Aufgabenstellung

<u>bisher:</u> Teachbox implementiert, mit der bekannte Gelenkstellungen angefahren werden können

<u>jetzt:</u> Teachbox, mit der Bewegung entlang der Koordinatenachsen (BKS) realisiert werden kann (xyz-Modus)

dabei: gelenkinterpolierte Bewegung,
d.h. alle Gelenke starten und stoppen gleichzeitig

Orientierungsanpassung 2

Prinzipielles Vorgehen

Gelenkstellung $\xrightarrow{\text{Vorwärtsrechnung}}$ homogene 4x4-Matrix

↓

Positionsveränderung
z.B. um Δz

↓

Addition von Δz auf S(3,4)

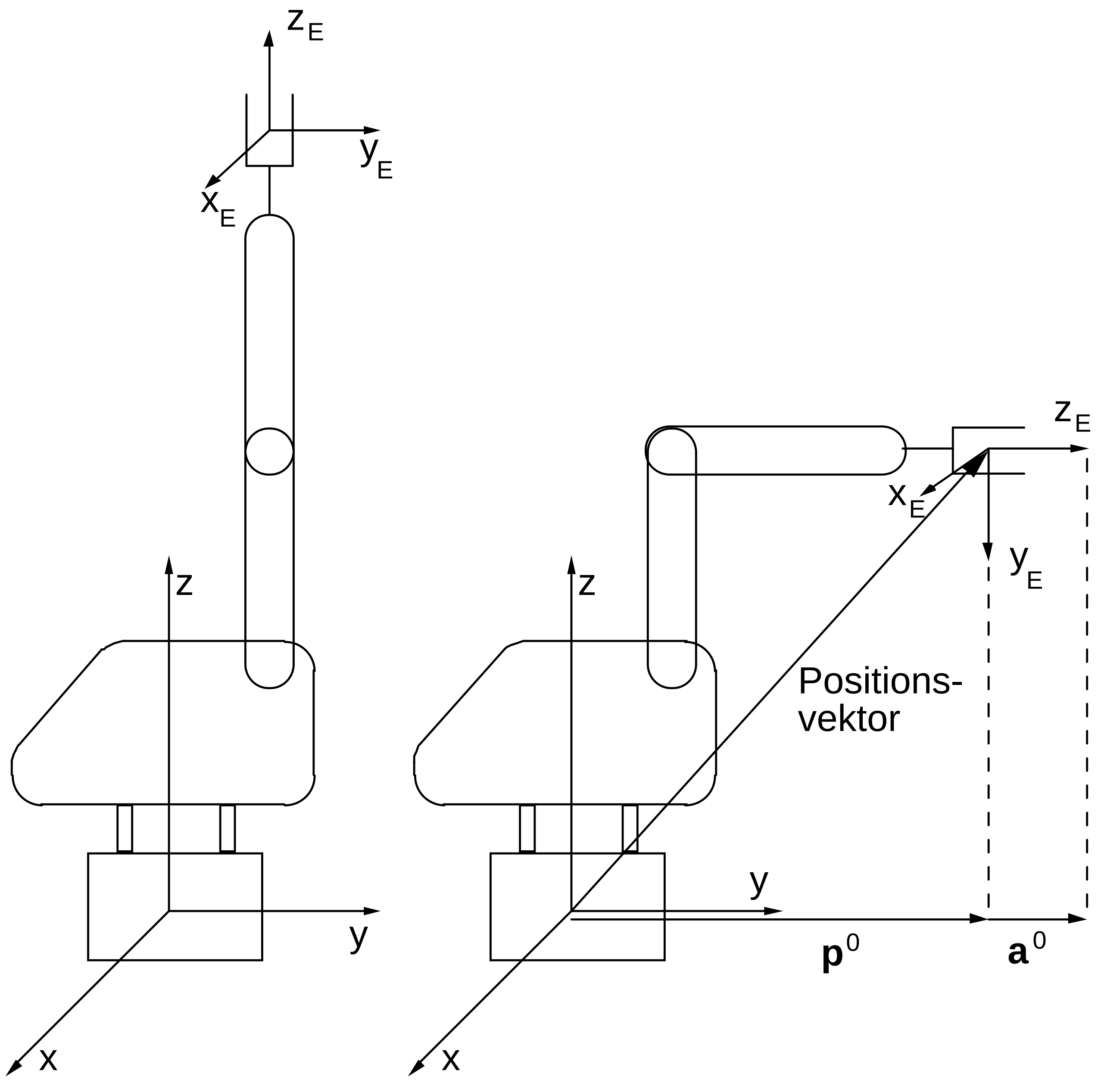
↓

neue Gelenkstellung $\xleftarrow{\text{Rückwärtsrechnung}}$ neue homogene 4x4-Matrix

Stellungsbeschreibung:

$$\mathbf{S} = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Beschreibung RM501:

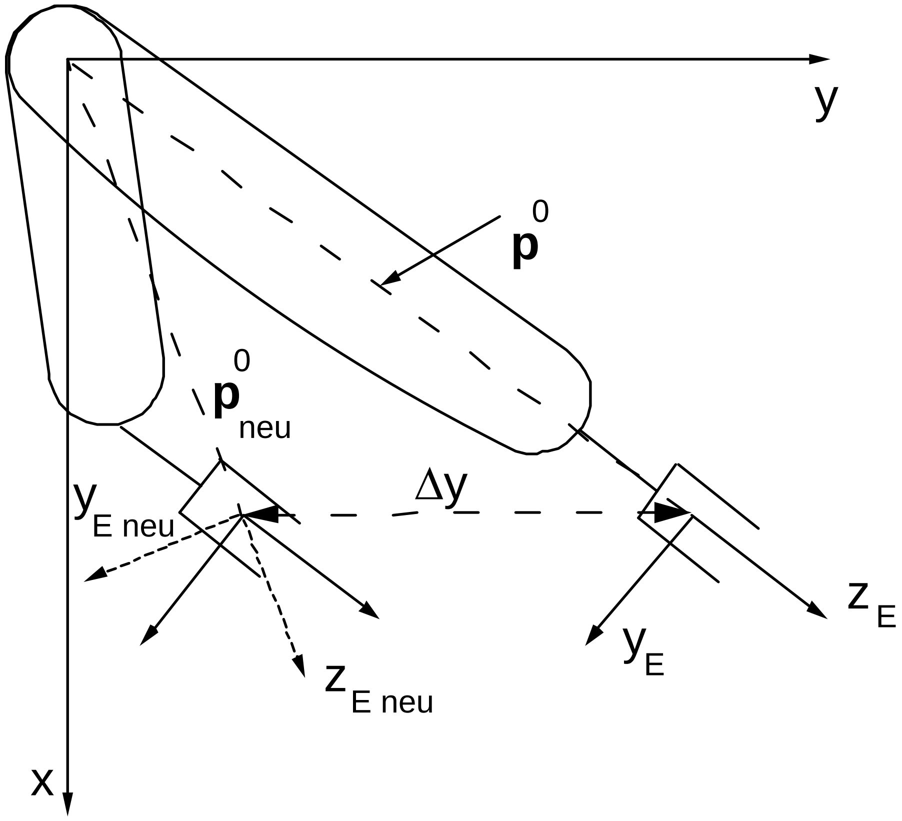


$\mathbf{p}^0$ ist Projektion des Positionsvektors in Ebene $z = 0$

$\mathbf{a}^0$ ist Projektion von z_E in Ebene $z = 0$

Beispielbewegung

- Bewegen um Δy
- Orientierung des Effektors bleibt gleich



Problem

- RM501 ist mit f=5 global degeneriert
- praktische Folge:
  Orientierung des Effektors nicht beliebig
- Orientierungsanpassung der Zielstellung erforderlich
- neue Orientierung muß mit verfügbaren Gelenken realisiert werden
- Degenerationsbedingung:

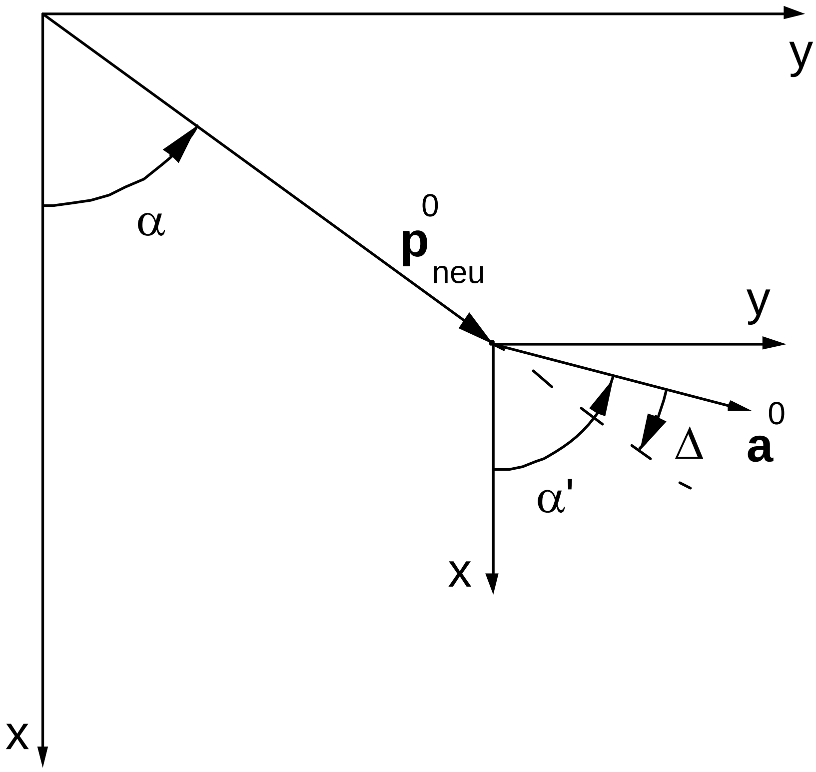
  $$a^0 = \begin{pmatrix} a_x \\ a_y \end{pmatrix} \quad \text{und} \quad p^0 = \begin{pmatrix} p_x \\ p_y \end{pmatrix}$$

  sind linear abhängig
- in homogener 4x4-Matrix:

  $$S(2,3) \cdot S(1,4) = S(1,3) \cdot S(2,4)$$

- Veränderung einzelner Elemente in homogener 4x4-Matrix verletzt Orthonormalitätsbedingung
- Folge: gesamter Orientierungsteil **RS** wird verändert

Verfahren



- es ist

  $\Delta = \alpha - \alpha'$ wegen $\alpha' + \Delta = \alpha$ mit

  $\alpha = \mathrm{ATAN2}\left(p_{y_{neu}}, p_{x_{neu}}\right)$ sowie $\alpha' = \mathrm{ATAN2}\left(a_y, a_x\right)$

- zusätzliche Drehung um Δ ist als Matrixgleichung

$$\mathbf{Rz}(\Delta)\cdot\mathbf{RS}=\mathbf{RS}'$$

und damit:

$$\mathbf{RS}'=\begin{bmatrix}\cos\Delta & -\sin\Delta & 0\\ \sin\Delta & \cos\Delta & 0\\ 0 & 0 & 1\end{bmatrix}\cdot\begin{bmatrix}n_x & o_x & a_x\\ n_y & o_y & a_y\\ n_z & o_z & a_z\end{bmatrix}$$

Vorgehen

- Ermittlung der homogenen 4x4-Matrix (Ergebnis der Vorwärtsrechnung)
- sechs von Δ abhängende Elemente aus alten Elementen $n_x \ldots a_z$ und sinΔ bzw. cosΔ neu bestimmen
- entstehende Matrix ist Ausgangspunkt für Rückwärtsrechnung

Hinweis: Benutzung von SIN, COS, … unter VADS Ada: GENERIC_ELEMENTARY_FUNCTIONS für FLOAT instantiieren

```
package ROBOTER_MATHE is

    PI              : constant FLOAT :=
    3.141592654;

    ATAN2_ERROR     : exception;

    WURZEL_ERROR    : exception;

    function BOGEN   ( GRAD              : in FLOAT )
                     return FLOAT;

    function GRAD    ( BOGEN             : in FLOAT )
                     return FLOAT;

    function  ISNULL ( WERT,
                     NULLUMGEBUNG        : in FLOAT )
                     return BOOLEAN;

    function  WURZEL ( WERT,
                     NULLUMGEBUNG        : in FLOAT )
                     return FLOAT;

    function  ATAN2  ( ZAEHLER,
                     NENNER              : in FLOAT;
                     NULLUMGEBUNG        : in FLOAT )
                     return FLOAT;

end ROBOTER_MATHE;
```

```
package RM501_KINEMATIK is

    FRAMEINVALID              : exception;

    type AXESVECTOR          is array (1..5) of FLOAT;

    type FRAME               is array (1..3, 1..4) of FLOAT;

    type T_RW_LOESUNGSTYP   is  (
        KEINE_LOESUNG,
        EINE_LOESUNG,
        ZWEI_LOESUNGEN,
        REDUKTIONSLOESUNG );

    type T_RW_LOESUNG   is record
        WINKELVEKTOR_1,
        WINKELVEKTOR_2   : AXESVECTOR;
        LSG_TYP          : T_RW_LOESUNGSTYP;
    end record;

    procedure VORWAERTSRECHNUNG (
        WINKELVEKTOR     : in AXESVECTOR;
        STELLUNG         : out FRAME);

    procedure RUECKWAERTSRECHNUNG (
        STELLUNG         : in FRAME;
        LOESUNG          : out T_RW_LOESUNG);

end RM501_KINEMATIK;
```