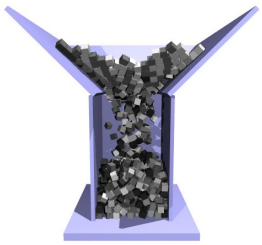
Virtuelle Realität
Rigid Body Simulation



G. Zachmann
Clausthal University, Germany
cg.in.tu-clausthal.de

Der Zustandsvektor

- Zustand eines Objektes ist eindeutig definiert durch:
  - Aktuelle Position/Orientierung
  - Aktuelle Geschwindigkeiten
- Definiere also den Zustandsvektor

$$\mathbf{s}(t) = \begin{pmatrix} \mathbf{x}(t) \\ R(t) \\ \mathbf{p}(t) \\ \mathbf{l}(t) \end{pmatrix}$$

x = Ort im Raum,
R = Rotationsmatrix, ausgehend vom Objektkoordinatensystem,
p = linearer Impuls (entspricht linearer Geschwindigkeit),
l = angular momentum (entspricht Winkelgeschwindigkeit)

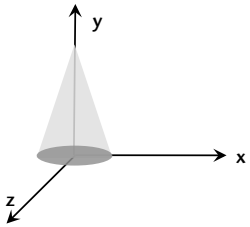
- Überblick über die Simulation des Objektes im freien Flug:

```
init state s(t0)
loop:
  berechne dS := d/dt S(t_i)
  setze    S+ = Δt·dS          {explicit Euler}
  render objects
```

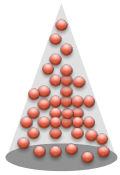
- Aufgabe im Folgenden: $\frac{d}{dt}S$ berechnen

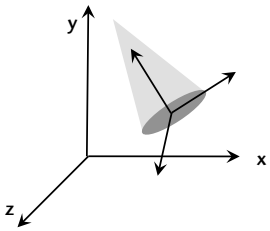
Position, Orientierung, Koordinatensysteme

- Es gibt 2 Koordinatensysteme:
  - das lokale (= Objektkoordinaten)
  - das globale (= Weltkoordinaten)



- Hilfsvorstellung: das Objekt ist aus sehr vielen Partikeln zusammengesetzt, die alle (auf „magische“ Weise) starr miteinander verbunden sind





- Bezeichnungen für einige Positionen:
  $\bar{\mathbf{x}}_i$ = Partikel i (Punkt) des Objektes in Objektkoordinaten
  $\mathbf{x}_i(t)$ = Partikel i (Punkt) des Objektes in Weltkoordinaten
  m_i = Masse des Partikels i
- Wahl des Objektkoordinatensystems: wähle dieses so, dass
  Ursprung = Masseschwerpunkt
  d.h. so, dass

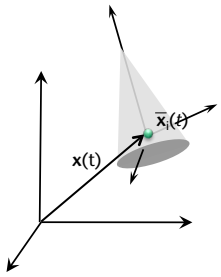
$$\sum_i m_i \bar{\mathbf{x}}_i = 0$$

- Der Masseschwerpunkt in Weltkoordinaten ist

$$\mathbf{x}(t) = \frac{1}{m} \sum_i m_i \mathbf{x}_i(t), \quad \text{mit} \quad m = \sum_i m_i$$

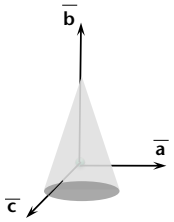
- Die Position eines Punktes zur Zeit t:

$$\mathbf{x}(t) = R(t)\bar{\mathbf{x}}_i + \mathbf{x}(t)$$



- Die geometrische Interpretation der Matrix R(t):
  - In lokalen Koordinaten ist

$$\bar{\mathbf{a}} = \mathbf{x} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$



  - In Weltkoordinaten ist

$$\mathbf{a}(t) = R(t)\bar{\mathbf{a}} = R(t)\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = \text{erste Spalte von } R(t)$$

- Analog für $\mathbf{b}(t)$ und $\mathbf{c}(t)$
- Fazit:

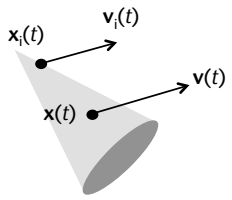
$$R(t) = \begin{pmatrix} \vdots & \vdots & \vdots \\ \mathbf{a}(t) & \mathbf{b}(t) & \mathbf{c}(t) \\ \vdots & \vdots & \vdots \end{pmatrix}$$

Die Geschwindigkeiten

- Die lineare Geschwindigkeit:
  - Annahme: das Objekt rotiert nicht und bewegt sich mit konstanter (linearer) Geschwindigkeit durch den Raum
  - Beobachtung: für die Geschwindigkeit des Schwerpunktes gilt

$$\dot{\mathbf{x}}(t) = \dot{\mathbf{x}}_i(t) =: \mathbf{v}_i(t) = \mathbf{v}(t)$$

Alle Punkte haben dieselbe Geschwindigkeit



- Die Winkelgeschwindigkeit (*angular velocity*):
  - Annahme: der Schwerpunkt bewegt sich nicht, das Objekt rotiert um Schwerpunkt
  - Zum Zeitpunkt t gibt es eine bestimmte Achse, um die es rotiert:

    $\omega(t)$ = Rotationsachse zum Zeitpunkt t

    $\|\omega(t)\|$ = Rotationsgeschwindigkeit, Einheit ist $\frac{\text{Umdrehungen}}{\text{sec}}$

- Frage: Wie hängen R(t) und $\omega(t)$ zusammen?
  - Zur Erinnerung: bei der linearen Geschwindigkeit gilt

$$\mathbf{v}(t) = \frac{d}{dt}\mathbf{x}(t)$$

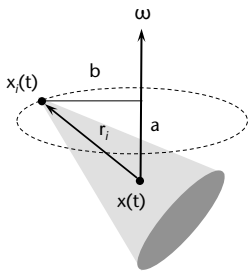
Die Winkelgeschwindigkeit

- Annahme: das Objekt rotiert um Achse ω und den Schwerpunkt mit Rotationsgeschwindigkeit $\|\omega\|$; der Schwerpunkt bleibt fest
- Beobachtung: alle Punkte $\mathbf{x}_i$ rotieren mit derselben Rotationsgeschwindigkeit ω
  - Achtung: die Ableitung $\dot{\mathbf{x}}_i$ ist i.A. verschieden!
- Setze

$$\mathbf{r}_i(t) = \mathbf{x}_i(t) - \mathbf{x}(t)$$

- Zerlege $\mathbf{r}_i$ in $\mathbf{r}_i = \mathbf{a} + \mathbf{b}$

  wobei $\mathbf{a} \| \omega$ und $\mathbf{b} \perp \omega$



- $\mathbf{x}_i$ beschreibt einen Kreis um die Achse ω und senkrecht zu ω

⇒ **a** bleibt fest, **b** rotiert im Kreis

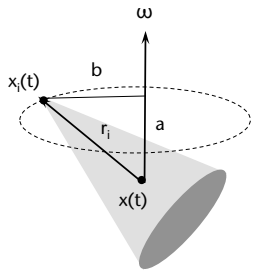
⇒ $\dot{\mathbf{r}}_i(t)$ muss senkrecht zu **b** und zu ω sein (1)

⇒ der Betrag

$$\|\dot{\mathbf{r}}_i(t)\| = \|\mathbf{b}\| \cdot \|\omega(t)\| \quad (2)$$

- Beobachtung: der Vektor (ω x **b**) erfüllt genau die Bedingungen (1) und (2)

⇒ $\dot{\mathbf{r}}_i(t) = \omega(t) \times \mathbf{b}$



- Außerdem ist

$$\omega(t) \times \mathbf{a} = 0$$

- Also können wir schreiben

$$\dot{\mathbf{r}}(t) = \omega(t) \times \mathbf{b} = \omega(t) \times \mathbf{b} + \omega(t) \times \mathbf{a}$$

- Fazit:

$$\dot{\mathbf{r}}_i(t) = \omega(t) \times \mathbf{r}(t)$$

- Schreibe die Rotationsmatrix R(t) als

$$R(t) = \begin{pmatrix} \vdots & \vdots & \vdots \\ \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{r}_3 \\ \vdots & \vdots & \vdots \end{pmatrix}$$

- Annahme: $\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3$ sind Punkte im Objekt
- Damit ist

$$\dot{\mathbf{R}}(t) = \begin{pmatrix} | & | & | \\ \dot{\mathbf{r}}_1 & \dot{\mathbf{r}}_2 & \dot{\mathbf{r}}_3 \\ | & | & | \end{pmatrix} = \begin{pmatrix} | & | & | \\ \omega(t) \times \mathbf{r}_1(t) & \omega(t) \times \mathbf{r}_2(t) & \omega(t) \times \mathbf{r}_3(t) \\ | & | & | \end{pmatrix}$$

Interludium: das Kreuzprodukt

- Das Kreuzprodukt

$$\mathbf{a} \times \mathbf{b}$$

kann man darstellen als

$$a^{\times} \cdot \mathbf{b}$$

wobei

$$a^{\times} = \begin{pmatrix} 0 & -a_2 & a_y \\ a_2 & 0 & -a_x \\ -a_y & a_x & 0 \end{pmatrix}$$

- Diese Matrix ist eine schief-symmetrische (*skew-symmetric*) Matrix, d.h.

$$(a^{\times})^{\mathsf{T}} = -a^{\times}$$

- Diese Darstellung hat viele Vorteile, u.a.:
  - $a^T \cdot (b \times c) \neq (a^T \cdot b) \times c$

    aber

    $a^T \cdot (b^\times \cdot c) = (a^T \cdot b^\times) \cdot c$
  - $a \times (b \times c) \neq (a \times b) \times c$

    aber

    $a^\times (b^\times c) = (a^\times b^\times) c$
  - Fazit: bei der Schreibweise mit schief-symmetrischer Matrix $a^\times$ gilt die Assoziativität!

Zurück zur Winkelgeschwindigkeit

- Mit der $a^\times$-Darstellung können wir die Ableitung der Rotation nun so schreiben:

$$\dot{R}(t) = \begin{pmatrix} \vdots & \vdots & \vdots \\ \omega^\times \cdot \mathbf{r}_1(t) & \omega^\times \cdot \mathbf{r}_2(t) & \omega^\times \cdot \mathbf{r}_3(t) \\ \vdots & \vdots & \vdots \end{pmatrix}$$

$$= \omega(t)^\times \begin{pmatrix} \mathbf{r}_1(t) & \mathbf{r}_2(t) & \mathbf{r}_3(t) \end{pmatrix}$$

$$= \omega(t)^\times \cdot R(t)$$

- Bemerkung: hier kommt also die ursprüngliche Orientierung mit ins Spiel!

Die Geschwindigkeit eines Punktes

- Erinnerung: der Ort eines Punktes des Objektes ist

$$\mathbf{x}_i(t) = R(t)\bar{\mathbf{x}}_i + \mathbf{x}(t)$$

- Vereinfachung: in Zukunft lassen wir den Zeitparameter t (meist) weg
- Die Geschwindigkeit dieses Punktes ist

$$\dot{\mathbf{x}}_i = \dot{R}\,\bar{\mathbf{x}}_i + \dot{\mathbf{x}} = \omega^{\times} R\,\bar{\mathbf{x}}_i + \mathbf{v}$$

- Anders schreiben:

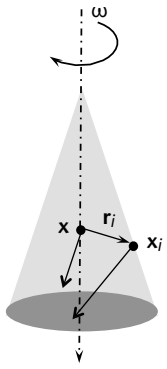
$$\dot{\mathbf{x}}_i = \omega^{\times}(R\,\bar{\mathbf{x}}_i + \mathbf{x} - \mathbf{x}) + \mathbf{v} = \omega^{\times}(\mathbf{x}_i - \mathbf{x}) + \mathbf{v}$$

- Bezeichne ab jetzt immer

$$\mathbf{r}_i = \mathbf{x}_i - \mathbf{x}$$

- Damit ist

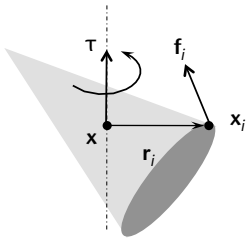
$$\dot{\mathbf{x}}_i = \omega^{\times}\mathbf{r}_i + \mathbf{v}$$



Lineare Kraft und Drehmoment

- Sei $\mathbf{f}_i(t)$ = Kraft auf ein Partikel i des Objektes
- Drehmoment (*torque*) = Rotationskraft, die durch $\mathbf{f}_i$ an Punkt i auf den Schwerpunkt ausgeübt wird:

$$\tau_i = \mathbf{r}_i \times \mathbf{f}_i$$

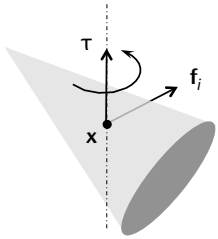


- Bemerkung:
  - Das Drehmoment ist die „Drehkraft“, die durch den Hebelarm $\mathbf{r}_i$ auf den Schwerpunkt ausgeübt wird
  - Das Drehmoment hängt von der Position ab, wo die Tangentialkraft angreift! (im Gegensatz zur linearen Kraft im Schwerpunkt)

- Gesamtkraft auf das Objekt:

$$\mathbf{f}(t) = \sum \mathbf{f}_i(t)$$

$$\tau(t) = \sum \tau_i(t) = \sum \mathbf{r}_i \times \mathbf{f}_i$$



Der lineare Impuls (*linear momentum*)

- Der lineare Impuls eines Partikels ist definiert als

$$\mathbf{p}_i = m_i \mathbf{v}_i$$

- Der gesamte lineare Impuls auf das Objekt ist

$$\begin{aligned}\mathbf{p} &= \sum m_i \dot{\mathbf{x}}_i \\ &= \sum m_i (\omega^x \mathbf{r}_i + \mathbf{v}) \\ &= \omega^x \cdot \sum m_i \mathbf{r}_i + \mathbf{v} \cdot \sum m_i \\ &= m \cdot \mathbf{v}\end{aligned}$$

- Wie verändert sich der lineare Impuls über die Zeit?

$$\dot{\mathbf{p}}(t) = m \cdot \dot{\mathbf{v}}(t) = m \cdot \mathbf{a}(t) = \mathbf{f}(t)$$

- Bemerkung:
  In vielen Gleichungen ist es praktischer, nicht die reine Geschwindigkeit, sondern den linearen Impuls als eigenständige Größe zu verwenden

Der Drehimpuls (*angular momentum*)

- Ist ein sehr unintuitives physikalisches Konzept, erlaubt es aber, viele Gleichungen einfacher zu schreiben, als wenn man nur die Winkelgeschwindigkeit verwendet
- Annahme im folgenden: der Schwerpunkt ruht im Ursprung!
- Analog zu Drehmoment (*torque*) und *linear momentum* ist der Drehimpuls (*angular momentum*) für ein Partikel:

$$\mathbf{l}_i = m_i(\mathbf{r}_i \times \dot{\mathbf{x}}_i) = \mathbf{r}_i \times (m_i\, \dot{\mathbf{x}}_i) = \mathbf{r}_i \times \mathbf{p}_i$$

- Plausibilitätsbetrachtung: das Kreuzprodukt stellt sicher, daß vom linearen *momentum* nur derjenige Anteil in das *angular momentum* eingeht, der tangential zur Partikelbahn verläuft!

- Der Drehimpuls auf das Gesamtobjekt ist

$$\mathbf{l} = \sum \mathbf{l}_i$$

- Wie entwickelt sich nun $\mathbf{l}$ über die Zeit?

$$\dot{\mathbf{l}} = \frac{d}{dt}\sum \mathbf{l}_i = \frac{d}{dt}\sum \mathbf{r}_i \times \mathbf{p}_i$$

$$= \underbrace{\sum \dot{\mathbf{r}}_i \times \mathbf{p}_i}_{=0\ (*)} + \sum \mathbf{r}_i \times \dot{\mathbf{p}}_i = \sum \mathbf{r}_i \times \mathbf{f}_i = \tau$$

- Beweis zu (*):

$$\sum \dot{\mathbf{r}}_i \times \mathbf{p}_i = \sum (\dot{\mathbf{x}}_i - \dot{\mathbf{x}}) \times \mathbf{p}_i$$

$= 0$, wegen Annahme zu Beginn des Abschnitts über Drehimpuls

$$= \sum \dot{\mathbf{x}}_i \times (m_i\, \dot{\mathbf{x}}_i) \;= 0$$

- Wegen unserer Annahme, dass $\mathbf{v} = \dot{\mathbf{x}} = 0$, gilt

$$\dot{\mathbf{x}}_i = \omega \times \mathbf{r}_i$$

- Damit kann man $\mathbf{l}$ auch so schreiben

$$\begin{aligned}\mathbf{l} &= \sum \mathbf{r}_i \times \mathbf{p}_i = \sum \mathbf{r}_i \times (m_i\, \dot{\mathbf{x}}_i) \\ &= \sum m_i \mathbf{r}_i \times (\omega \times \mathbf{r}_i) \\ &= \sum -m_i \mathbf{r}_i \times (\mathbf{r}_i \times \omega) \\ &= \sum (-m_i \mathbf{r}_i^{\times} \mathbf{r}_i^{\times})\omega \\ &= J\omega\end{aligned}$$

- Die Matrix J heißt Trägheitstensor (*inertia tensor*)

Zum Trägheitstensor

- J kann man auch so schreiben:

$$J = \sum m_i \left((\mathbf{r}_i)^2 \cdot I - \mathbf{r}_i \mathbf{r}_i^{\mathrm{T}} \right)$$

- Oder so:

$$J = \sum m_i \begin{pmatrix} r_{iy}^2 + r_{iz}^2 & -r_{ix} r_{iy} & -r_{ix} r_{iz} \\ -r_{iy} r_{ix} & r_{ix}^2 + r_{iz}^2 & -r_{iy} r_{iz} \\ -r_{iz} r_{ix} & -r_{iz} r_{iy} & r_{ix}^2 + r_{iy}^2 \end{pmatrix}$$

- Achtung: im Gegensatz zur Masse m ist J eine Funktion der Zeit:

$$J = J(t)\ !$$

  - Denn die Partikelpositionen $\mathbf{r}_i$ sind auch zeitveränderlich: $\mathbf{r}_i = \mathbf{r}_i(t)$

- Frage: kann man $J(t)$ effizient berechnen?

Die effiziente Berechnung des Trägheitstensors $J(t)$

$$
\begin{aligned}
J(t) &= \sum m_i \left(\mathbf{r}_i^\mathsf{T}\mathbf{r}_i \cdot I - \mathbf{r}_i\mathbf{r}_i^\mathsf{T}\right) \\
&= \sum m_i \left((R\bar{\mathbf{r}}_i)^\mathsf{T}(R\bar{\mathbf{r}}_i)\cdot I - (R\bar{\mathbf{r}}_i)\cdot(R\bar{\mathbf{r}}_i)^\mathsf{T}\right) \\
&= \sum m_i \left(\bar{\mathbf{r}}_i^\mathsf{T} R^\mathsf{T} R\,\bar{\mathbf{r}}_i \cdot I - R\,\bar{\mathbf{r}}_i\bar{\mathbf{r}}_i^\mathsf{T} R^\mathsf{T}\right) \\
&= \sum m_i \left(\bar{\mathbf{r}}_i^\mathsf{T}\bar{\mathbf{r}}_i \cdot I - R\,\bar{\mathbf{r}}_i\bar{\mathbf{r}}_i^\mathsf{T} R^\mathsf{T}\right) \\
&= \sum m_i \left(RR^\mathsf{T}\bar{\mathbf{r}}_i^\mathsf{T}\bar{\mathbf{r}}_i \cdot I - R\,\bar{\mathbf{r}}_i\bar{\mathbf{r}}_i^\mathsf{T} R^\mathsf{T}\right) \\
&= \sum m_i \left(R\,\bar{\mathbf{r}}_i^\mathsf{T}\bar{\mathbf{r}}_i I R^\mathsf{T} - R\,\bar{\mathbf{r}}_i\bar{\mathbf{r}}_i^\mathsf{T} R^\mathsf{T}\right) \\
&= R\Big(\sum m_i \left(\bar{\mathbf{r}}_i^\mathsf{T}\bar{\mathbf{r}}_i I - \bar{\mathbf{r}}_i\bar{\mathbf{r}}_i^\mathsf{T}\right)\Big) R^\mathsf{T} \\
&= R(t) J_0 R(t)^\mathsf{T}
\end{aligned}
$$

Wobei also J_0 der Trägheitstensor im lokalen (Objekt-) Koordinatensystem ist

Die Bewegungsgleichungen

- Nun haben wir alle Teile, um ein Objekt im freien Flug, auf das an verschiedenen Punkten Kräfte wirken, zu simulieren
- Der Zustand (zur Erinnerung):

$$
\mathbf{s}(t) = \begin{pmatrix} \mathbf{x}(t) \\ R(t) \\ \mathbf{p}(t) \\ \mathbf{l}(t) \end{pmatrix}
$$

- Die Veränderung von $s(t)$ über die Zeit ist

$$
\dot{s}(t) = \begin{pmatrix} \dot{\mathbf{x}} \\ \dot{R} \\ \dot{\mathbf{p}} \\ \dot{\mathbf{l}} \end{pmatrix} = \begin{pmatrix} m^{-1}\mathbf{p} \\ \omega \times R \\ \mathbf{f} \\ \tau \end{pmatrix} = \begin{pmatrix} m^{-1}\mathbf{p} \\ (J^{-1}\mathbf{l}) \times R \\ \mathbf{f} \\ \tau \end{pmatrix}
$$

- Damit sieht der Algorithmus für eine explizite Euler-Integration der Bewegungsgleichungen so aus:

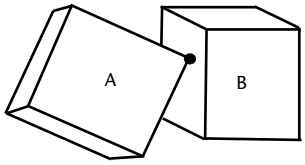
```
berechne m, Schwerpunkt, und J_0^-1
init x, v, R, l

loop:
  berechne f = Σ f_i  und  τ = Σ r_i × f_i
  x = x + Δt m^-1 p
  p = p + Δt f
  R = R + Δt · (J^-1 l) × R
  l = l + Δt τ
```


Collision Handling

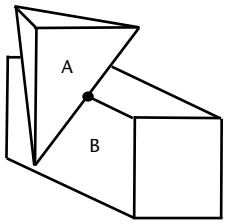
- Es gibt 2 Arten von Kontakten zwischen *rigid bodies*:
  - *Colliding contact* → die Geschwindigkeit ändert sich unstetig
  - *Resting contact* → die Geschwindigkeit ändert sich nicht (an dieser Stelle)
    - Bsp.: Würfel, der auf einer schiefen Ebene rutscht

1. Schritt: exakten Kollisionszeitpunkt t_0 berechnen
  - Durch Intervallhalbierung
  - Oder durch *continuous collision detection*

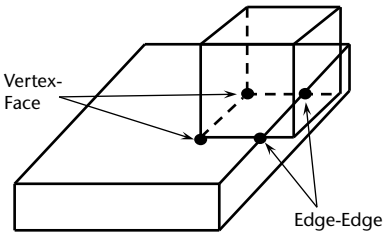
2. Schritt (i.A. zusammen mit dem ersten): alle Kontakte berechnen
   - Jeder Kontakt ist von einer der folgenden beiden Arten
   - Vertex-Face:



   - Edge-Edge:

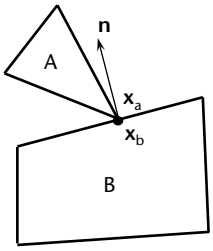


3. Schritt: Kontaktnormale **n** berechnen
   - Bei Vertex-Face: **n** = Normale des einen beteiligten Polygons
   - Bei Edge-Edge: $\mathbf{n} = \mathbf{e}_a \times \mathbf{e}_b$
- Beispiel:



- Sei t_0 = Kollisionszeitpunkt, $\mathbf{x}_a$, $\mathbf{x}_b$ = die Punkte auf Objekt A bzw. Obj B, die zum Zeitpunkt t_0 in Kontakt sind, d.h.

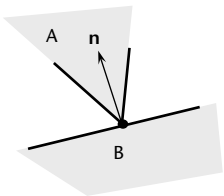
$$\mathbf{x}_a(t_0) = \mathbf{x}_b(t_0)$$



4. Schritt: Kollisionsart bestimmen

- Bezeichne mit $\dot{\mathbf{x}}_a^-$, $\dot{\mathbf{x}}_b^-$ die Geschwindigkeit von $\mathbf{x}_a$, $\mathbf{x}_b$ unmittelbar vor der Kollision
- Berechne

$$v_{\text{rel}}^- = (\dot{\mathbf{x}}_a^- - \dot{\mathbf{x}}_b^-) \cdot \mathbf{n}$$



- Falls

$$v_{\text{rel}}^- > 0 \quad \Rightarrow \quad \text{Kontakt löst sich auf}$$
$$v_{\text{rel}}^- < 0 \quad \Rightarrow \quad \text{colliding contact}$$
$$v_{\text{rel}}^- \approx 0 \quad \Rightarrow \quad \text{resting contact}$$

- Jetzt kommt ein Trick: wir führen keine zusätzlichen Kräfte ein, sondern wir ändern die Geschwindigkeit direkt, so daß der Kontakt sich im nächsten Frame wieder auflöst.
  M.a.W.: wir führen einen zusätzlichen Kontaktimpuls ein, der auf die beiden Kontaktpunkte $\mathbf{x}_a$ und $\mathbf{x}_b$ wirkt
- Bezeichne mit $\Delta\mathbf{p}$ diesen zusätzlichen Kontaktimpuls
- Beobachtung: $\Delta\mathbf{p}$ kann nur entlang der Kontaktnormalen wirken!
- Wir wisen also schon:

$$\Delta\mathbf{p} = \Delta p \cdot \mathbf{n}$$

5. Schritt: bestimmte Δp so, daß unmittelbar nach der Kollision gilt

$$v_{\text{rel}}^{+} = -\varepsilon v_{\text{rel}}^{-}$$

wobei ε = *coefficient of restitution* (Stoßzahl)

- $\varepsilon = 1$ → voll elastischer Stoß
- $\varepsilon = 0$ → voll inelastischer Stoß

- Die Änderung im linearen Impuls und Drehimpuls (*linear / angular momentum*) sind:

$$\Delta\mathbf{p} = m_A \Delta\mathbf{v}_A$$

$$\mathbf{r}_A \times (\Delta\mathbf{p}) = J_A(\Delta\omega_A)$$

- Das bewirkt eine (unstetige!) Änderung der Geschwindigkeit für $\mathbf{x}_a$ entlang der Kontaktnormale $\mathbf{n}$:

$$\begin{aligned}(\Delta\dot{\mathbf{x}}_a)\cdot\mathbf{n} &= (\Delta\omega_A \times \mathbf{r}_a + \Delta\mathbf{v}_A)\cdot\mathbf{n} \\ &= \left((J_A^{-1}(\mathbf{r}_a \times \Delta\mathbf{p})) \times \mathbf{r}_a + m_A^{-1}\cdot\Delta\mathbf{p}\right)\cdot\mathbf{n} \\ &= \Delta p\left((J_A^{-1}(\mathbf{r}_a \times \mathbf{n})) \times \mathbf{r}_a + m_A^{-1}\cdot\mathbf{n}\right)\cdot\mathbf{n} \\ &= \Delta p\left((J_A^{-1}(\mathbf{r}_a \times \mathbf{n})) \times \mathbf{r}_a\cdot\mathbf{n} + m_A^{-1}\cdot\mathbf{n}\cdot\mathbf{n}\right) \\ &= \Delta p\left((\mathbf{r}_a \times \mathbf{n})^{\mathsf{T}} J_A^{-1}(\mathbf{r}_a \times \mathbf{n}) + m_A^{-1}\right) \\ &=: \Delta p\cdot w\end{aligned}$$

Verwendete Rechenregeln

- Eine Identität für das Tripel-Produkt (*triple product*):

$$(\mathbf{a}\times\mathbf{b})\cdot\mathbf{c} = \mathbf{c}\cdot(\mathbf{a}\times\mathbf{b}) = \mathbf{a}\cdot(\mathbf{b}\times\mathbf{c})$$

  - Hier mit folgender Belegung angewendet:

$$\begin{aligned}J_A^{-1}(\mathbf{r}_A \times \mathbf{n}) &= \mathbf{a} \\ \mathbf{r}_A &= \mathbf{b} \\ \mathbf{n} &= \mathbf{c}\end{aligned}$$

  - Danach muss $J_A^{-1}(\mathbf{r}_A \times \mathbf{n})$ noch transponiert werden, da es jetzt auf der linken Seite eines Skalarproduktes steht
- Für symmetrische Matrizen A ist A^{-1} wieder symmetrisch, also gilt

$$(A^{-1})^{\mathsf{T}} = A^{-1}$$

- Jetzt noch einsetzen, um $\Delta \mathrm{p}$ zu bestimmen:

$$\begin{aligned} v_{rel}^{+} &= v_{rel}^{-} + \Delta\dot{\mathbf{x}}_a \cdot \mathbf{n} - \Delta\dot{\mathbf{x}}_b \cdot \mathbf{n} \\ &= v_{rel}^{-} + pw_a - pw_b \\ &= v_{rel}^{-} + p(w_a - w_b) \\ &\stackrel{!}{=} -\varepsilon\, v_{rel}^{-} \end{aligned}$$

- Nach p auflösen liefert:

$$p = \frac{(1+\epsilon)v_{rel}^{-}}{w_b - w_a}$$

6. Schritt: neue Geschwindigkeiten berechnen

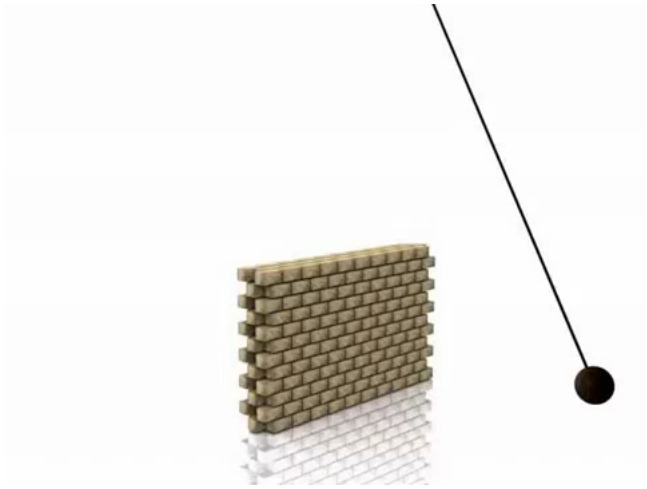
$$\mathbf{v}_A^{+} = \mathbf{v}_A^{-} + \Delta\mathbf{v}_A = \mathbf{v}_A^{-} + m_A^{-1} p\mathbf{n}$$

$$\mathbf{v}_B^{+} = \mathbf{v}_B^{-} - m_B^{-1} p\mathbf{n}$$

$$\omega_A^{+} = \omega_A^{-} + \Delta\omega_A = \omega_A^{-} + J_A^{-1}(\mathbf{r}_A \times \Delta\mathbf{p}) = \omega_A^{-} + J_A^{-1}(\mathbf{r}_A \times p\mathbf{n})$$

$$\omega_B^{+} = \cdots$$

Videos



3D Studio Max 9

Bullet Physics engine

TSN Studios 2009