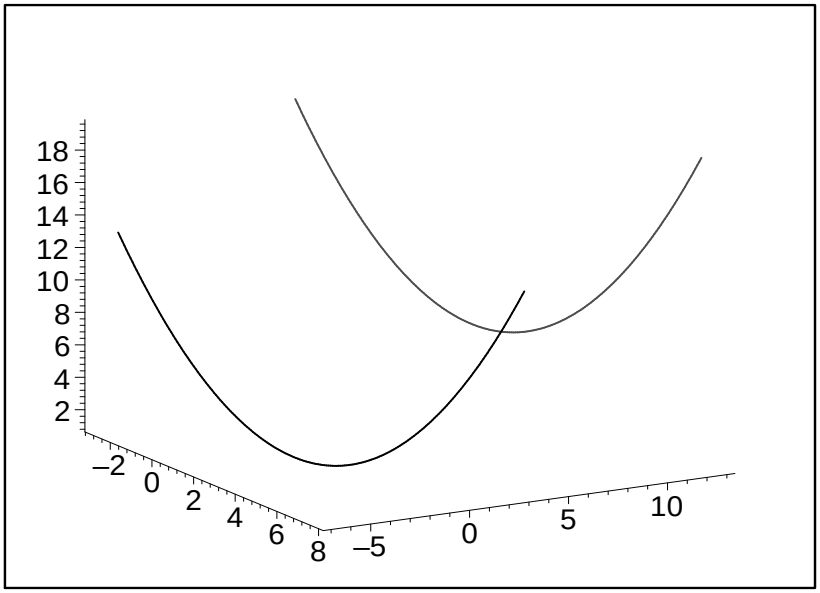
```
> Bahnku1:=subs(x=2,y=5,z=7,op(Bahnku));
```

$$Bahnku1 := \left[1,\, t+2,\, 8+t,\, \frac{t^2}{2}+7\right]$$

```
> Bahnku2:=subs(x=3,y=-5,z=1,op(Bahnku));
```

$$Bahnku2 := \left[1,\, 3+t,\, -2+t,\, \frac{t^2}{2}+1\right]$$

```
> p1:=spacecurve([Bahnku1[2],Bahnku1[3],Bahnku1[4]],
> t=-5..5,thickness=3,numpoints=200,color=red,axes=FRAME):
> p2:=spacecurve([Bahnku2[2],Bahnku2[3],Bahnku2[4]],
> t=-5..5,thickness=3,numpoints=200,color=black,axes=FRAME,orientation=[
> -30,70]):
> display3d({p1,p2});
```


18
16
14
12
10
8
6
4
2
–2
0
2
4
6
8
–5
0
5
10