Computergraphik II

Winter 2011/2012

10 Mehrgliedrige Modelle (Skelette)

Versionsdatum: 6. Dezember 2011

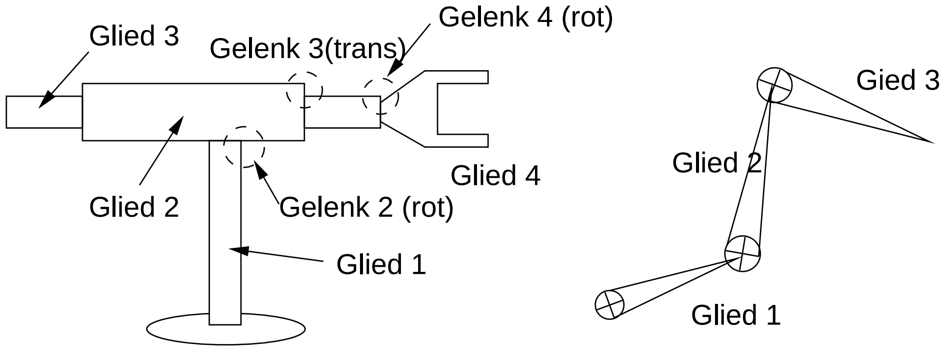
10 Mehrgliedrige Modelle (Skelette) ...

Ziel: Animation von Charakteren, d.h. zunächst von Skeletten

Bezeichnung:

Skelette: Hierarchie starrer Glieder (Links), die mit Gleit- oder Rotationsgelenken verbunden sind (Baumstruktur).

Endeffektor: Freie Enden des Skelettes (Blattknoten)



Beispiele: Robotik (Rotations- und Gleitgelenke) und Animation (nur Rotationsgelenke).

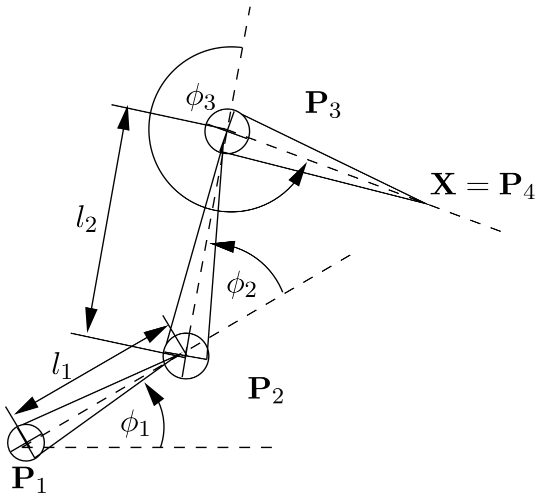
10.1 Mehrgliedrige Modelle in $2D$

Beschreibung von Skeletten ($2D$)

Ausgangslage:
1. n-gliedriges Skelett mit fixen Längen $l_1, \ldots, l_n$
2. Drehpunkt $\mathbf{P}_i$: Anfangspunkt von Glieds i
3. Endeffektor $\mathbf{X} = \mathbf{P}_{n+1}$: Endpunkt des Skeletts

Freiheitsgrade: $n + 2$ Stück:
- ❍ Drehpunkt $\mathbf{P}_1$ des ersten Glieds
- ❍ Winkel ϕ_i zum Vorgänger ($n > 1$) bzw. Orientierung des ersten Glieds ($n = 1$)

Zustandvektor: Beschreibt alle freien Winkel $\vec{\phi} = (\phi_1, \ldots, \phi_n)$



$2D$-Skelett ohne Verzweigung

10.1 Mehrgliedrige Modelle in $2D$ ...

Beschreibung von Skeletten ($2D$)

Sukzessive Ermittlung der $\mathbf{P}_{i+1}$ bei gegebenem $\mathbf{P}_1$:

$$i = 1: \qquad \mathbf{P}_2 = T_{1\to 2}(\mathbf{P}_1) = \mathbf{P}_1 + l_1 \begin{pmatrix} \cos\phi_1 \\ \sin\phi_1 \end{pmatrix}$$

$$i = 2: \qquad \mathbf{P}_3 = T_{2\to 3}(\mathbf{P}_2) = \mathbf{P}_2 + l_2 \begin{pmatrix} \cos(\phi_1 + \phi_2) \\ \sin(\phi_1 + \phi_2) \end{pmatrix}$$

$$\text{allgemein:} \qquad \mathbf{P}_{i+1} = T_{i\to i+1}(\mathbf{P}_i) = \mathbf{P}_i + l_i \begin{pmatrix} \cos(\phi_1 + \ldots + \phi_i) \\ \sin(\phi_1 + \ldots + \phi_i) \end{pmatrix}$$

Definition: **Arten von Kinematik**

Vorwärtskinematik: Variation der Freiheitsgrade zur Bewegung des Endeffektors $\mathbf{X}$

$$\mathbf{X} = f(\vec{\phi})$$

Inverse Kinematik: Direkte Bewegung des Endeffektors und Ermittlung der Freiheitsgrade:

$$\vec{\phi} = f^{-1}(\mathbf{X})$$

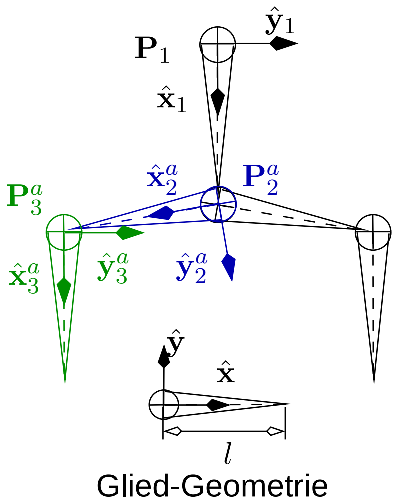
10.1.1 Hierarchische Animation

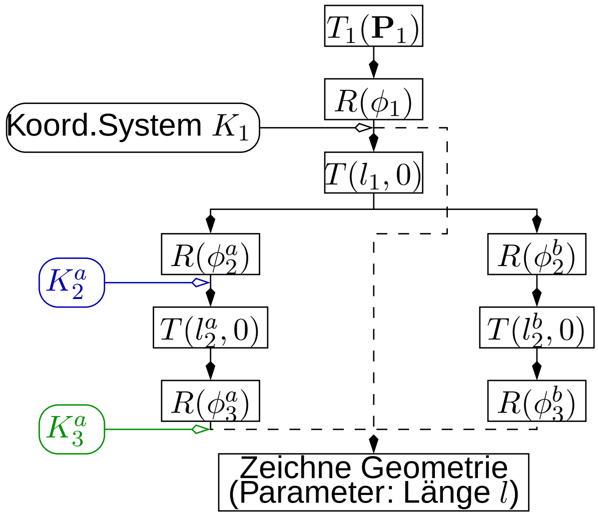
Hierarchische Koordinatensysteme

Ziel: Korrekte Ausrichtung von Objekten bzgl. eines Gliedes

Beobachtung: Skelett beschreibt Hierarchie von Koordinatensystemen
$K_i = \{\mathbf{P}_i, \hat{\mathbf{x}}_i, \hat{\mathbf{y}}_i\}$

Transformation $T_{i \to i+1} = T(l_i, 0) \cdot R(\phi_{i+1})$





10.1.1 Hierarchische Animation ...

Hierarchische Animation

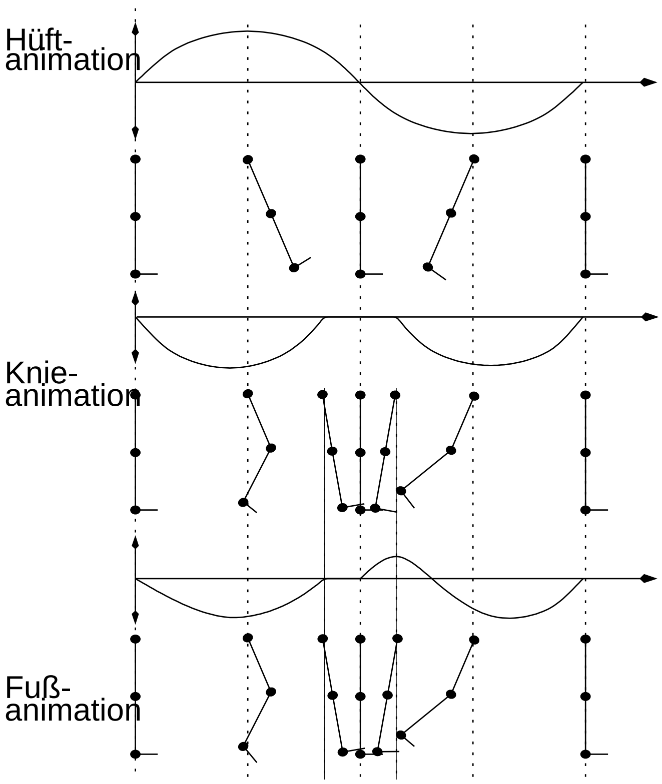
Hierarchische Modelle:
Hierarchische top-down Animation

Beispiel Bein:

1. Hüfte-Animation
2. Knie-Animation
3. Fuß-Animation

Probleme bei obigem Beispiel

- Vermeidung von horizontalem *sliding*
- Vermeidung von vertikalem *Eintauchen*



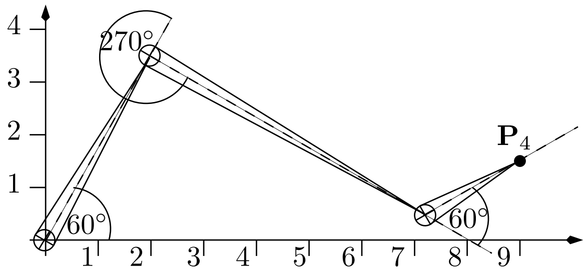
10.1.2 Vorwärtskinematik

Wesentliche Aspekte: ❍ Rotationen werden über Hierarchie vererbt

❍ Animation erfolgt über Veränderung der Winkel über die Zeit $\phi_i(t)$

❍ low-level-Form der Animation, die bei großen Strukturen schnell unübersichtlich wird

Beispiel mit 3 Glieder:



$$\phi_1 = 60^\circ \qquad \phi_2 = 270^\circ \qquad \phi_3 = 60^\circ$$
$$l_1 = 4 \qquad l_2 = 6 \qquad l_3 = 2$$
$$\mathbf{P}_1 = (0,0)$$

$$\mathbf{P}_2 = P_1 + l_1 \binom{\cos\phi_1}{\sin\phi_1} = \binom{0}{0} + 4\cdot\binom{\cos 60^\circ}{\sin 60^\circ} = \binom{2}{2\sqrt{3}}$$

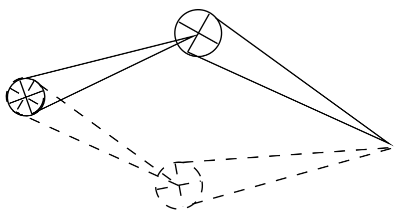
$$\mathbf{P}_3 = P_2 + l_2 \binom{\cos(\phi_1+\phi_2)}{\sin(\phi_1+\phi_2)} = \binom{2}{2\sqrt{3}} + 6\cdot\binom{\cos 330^\circ}{\sin 330^\circ} = \binom{2+3\sqrt{3}}{2\sqrt{3}-3}$$

$$\mathbf{P}_4 = P_3 + l_3 \binom{\cos(\phi_1+\phi_2+\phi_3)}{\sin(\phi_1+\phi_2+\phi_3)} = \binom{2+3\sqrt{3}}{2\sqrt{3}-3} + 2\cdot\binom{\cos 30^\circ}{\sin 30^\circ} = \binom{2+4\sqrt{3}}{2\sqrt{3}-2}$$

10.1.3 Inverse Kinematik

Probleme der Inverse Kinematik

1. f^{-1} i.a. nicht analytisch ermittelbar $\Rightarrow$ numerische Lösung der Inversen Kinematik
2. Inverse Kinematik i.a. nicht eindeutig lösbar:



3. Automatismus der inverse Kinematik erschwert Modellierung typischer Bewegungsformen (Hincken etc.)

Bezeichnung:

Unterbestimmtes System: „Anzahl Freiheitsgrade“ $>$ „Anzahl Bedingungen“

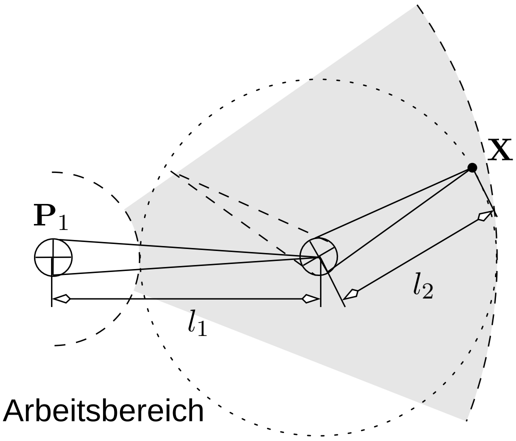
Erreichbarer Arbeitsbereich: Menge der durch Endeffektor erreichbaren Positionen

10.1.3 Inverse Kinematik ...

Beispiel: Analytische Lösung der Inversen Kinematik

Gegeben: Zweigliedriges Modell

- Positionen $\mathbf{P}_1 = \binom{x_1}{y_1}$
- Endeffektor $\mathbf{X}\binom{x}{y}$
- Längen l_1, l_2



Erreichbarer Arbeitsbereich:

$$\{\mathbf{Q} \;:\; \|\mathbf{Q} - \mathbf{P}_1\| \in [|l_1 - l_2|\,, l_1 + l_2]\}$$

Analytische Lösung der inversen Kinematik:

$$\phi_2 = \cos^{-1}\left(\frac{(x - x_1)^2 + (y - y_1)^2 - l_1^2 - l_2^2}{2 l_1 l_2}\right)$$

$$\phi_1 = \tan^{-1}\left(\frac{(y - y_1)(l_1 + l_2\cos\phi_2) - (x - x_1) l_2 \sin\phi_2}{(x - x_1)(l_1 + l_2\cos\phi_2) + (y - y_1) l_2 \sin\phi_2}\right)$$

10.1.3 Inverse Kinematik ...

Herleitung der analytischen Lösung

Mit $\Delta x = x - x_1$, $\Delta y = y - y_1$, $s_i = \sin\phi_i$, $c_i = \cos\phi_i$, $s_{1+2} = \sin(\phi_1 + \phi_2)$, $c_{1+2} = \cos(\phi_1 + \phi_2)$

Aus der Vorwärtsdarstellung folgt

$$\begin{aligned}
\binom{\Delta x}{\Delta y} &= \binom{l_1 c_1 + l_2 c_{1+2}}{l_1 s_1 + l_2 s_{1+2}} \\
\Rightarrow \Delta x^2 + \Delta y^2 &= l_1^2 \underbrace{(c_1^2 + s_1^2)}_{=1} + l_2^2 \underbrace{(c_{1+2}^2 + s_{1+2}^2)}_{=1} + 2 l_1 l_2 (\underbrace{c_1 c_{1+2} + s_1 s_{1+2}}_{=\cos(\phi_1+\phi_2-\phi_1)=\cos\phi_2}) \\
&= l_1^2 + l_2^2 + 2 l_1 l_2 \cos\phi_2 \Rightarrow \phi_2 = \arccos\left(\frac{\Delta x^2 + \Delta y^2 - l_1^2 - l_2^2}{2 l_1 l_2}\right)
\end{aligned}$$

und für ϕ_1 mit $t_i = \tan(\phi_i)$:

$$\begin{aligned}
\binom{\Delta x}{\Delta y} &= \binom{l_1 c_1 + l_2(c_1 c_2 - s_1 s_2)}{l_1 s_1 + l_2(s_1 c_2 + c_1 s_2)} \\
\Rightarrow \Delta x\,(l_1 s_1 + l_2(s_1 c_2 + c_1 s_2)) &= \Delta y\,(l_1 c_1 + l_2(c_1 c_2 - s_1 s_2)) \quad |:\cos(\phi_1) \\
\Rightarrow \Delta x\,(l_1 t_1 + l_2(t_1 c_2 + s_2)) &= \Delta y\,(l_1 + l_2(c_2 - t_1 s_2)) \\
\Leftrightarrow t_1(\Delta x(l_1 + l_2 c_2) + \Delta y l_2 s_2) &= (\Delta y(l_1 + l_2 c_2) - \Delta x l_2 s_2)
\end{aligned}$$

10.1.3 Inverse Kinematik ...

Beispiel: Zahlenbeispiel zur expliziten Lösung

Gegeben: $\mathbf{P}_1 = (0,0),\ \mathbf{X} = (10,0),\ l_1 = l_2 = 10$

Berechnung von ϕ_2:

$$\phi_2 = \cos^{-1}\left(\frac{(x-x_1)^2+(y-y_1)^2-l_1^2-l_2^2}{2l_1l_2}\right) = \cos^{-1}\left(\frac{-10^2}{2\cdot 10^2}\right) \in \{-120°, 120°\}$$

Berechnung von ϕ_1:

Allg.: $$\phi_1 = \tan^{-1}\left(\frac{(y-y_1)(l_1+l_2\cos\phi_2)-(x-x_1)l_2\sin\phi_2}{(x-x_1)(l_1+l_2\cos\phi_2)+(y-y_1)l_2\sin\phi_2}\right)$$

Für $\phi_2 = 120°$:

$$\phi_1 = \tan^{-1}\left(\frac{0-10\cdot 10\sin(120°)}{10(10+10\cos(120°))+0}\right) = \tan^{-1}(-\sqrt{3}) \Rightarrow \phi_1 \in \{-60°, 120°\}$$

Für $\phi_2 = -120°$:

$$\phi_1 = \tan^{-1}\left(\frac{0-10\cdot 10\sin(-120°)}{10(10+10\cos(-120°))+0}\right) = \tan^{-1}(\sqrt{3}) \Rightarrow \phi_1 \in \{-120°, 60°\}$$

10.1.4 Numerische Lösung der Inversen Kinematik

Probleme: Im allgemeinen

1. kann IK-Lösung nicht analytisch bestimmt werden $\rightarrow$ Numerik
2. gibt es keine, eine oder unendliche viele Lösungen

Ausgangslage: $\vec{\phi}^{alt}, \mathbf{X}^{alt}$ als VK-Lösung, also $f(\vec{\phi}^{alt}) = \mathbf{X}^{alt}$, zudem gegeben ist $\mathbf{X}^{neu}$

Gesucht: $\vec{\phi}^{neu}$, so dass $f(\vec{\phi}^{neu}) = \mathbf{X}^{neu}$, also Nullst. von $g(\vec{\phi}) := f(\vec{\phi}) - \mathbf{X}^{neu}$

Erinnerung: Eindimensionales Newton-Verfahren

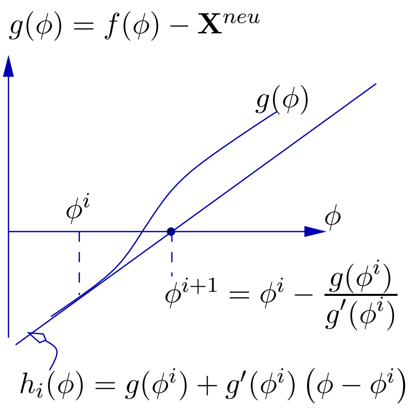
Initial: $\phi^0 = \phi^{alt}$

Iteration: ϕ^{i+1} = Nullst. der Tangente an g in ϕ^i

Tangente: $h_i(\phi) = g(\phi^i) + g'(\phi^i)(\phi - \phi^i)$

Nullstelle: $\phi^{i+1} = \phi^i - \frac{g(\phi^i)}{g'(\phi^i)} = \phi^i - \frac{f(\phi^i)-\mathbf{X}^{neu}}{f'(\phi^i)}$

Ableitung: $f'(\phi^i) \approx \dfrac{f(\phi^i+\Delta)-f(\phi^i)}{\Delta}$



Allgemeine Inverse Kinematik

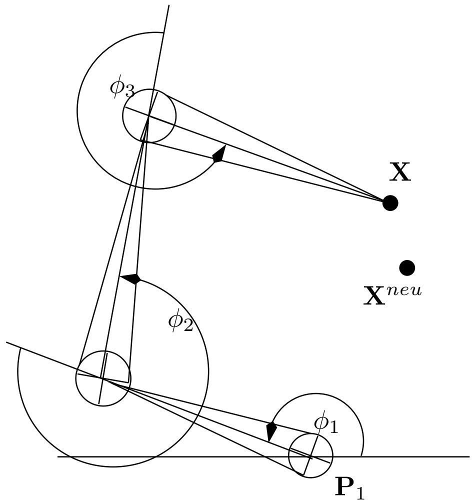
In der Animation ein iteratives Problem:

Bekannt: $\vec{\phi}^{alt}, \mathbf{X}^{alt} = f(\vec{\phi}^{alt}), \mathbf{X}^{neu}$, sowie fixer Startpunkt $\mathbf{P}_1$

Gesucht: $\vec{\phi}^{neu}$ mit $\mathbf{X}^{neu} = f(\vec{\phi}^{neu})$

$\Leftrightarrow \quad g(\vec{\phi}^{neu}) := f(\vec{\phi}^{neu}) - \mathbf{X}^{neu} = 0$ mit

$$\left\| \vec{\phi}^{neu} - \vec{\phi}^{alt} \right\| = \min\{ \left\| \vec{\phi} - \vec{\phi}^{alt} \right\| \mid g(\vec{\phi}) = 0 \}$$



Beachte: f und g sind Funktionen mit n Variablen und m $(m = 2, 3)$ Komponenten, wobei meist $n \geq m$

$$f : \mathbb{R}^n \mapsto \mathbb{R}^m, \quad f(\vec{\phi}) = f(\phi_1, \ldots, \phi_n) = \begin{pmatrix} f_1(\phi_1, \ldots, \phi_n) \\ \vdots \\ f_m(\phi_1, \ldots, \phi_n) \end{pmatrix}$$

Mehrdimensionales Newton-Verfahren (Fall $n = m > 1$)

Vorgehen analog wie im eindimensionalen Fall

Anpassungen: $\vec{\phi}, \mathbf{X}$ ist mehrdimensionaler Vektor bzw. Punkt

Funktion: $f : \mathbb{R}^n \mapsto \mathbb{R}^n$

Ableitung: D_f *Jakobi-Matrix* ersetzt f' : $D_f(\vec{\phi}) = \begin{pmatrix} \frac{\partial f_1}{\partial \phi_1} & \cdots & \frac{\partial f_1}{\partial \phi_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial \phi_1} & \cdots & \frac{\partial f_n}{\partial \phi_n} \end{pmatrix} \in \mathbb{R}^{n \times n}$

Newton-Iteration: $\vec{\phi}^{i+1}$ ist Nullstelle der linearen Funktion an g in $\vec{\phi}^i$

Lin. Funkt.: $h_i(\vec{\phi}) = g(\vec{\phi}^i) + D_g(\vec{\phi}^i)(\vec{\phi} - \vec{\phi}^i), \quad D_g = D_f$

Nullstelle: $\vec{\phi}^{i+1} = \vec{\phi}^i - \left(D_g(\vec{\phi}^i)\right)^{-1} g(\vec{\phi}^i) = \vec{\phi}^i - \left(D_f(\vec{\phi}^i)\right)^{-1} \left(f(\vec{\phi}^i) - \mathbf{X}^{neu}\right)$

falls $\left(D_f(\vec{\phi}^i)\right)^{-1}$ existiert

10.1.4 Numerische Lösung der Inversen Kinematik ...

Mehrdimensionales Newton-Verfahren (Fall $n > m$)

Problem: Jakobi-Matrix nicht mehr quadratisch, d.h.

$$g(\phi^i) + D_g(\vec{\phi}^i)(\vec{\phi} - \vec{\phi}^i) = 0 \Leftrightarrow D_g(\vec{\phi}^i)(\vec{\phi} - \vec{\phi}^i) = -g(\phi^i) \qquad (1)$$

hat $0, 1$ oder ∞ viele Lösungen.

Pseudo-Inverse: Hat D_f vollen Rang, dann kann die *Pseudo-Inverse* bestimmt werden:

$$(D_f)^+ = D_f^T \cdot (D_f D_f^T)^{-1}, \qquad D_f D_f^T \in \mathbb{R}^{m\times m} \text{ invertierbar}$$

Die Pseudo-Inverse liefert *eine Lösung* von Gleichung (1):

$$\vec{\phi}^{i+1} - \vec{\phi}^i = -(D_f)^+ g(\phi^i) \text{ wobei } \left\| (\vec{\phi}^{i+1} - \vec{\phi}^i) \right\| \text{ minimal unter allen Lösungen}$$

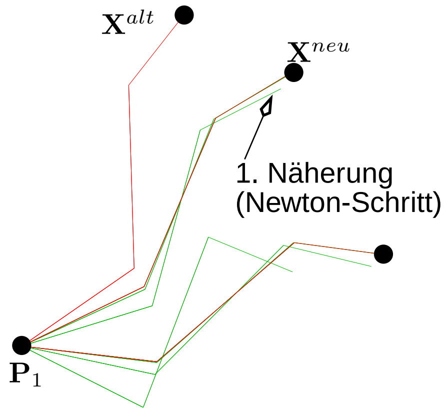
Interpretation: $\left\| (\vec{\phi}^{i+1} - \vec{\phi}^i) \right\|$ ergibt minimale Abweichung der neuen Winkel zu den alten Winkeln

10.1.4 Numerische Lösung der Inversen Kinematik ...

Beispiel für Inverse Kinematik

Initial: $\mathbf{P}_1 = (50, 400),\ l_1 = 150,\ l_2 = 200,\ l_3 = 100,$
$\vec{\phi}^{alt} = \vec{\phi}_0 = (0.6, 1.0, -0.7),\ \mathbf{X}^{alt} = (230, 763), \mathbf{X}^{neu} = (350, 700)$

1. Newton-Schritt:



$$\frac{\partial f}{\partial \phi_1} = \begin{pmatrix} -363 \\ 180 \end{pmatrix}, \frac{\partial f}{\partial \phi_2} = \begin{pmatrix} -278 \\ 56 \end{pmatrix}, \frac{\partial f}{\partial \phi_3} = \begin{pmatrix} -78 \\ 62 \end{pmatrix}$$

$$\Rightarrow D_f = \begin{pmatrix} -363 & -278 & -78 \\ 180 & 56 & 62 \end{pmatrix}$$

und $D_f^+ = \begin{pmatrix} 0.001025 & 0.006793 \\ -0.005497 & -0.010536 \\ 0.002010 & 0.00595 \end{pmatrix}$

$$\vec{\phi}^1 - \vec{\phi}^0 = -D_f^+ g(\vec{\phi}^0) = -D_f^+ (\mathbf{X}^{alt} - \mathbf{X}^{neu})$$

$$\Rightarrow \vec{\phi}^1 = \begin{pmatrix} 0.6 \\ 1.0 \\ -0.7 \end{pmatrix} - \begin{pmatrix} 0.001025 & 0.006793 \\ -0.005497 & -0.010536 \\ 0.002010 & 0.00595 \end{pmatrix} \begin{pmatrix} -120 \\ 63 \end{pmatrix} = \begin{pmatrix} 0.295 \\ 1. \\ -0.833 \end{pmatrix}, \ f(\phi^1) = \begin{pmatrix} 336.44 \\ 681.27 \end{pmatrix}$$

10.1.4 Numerische Lösung der Inversen Kinematik ...

Alternative kinematische Ansätze

Nachteile von IK überwiegen für highend-Animationen meist deren Vorteil.

Motion-Capturing: Bewegungspfad durch Bewegungserfassung von Charakteren

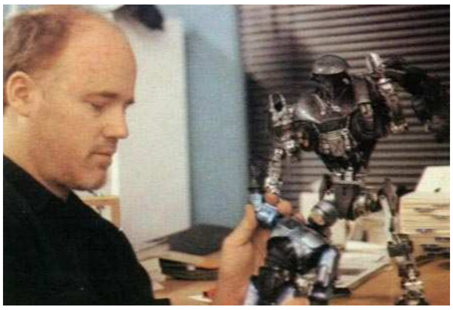
Stop-Motion-Animation: ❍ Klassisch: Frameweise Bewegungserfassung

❍ Keyframe-weise: Abgreifen der Gelenkwinkel und Keyframe-Animation

Nachbearbeitung: Glätten, Überlagern etc. von Bewegungspfaden

Paralleler Einsatz von VK und IK

Motion-Capturing



Klassische Stop-Motion-Animation

10.2 Mehrgliedrige Modelle in $3D$

Hierarchische Koordinatensysteme in 3D

Ansatz: Beschreibung der Glied-Hierarchie wie in 2D

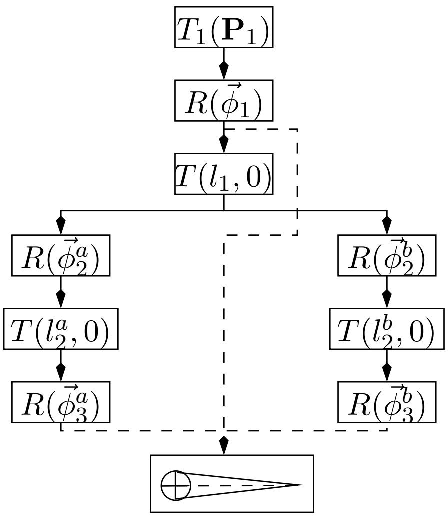
Transformation $T_{i \to i+1} = T(l_i, 0) \cdot R(\vec{\phi}_{i+1})$
wobei $R(\vec{\phi}_{i+1}) = R(\phi^x_{i+1}, \phi^y_{i+1}, \phi^z_{i+1})$ (Rotationsmatrizen mit Eulerwinkeln)

Vorwärts-Kinematik kann direkt mit diesem Ansatz verwendet werden

Inverse Kinematik erfordert Optimierung der Rotationsparameter:

❍ Vermeide Mehrdeutigkeiten durch Quaternionen

❍ Problem: Optimierung von Quaternionen erzwingt Einhaltung der Randbedingung $||\underline{\mathbf{q}}|| = 1$



10.2 Mehrgliedrige Modelle in $3D$ ...

<u>**Ansatz:**</u> **Denavit-Hardenberg**

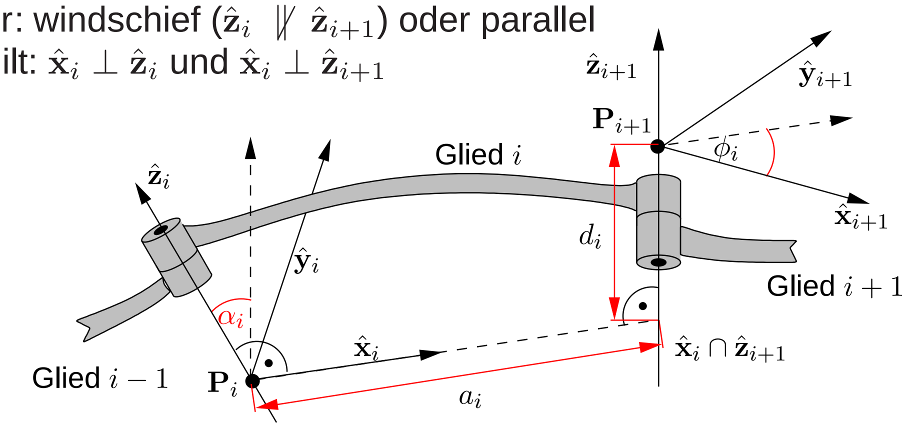
Alternativer Ansatz mit einfachen Winkelparametern

Ausgangslage: ❍ n-gliedriges Skelett ohne Verzweigung

❍ jedes Glied hat eine Rotationsachse

Koordinatensysteme für i-tes Glied $K_i = \{\mathbf{P}_i, \hat{\mathbf{x}}_i, \hat{\mathbf{y}}_i, \hat{\mathbf{z}}_i\}$, wobei

- ❍ $\hat{\mathbf{z}}_i$ entspricht der Drehachse
- ❍ $\hat{\mathbf{x}}_i$ und $\mathbf{P}_i$ durch kürzeste Verbindung zwischen $\hat{\mathbf{z}}_i, \hat{\mathbf{z}}_{i+1}$ festgelegt:
  - ❑ existiert immer: windschief ($\hat{\mathbf{z}}_i \not\parallel \hat{\mathbf{z}}_{i+1}$) oder parallel
  - ❑ automatisch gilt: $\hat{\mathbf{x}}_i \perp \hat{\mathbf{z}}_i$ und $\hat{\mathbf{x}}_i \perp \hat{\mathbf{z}}_{i+1}$



10.2 Mehrgliedrige Modelle in $3D$ ...

<u>**Denavit-Hardenberg (Forts.)**</u>

Kontrollparameter:

a_i Abstand zwischen Achsen $\hat{\mathbf{z}}_i$ und $\hat{\mathbf{z}}_{i+1}$

α_i Winkel zwischen $\hat{\mathbf{z}}_i$ und $\hat{\mathbf{z}}_{i+1}$ (beide z-Achsen $\perp \hat{\mathbf{x}}_i$)

d_i Abstand des Schnittpunktes $\hat{\mathbf{x}}_i \cap \hat{\mathbf{z}}_{i+1}$ zum Ursprung $\mathbf{P}_{i+1}$ von K_{i+1}

ϕ_i Winkel zwischen $\hat{\mathbf{x}}_i, \hat{\mathbf{x}}_{i+1}$ (beide $\perp \hat{\mathbf{z}}_{i+1}$)

Mit Startpunkt $\mathbf{P}_1$: $4 \times n + 3$ Freiheitsgrade

Transformation von K_i nach K_{i+1} in *lokalen Koordinaten*:

$R(x, \alpha_i)$**:** Bilde $\hat{\mathbf{z}}_i$ auf $\hat{\mathbf{z}}_{i+1}$ ab (aktuelle x-Achse ist $\hat{\mathbf{x}}_i$).

$T(a_i, 0, d_i)$**:** Verschiebe $\mathbf{P}_i$ nach $\mathbf{P}_{i+1}$

$R(z, \phi_i)$**:** Bilde $\hat{\mathbf{x}}_i$ auf $\hat{\mathbf{x}}_{i+1}$ ab.

Kombiniert:

$$\mathbf{Q}^{i+1} = T_{i \to i+1}(\mathbf{Q}^i), \quad T_{i \to i+1} = R(x, \alpha_i) \circ T(a_i, 0, d_i) \circ R(z, \phi_i)$$

10.3 Soft-Body Animation

Ziel: Animation von Körpern inkl. Muskeln und Haut auf Skelett-Basis

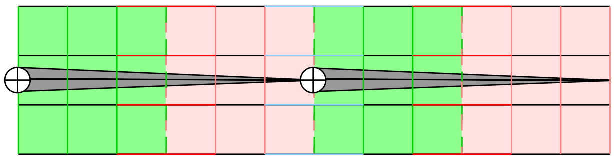
Einfaches Layer-Modell: Kontrolle des Skeletts mit *Anbindung* an Haut

- ***Skelett-Ebene:*** Unterste Ebene; sie entspricht dem mehrgliedrigen Modell
- ***Haut-Ebene:*** Geometrie aus NURBS-Fläche oder anderer Geometrie-Repräsentation
- ***Muskel-Ebene:*** Verbindet Skelett und Haut über FFD's

<u>Ansatz:</u> Chadwick '89 (Spezialfall: Arm oder Bein)

Ausgangslage: Zwei Glieder, mit Null-Rotation

Definition der FFD-Volumen: Jedes Glied wird mit zwei tri-kubischen FFD Bézier-Volumen versehen.



Ausgangslage

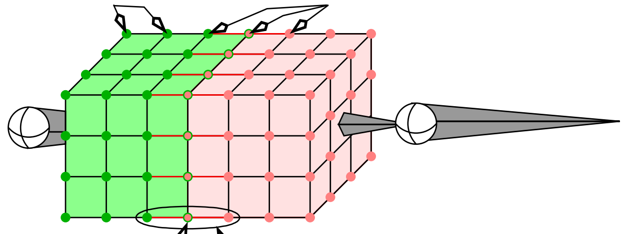
10.3 Soft-Body Animation ...

Chadwick (Forts.)

Randbedingungen für C^1-Übergänge *zwischen* FFD-Volumen:

1. KPs der Randebenen sind identisch
2. KPs der benachbarten Ebenen liegen kollinear (Teilungsverhältnis 1:1)

Sieben Kontrollpunktebenen pro Glied, *drei innere Ebenen*, je zwei *Verbindungs-Ebenen*



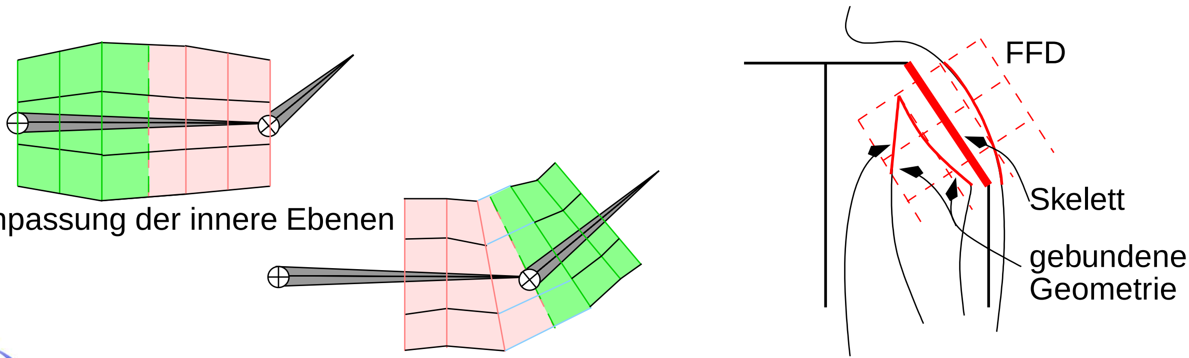
Animation des Layer-Modells

Beugung des Gelenkes: Unter Einhaltung der C^1-Übergänge

1. Skalierung der zwei freien inneren Ebenen: Muskelkontrolle
2. Rotation der zwei freien Verbindungs-Ebenen: Angleichung zwischen Gliedern

Binding: Geoemtrie wird entsprechend ihrer Lage im FFD-Volumen deformiert

Problem: Zuordnung nicht immer eindeutig $\rightarrow$ interaktive Zuordnung Geometrie zu Skelett



Anpassung der innere Ebenen

Anpassung der Verbindungsebenen

Binding-Problem