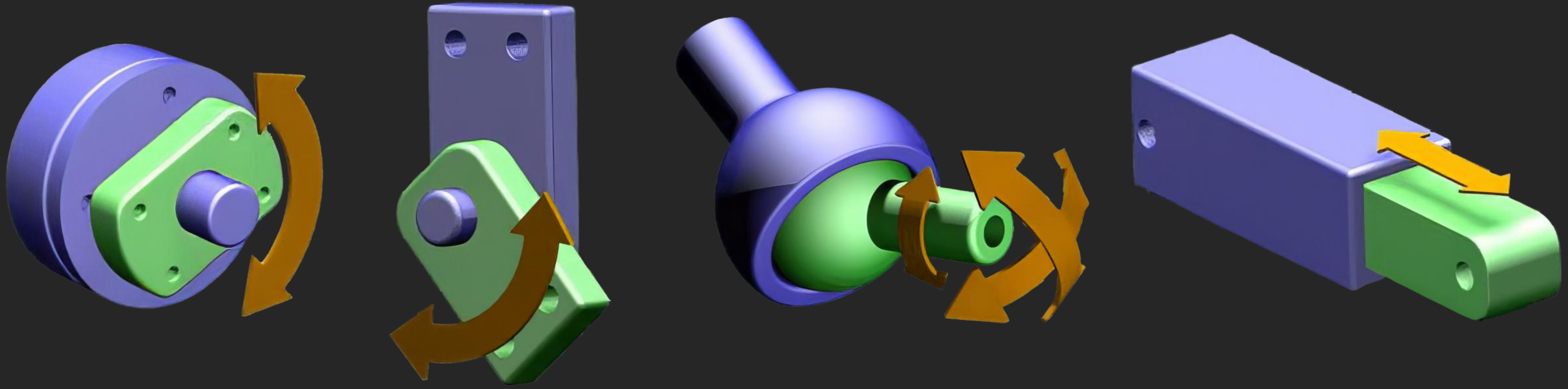
Joint Simulation



Matthias Müller, Ten Minute Physics

matthiasmueller.info/tenMinutePhysics

Simulation Model

- XPBD: Extended Position Based Dynamics, (tutorial 9)

- Simple to Implement
- Unconditionally stable!

- Physically derived (unlike PBD)

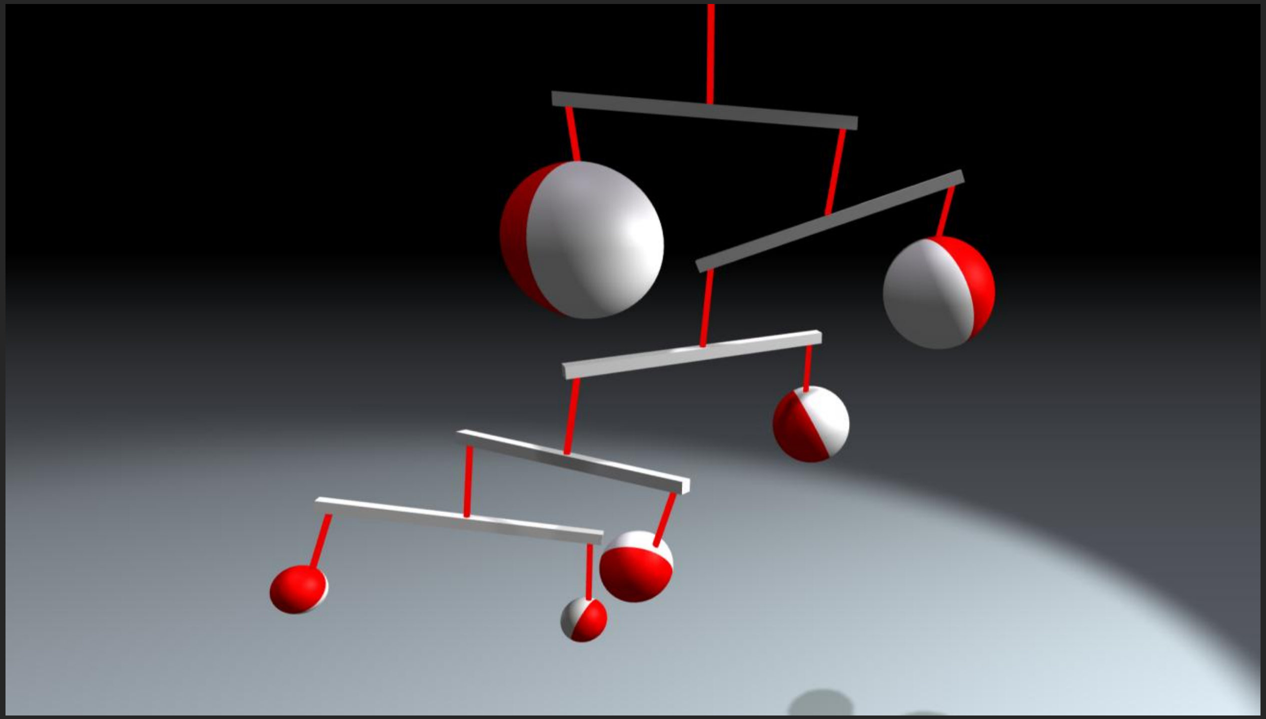
- Traditional Methods:
  Solve complicated large systems of complementarity problems 🤔

$$0 \le \begin{bmatrix} {}^u\mathbf{J}_n^{(\ell)} \mathbf{u}^{(\ell+1)} + \frac{{}^u\mathbf{C}_n^{(\ell)}}{\Delta t} + \frac{\partial {}^u\mathbf{C}_n^{(\ell)}}{\partial t} \\ {}^u\mathbf{D}^T\, {}^u\mathbf{J}_f \mathbf{u}^{(\ell+1)} + {}^u\mathbf{E}\, {}^u\beta \\ {}^b\mathbf{D}^T\, {}^b\mathbf{J}_f \mathbf{u}^{(\ell+1)} + {}^b\mathbf{E}\, {}^b\beta \\ \mathbf{U}\, {}^u\mathbf{p}_n^{(\ell+1)} - {}^u\mathbf{E}^T\, {}^u\alpha \\ {}^b\mathbf{p}_{f\max} - {}^b\mathbf{E}^T\, {}^b\alpha \end{bmatrix} \perp \begin{bmatrix} {}^u\mathbf{p}^{(\ell+1)} \\ {}^u\alpha \\ {}^b\alpha \\ {}^u\beta \\ {}^b\beta \end{bmatrix} \ge 0.$$

$$\begin{aligned} \widehat{{}^\kappa C_{i\sigma}}(\tilde{\mathbf{q}}, \tilde{t}) &= {}^\kappa C_{i\sigma}(\mathbf{q}, t) \\ &+ \frac{\partial {}^\kappa C_{i\sigma}}{\partial \mathbf{q}} \Delta\mathbf{q} + \frac{\partial {}^\kappa C_{i\sigma}}{\partial t} \Delta t \\ &+ \frac{1}{2}\left((\Delta\mathbf{q})^T \frac{\partial^2 {}^\kappa C_{i\sigma}}{\partial \mathbf{q}^2} \Delta\mathbf{q} + 2 \frac{\partial^2 {}^\kappa C_{i\sigma}}{\partial \mathbf{q}\partial t} \Delta\mathbf{q}\Delta t + \frac{\partial^2 {}^\kappa C_{i\sigma}}{\partial t^2} \Delta t^2 \right) \\ {}^\kappa\mathbf{J}_{i\sigma} &= \frac{\partial({}^\kappa\mathbf{C}_{i\sigma})}{\partial \mathbf{q}} \mathbf{H} \\ {}^\kappa\mathbf{k}_{i\sigma}(\mathbf{q}, \mathbf{u}, t) &= \frac{\partial({}^\kappa\mathbf{C}_{i\sigma})}{\partial \mathbf{q}} \frac{\partial \mathbf{H}}{\partial t} \mathbf{u} + \frac{\partial^2({}^\kappa\mathbf{C}_{i\sigma})}{\partial \mathbf{q}\partial t} \mathbf{H}\mathbf{u} + \frac{\partial^2({}^\kappa\mathbf{C}_{i\sigma})}{\partial t^2}, \end{aligned}$$

- XPBD: Forward execution of simple formulas that are easy to understand 🙂

Rigid Body Simulation Recap



Tutorial 22

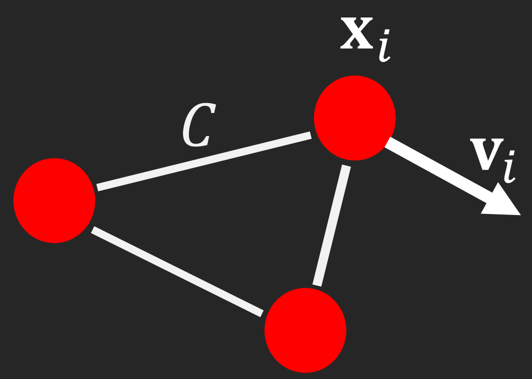
XPBD Algorithm for Particles

while simulating

  for all particles i
    $\mathbf{v}_i \leftarrow \mathbf{v}_i + \Delta t\ \mathbf{g}$
    $\mathbf{p}_i \leftarrow \mathbf{x}_i$
    $\mathbf{x}_i \leftarrow \mathbf{x}_i + \Delta t\ \mathbf{v}_i$

  for n iterations
    for all constraints C
      solve$(C, \Delta t)$

  for all particles i
    $\mathbf{v}_i \leftarrow (\mathbf{x}_i - \mathbf{p}_i)/\Delta t$



solve$(C, \Delta t)$:

  for all particles i in C
    compute $\Delta \mathbf{x}_i$
    $\mathbf{x}_i \leftarrow \mathbf{x}_i + \Delta \mathbf{x}_i$

Sub-Stepping

Magic trick: Much faster convergence (like a global solver):

while simulating

  for all particles i

    $\mathbf{v}_i \leftarrow \mathbf{v}_i + \Delta t\ \mathbf{g}$

    $\mathbf{p}_i \leftarrow \mathbf{x}_i$

    $\mathbf{x}_i \leftarrow \mathbf{x}_i + \Delta t\ \mathbf{v}_i$

  for n iterations

    for all constraints C

      solve$(C, \Delta t)$

  for all particles i

    $\mathbf{v}_i \leftarrow (\mathbf{x}_i - \mathbf{p}_i)/\Delta t$

while simulating

  for n sub-steps

    for all particles i

      $\mathbf{v}_i \leftarrow \mathbf{v}_i + \Delta t\ \mathbf{g}$

      $\mathbf{p}_i \leftarrow \mathbf{x}_i$

      $\mathbf{x}_i \leftarrow \mathbf{x}_i + \Delta t\ \mathbf{v}_i$

    for all constraints C

      solve$(C, \Delta t)$

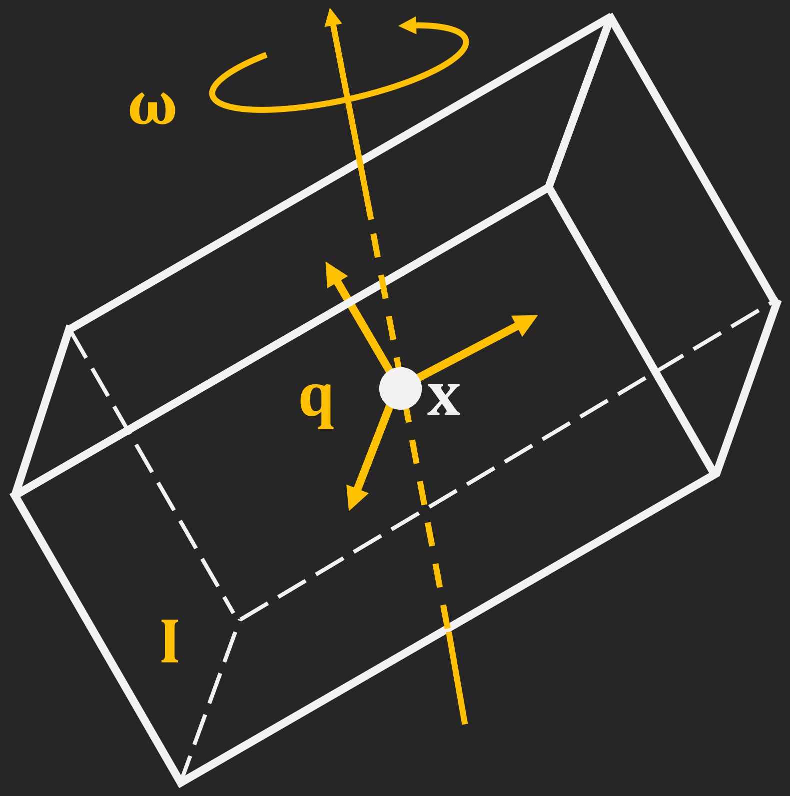
    for all particles i

      $\mathbf{v}_i \leftarrow (\mathbf{x}_i - \mathbf{p}_i)/\Delta t$

Orientational Quantities

A rigid body also has

- an orientation $\mathbf{q}$
- an angular velocity $\boldsymbol{\omega}$
- and the moment of inertia $\mathbf{I}$



XPBD Algorithm for Rigid Bodies

while simulating

  for n sub-steps

    for all bodies i

      integrate $\mathbf{v}_i, \mathbf{x}_i, \boldsymbol{\omega}_i, \mathbf{q}_i$

    for all constraints C

      solve$(C, \Delta t)$

    for all bodies i

      update $\mathbf{v}_i, \boldsymbol{\omega}_i$

solve$(C, \Delta t)$:

  for all bodies i in C

    compute $\Delta\mathbf{x}_i, \Delta\mathbf{q}_i$

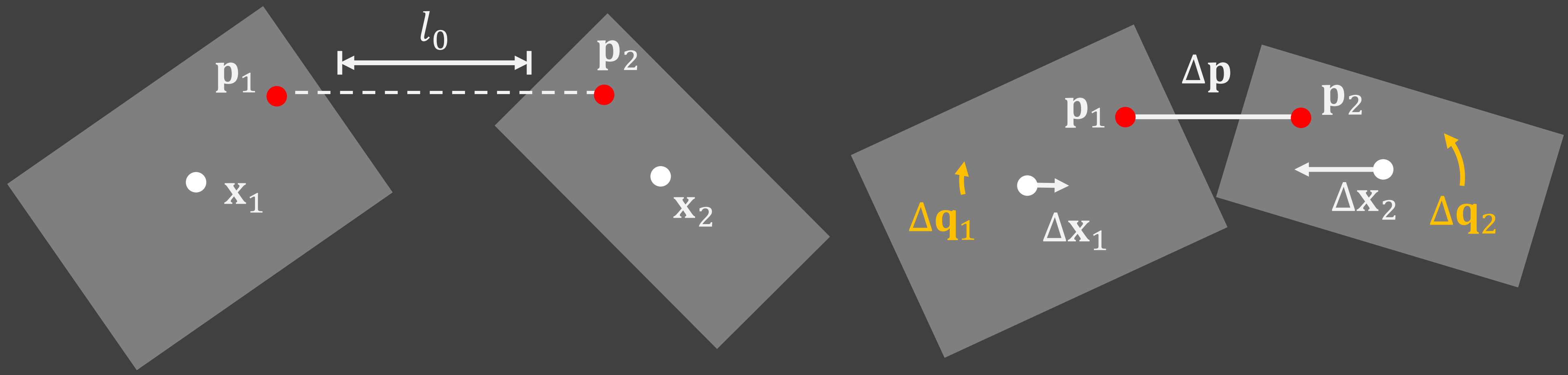
    $\mathbf{x}_i \leftarrow \mathbf{x}_i + \Delta\mathbf{x}_i$

    $\mathbf{q}_i \leftarrow \mathbf{q}_i + \Delta\mathbf{q}_i$

Constraints

Distance Constraint

corrections proportional to m^{-1}



corrections proportional to $m^{-1}$and $\mathbf{I}^{-1}$

Linear Correction

ApplyLinearCorrection$(\mathbf{p}_1, \mathbf{p}_2, \Delta\mathbf{p}, \alpha)$

$C \leftarrow |\Delta\mathbf{p}|$

$\mathbf{n} \leftarrow \Delta\mathbf{p} / |\Delta\mathbf{p}|$

$w_i \leftarrow m_i^{-1} + ((\mathbf{p}_i - \mathbf{x}_i) \times \mathbf{n})^{\mathrm{T}} \mathbf{I}_i^{-1} ((\mathbf{p}_i - \mathbf{x}_i) \times \mathbf{n})$

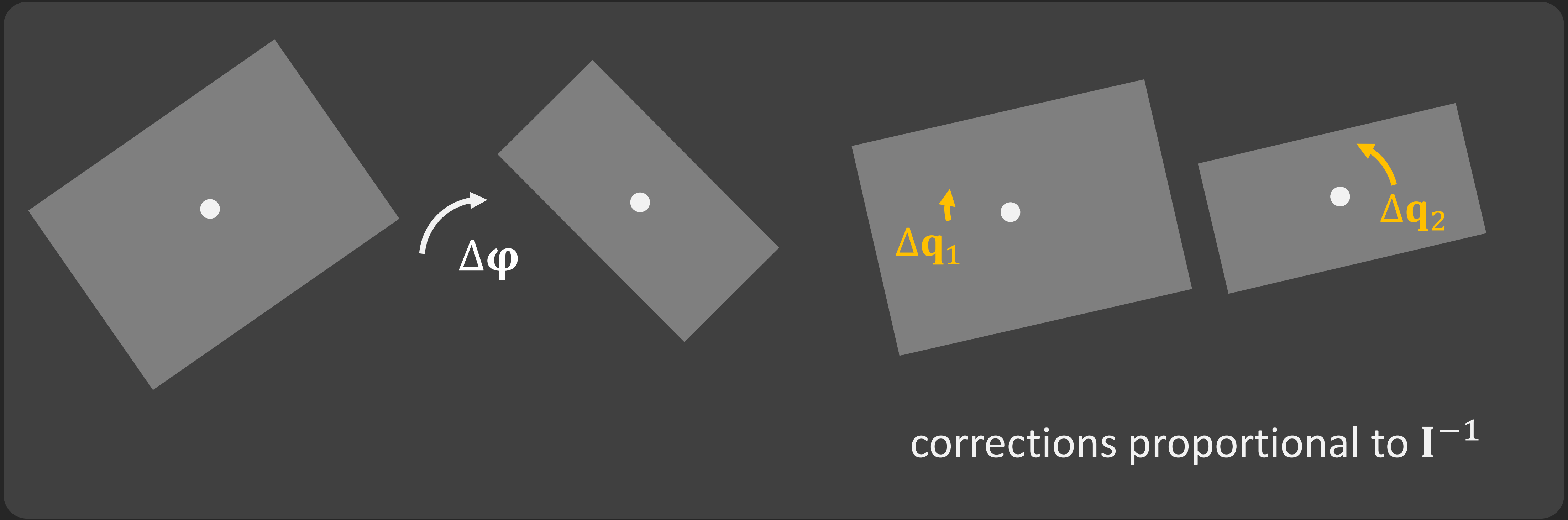
$\lambda \leftarrow -C \cdot (w_1 + w_2 + \frac{\alpha}{\Delta t^2})^{-1}$

$\mathbf{x}_i \leftarrow \mathbf{x}_i \pm \lambda \mathbf{n}\, m_i^{-1}$

$\mathbf{q}_i \leftarrow \mathbf{q}_i \pm \frac{1}{2} \lambda [\mathbf{I}_i^{-1} ((\mathbf{p}_i - \mathbf{x}_i) \times \mathbf{n}), 0]\, \mathbf{q}_i$

- Compliance α is the inverse of stiffness
- w is the inverse of mass
- Stable handling of infinite stiffness: $\alpha = 0$
- Stable handling of infinite mass: $w_i = 0$
- $\lambda \mathbf{n} / \Delta t^2$ yields the constraint force

Orientation Constraint



Angular Correction

ApplyAngularCorrection$(\Delta\boldsymbol{\varphi}, \alpha)$

$$C \leftarrow |\Delta\boldsymbol{\varphi}|$$

$$\mathbf{n} \leftarrow \Delta\boldsymbol{\varphi} / |\Delta\boldsymbol{\varphi}|$$

$$w_i \leftarrow \mathbf{n}^T \mathbf{I}_i^{-1} \mathbf{n}$$

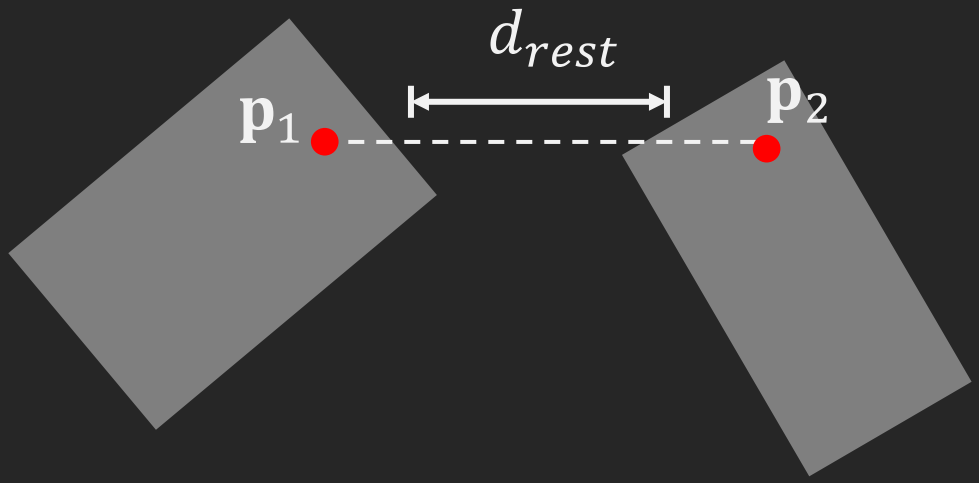
$$\lambda \leftarrow -C \cdot \left(w_1 + w_2 + \frac{\alpha}{\Delta t^2}\right)^{-1}$$

$$\mathbf{q}_i \leftarrow \mathbf{q}_i \pm \frac{1}{2} \lambda \left[\mathbf{I}_i^{-1} \mathbf{n}, 0\right] \mathbf{q}_i$$

$\lambda \mathbf{n} / \Delta t^2$ yields the constraint torque

Building Blocks

Attach Bodies

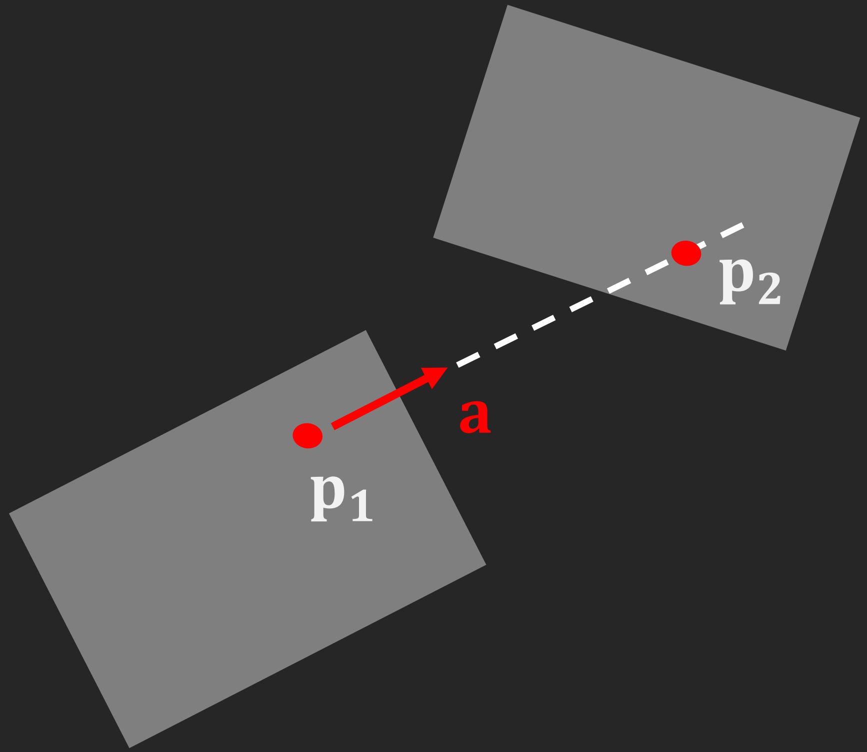


Attach$(\mathbf{p}_1, \mathbf{p}_2, d_{rest}, \alpha)$

$d \leftarrow |\mathbf{p}_2 - \mathbf{p}_1|$
$\mathbf{n} \leftarrow (\mathbf{p}_2 - \mathbf{p}_1)/|\mathbf{p}_2 - \mathbf{p}_1|$

ApplyLinearCorrection$(\mathbf{p}_1, \mathbf{p}_2, -(d - d_{rest})\mathbf{n}, \alpha)$

Restrict to Axis



RestrictToAxis$(\mathbf{a}, \mathbf{p}_1, \mathbf{p}_2, p_{min}, p_{max}, \alpha)$

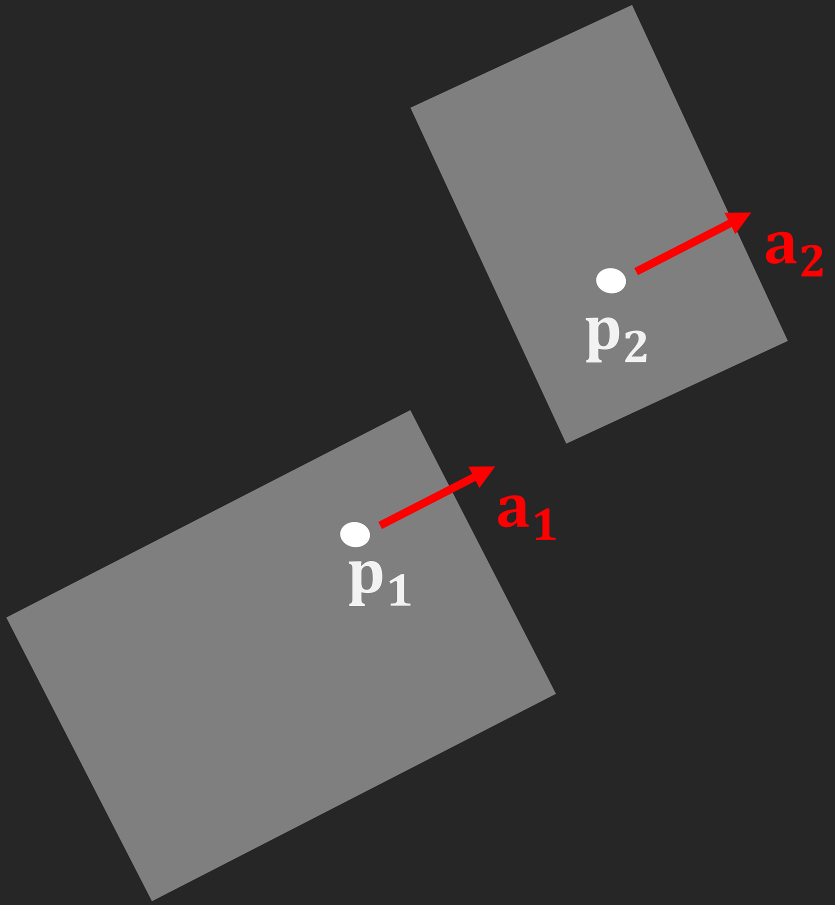
$\mathbf{p} \leftarrow \mathbf{p}_2 - \mathbf{p}_1$
$p \leftarrow \mathbf{a} \cdot \mathbf{p}$

if $p < p_{min}$ **then** $p \leftarrow p_{min}$
else if $p > p_{max}$ **then** $p \leftarrow p_{max}$

$\mathbf{p} \leftarrow \mathbf{p} - p\mathbf{a}$

ApplyLinearCorrection$(\mathbf{p}_1, \mathbf{p}_2, -\mathbf{p}, \alpha)$

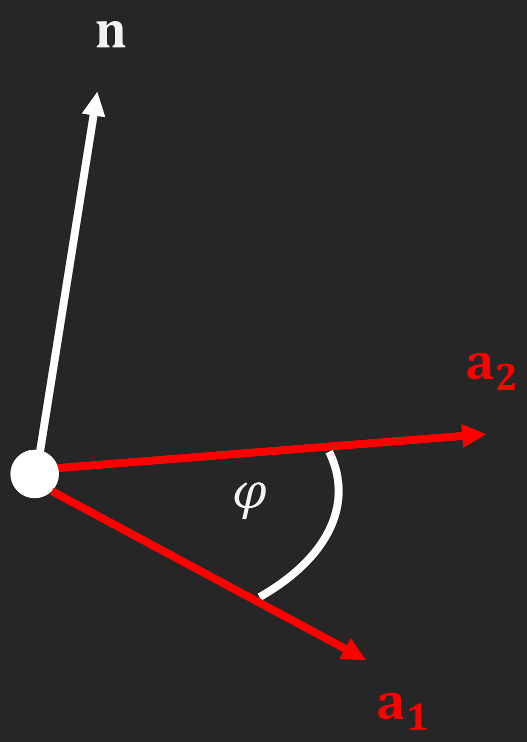
Align two Axes



AlignAxes$(\mathbf{a}_1, \mathbf{a}_2, \alpha)$

ApplyAngularCorrection$(-\mathbf{a}_1 \times \mathbf{a}_2, \alpha)$

Limit Angle



LimitAngle$(\mathbf{n}, \mathbf{a}_1, \mathbf{a}_2, \varphi_{min}, \varphi_{max}, \alpha)$

$\varphi \leftarrow angle(\mathbf{n}, \mathbf{a}_1, \mathbf{a}_2)$

if $\varphi < \varphi_{min}$ or $\varphi > \varphi_{max}$

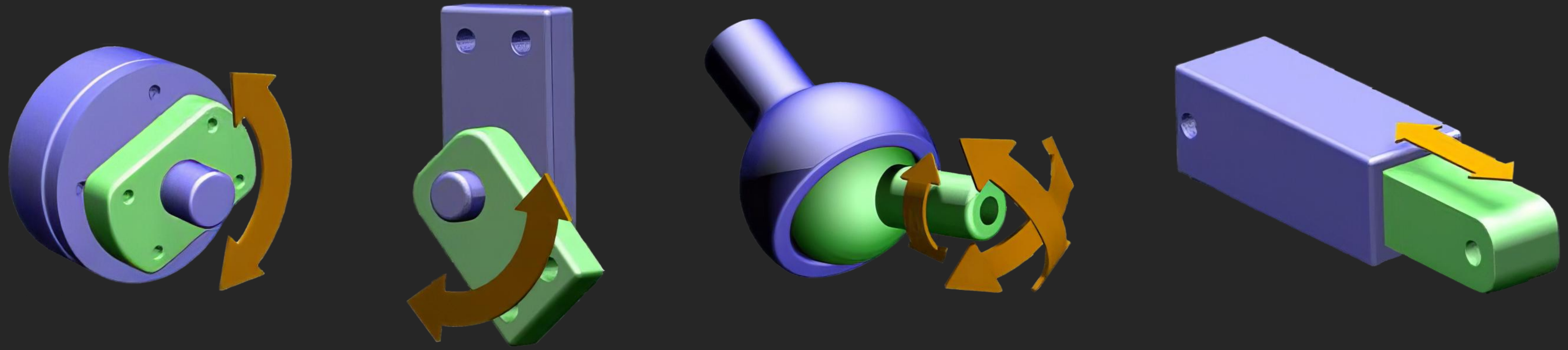
$\varphi \leftarrow clamp(\varphi, \varphi_{min}, \varphi_{max})$

$\mathbf{q} \leftarrow rotation(\mathbf{n}, \varphi)$

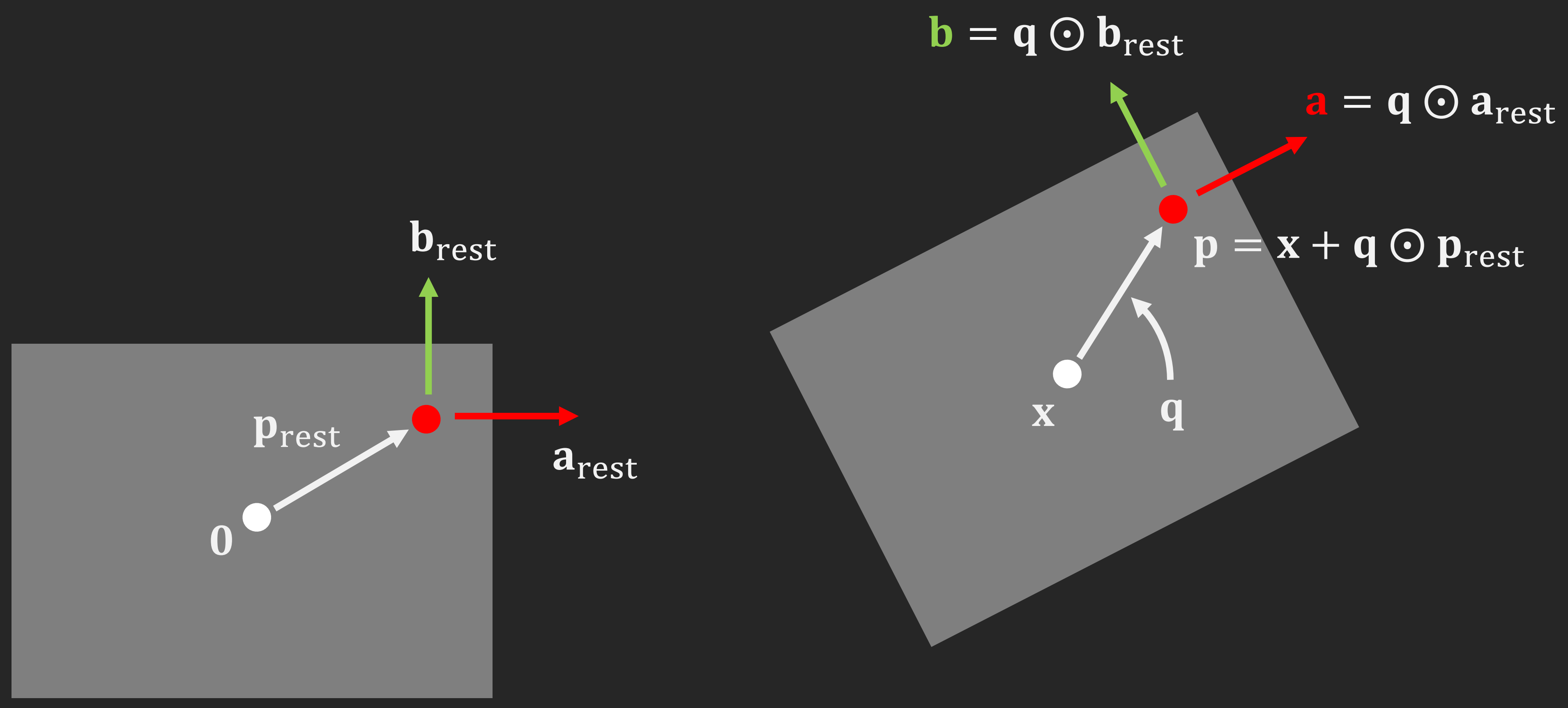
$\mathbf{a}_2{}' \leftarrow \mathbf{q} \odot \boldsymbol{a}_1$

ApplyAngularCorrection$(-\mathbf{a}_2 \times \mathbf{a}_2{}', \alpha)$

Joints



Attachment Frames (2d)



Rest state stored on the body

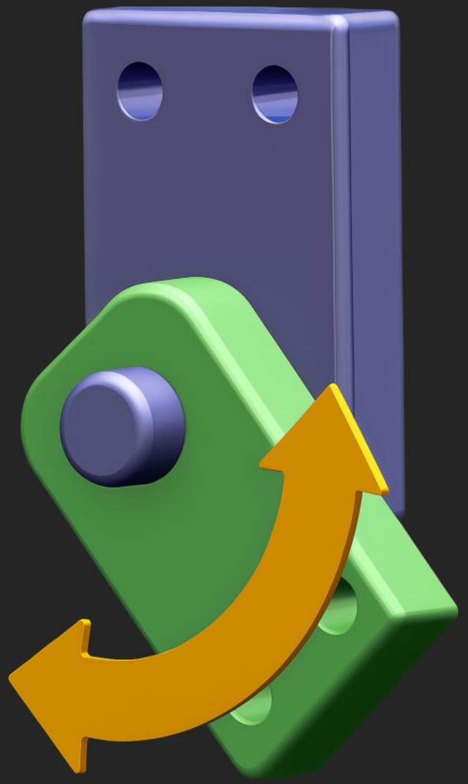
Current state

Hinge Joint

$\text{Attach}(\mathbf{p}_1, \mathbf{p}_2, d_{rest} = 0, \alpha = 0)$

$\text{AlignAxes}(\boldsymbol{a}_1, \mathbf{a}_2, \alpha = 0)$

$\text{LimitAngle}(\mathbf{a}_1, \mathbf{b}_1, \mathbf{b}_2, \varphi_{min}, \varphi_{max}, \alpha = 0)$

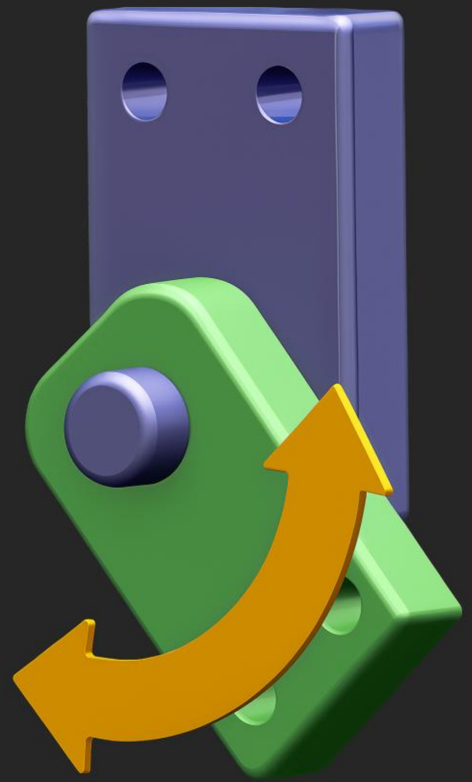


Servo

Attach$(\mathbf{p}_1, \mathbf{p}_2, d_{rest} = 0, \alpha = 0)$

AlignAxes$(\mathbf{a}_1, \mathbf{a}_2, \alpha = 0)$

LimitAngle$(\mathbf{a}_1, \mathbf{b}_1, \mathbf{b}_2, \varphi_{servo}, \varphi_{servo}, \alpha = 0)$



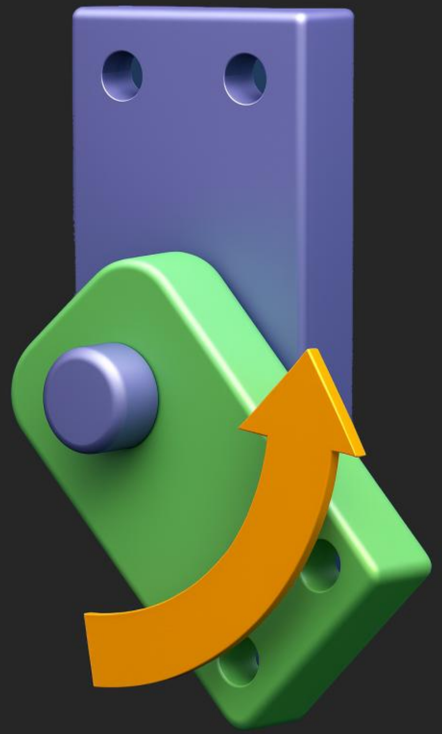
Velocity Motor

$\text{Attach}(\mathbf{p}_1, \mathbf{p}_2, d_{rest} = 0, \alpha = 0)$

$\text{AlignAxes}(\mathbf{a}_1, \mathbf{a}_2, \alpha = 0)$

$\text{LimitAngle}(\mathbf{a}_1, \mathbf{b}_1, \mathbf{b}_2, \varphi_{motor}, \varphi_{motor}, \alpha = 0)$

$$\varphi_{motor} \leftarrow \varphi_{motor} + \Delta t\ \omega_{motor}$$



Ball Joint

$\text{Attach}(\mathbf{p}_1, \mathbf{p}_2, d_{rest} = 0, \alpha = 0)$

$\mathbf{n} \leftarrow (\mathbf{a}_1 \times \mathbf{a}_2)/|\mathbf{a}_1 \times \mathbf{a}_2|$

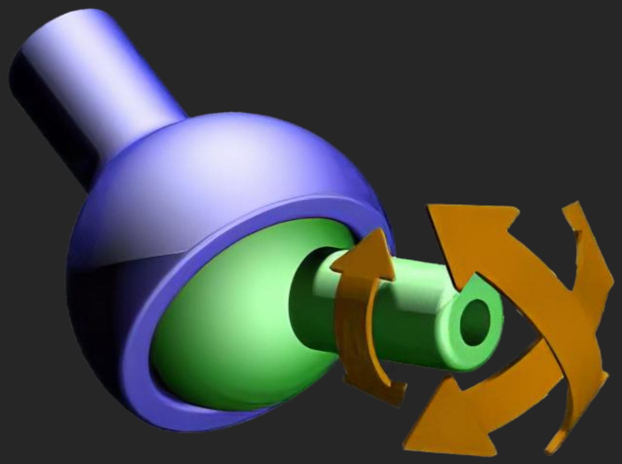
$\text{LimitAngle}(\mathbf{n}, \mathbf{a}_1, \mathbf{a}_2, 0, \varphi_{swing\_max}, \alpha = 0)$

$\mathbf{n} \leftarrow (\mathbf{a}_1 + \mathbf{a}_2)/|\mathbf{a}_1 + \mathbf{a}_2|$

$\mathbf{b}_1{}' \leftarrow \mathbf{b}_1 - \mathbf{n}(\mathbf{n} \cdot \mathbf{b}_1)$

$\mathbf{b}_2{}' \leftarrow \mathbf{b}_2 - \mathbf{n}(\mathbf{n} \cdot \mathbf{b}_2)$

$\text{LimitAngle}(\mathbf{n}, \mathbf{b}_1{}', \mathbf{b}_2{}', \varphi_{twist\_min}, \varphi_{twist\_max}, \alpha = 0)$

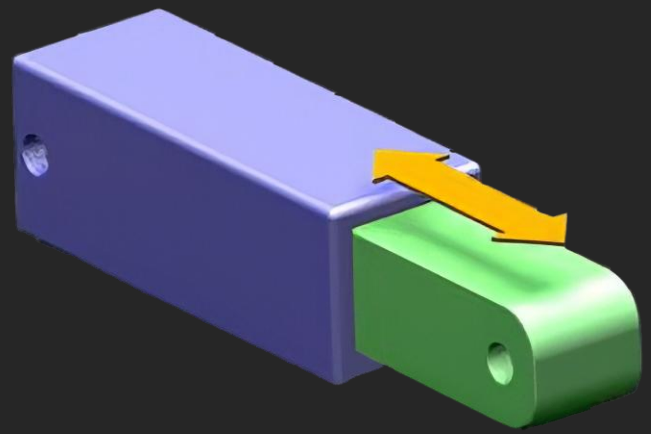


Prismatic Joint

RestrictToAxis$(\mathbf{a}_1, \mathbf{p}_1, \mathbf{p}_2, p_{min}, p_{max}, \alpha)$

AlignAxes$(\mathbf{a}_1, \mathbf{a}_2, \alpha = 0)$

LimitAngle$(\mathbf{a}_1, \mathbf{b}_1, \mathbf{b}_2, \varphi_{min}, \varphi_{max}, \alpha)$

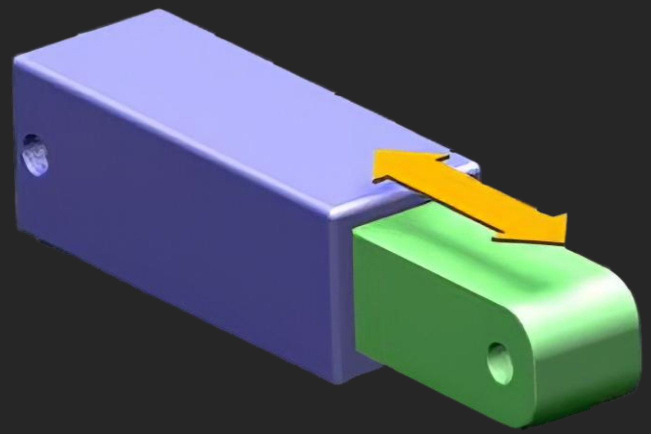


Cylinder

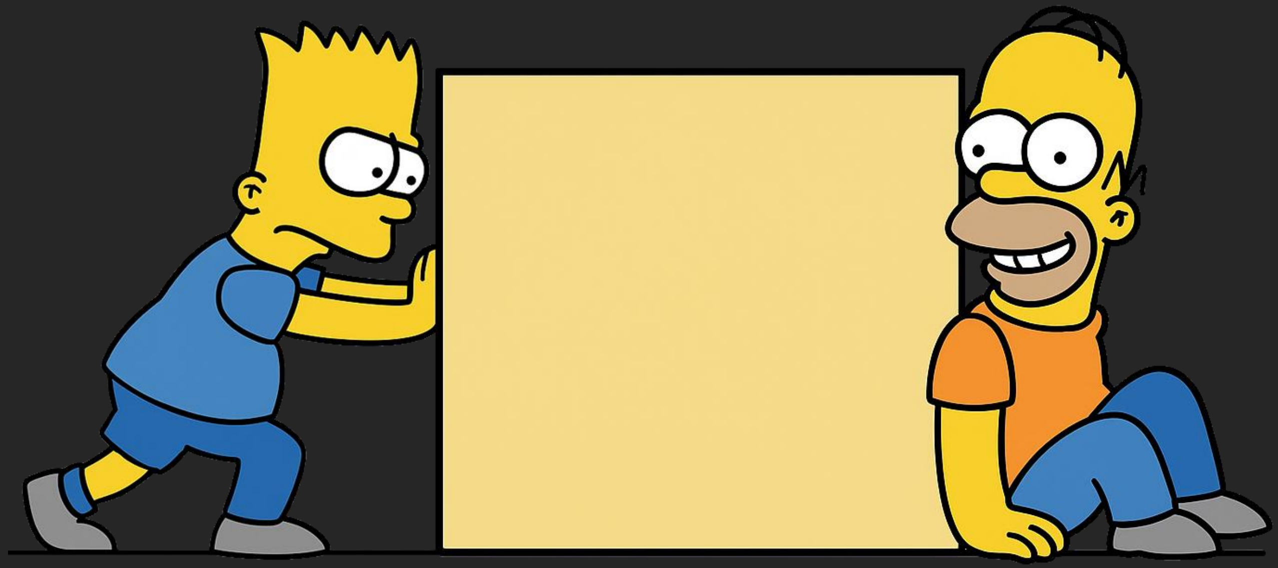
RestrictToAxis$(\mathbf{a}_1, \mathbf{p}_1, \mathbf{p}_2, p_{target}, p_{target}, \alpha = 0)$

AlignAxes$(\mathbf{a}_1, \mathbf{a}_2, \alpha = 0)$

LimitAngle$(\mathbf{a}_1, \mathbf{b}_1, \mathbf{b}_2, \varphi_{cylinder}, \varphi_{cylinder}, \alpha)$



Velocity Level



Forces, Torques, Damping

Velocity Step

while simulating

  for n sub-steps

    for all bodies i

      integrate $\mathbf{v}_i, \mathbf{x}_i, \boldsymbol{\omega}_i, \mathbf{q}_i$

    for all constraints C

      solve$(C, \Delta t)$

    for all bodies i

      update $\mathbf{v}_i, \boldsymbol{\omega}_i$

    for all constraints C

      apply velocity corrections

Linear Velocity Correction

ApplyLinearVelocityCorrection$(\mathbf{p}_1, \mathbf{p}_2, \Delta\mathbf{v})$

$$\Delta v \leftarrow |\Delta\mathbf{v}|$$

$$\mathbf{n} \leftarrow |\Delta\mathbf{v}| / |\Delta\mathbf{v}|$$

$$w_i \leftarrow m_i^{-1} + ((\mathbf{p}_i - \mathbf{x}_i) \times \mathbf{n})^{\mathrm{T}} \mathbf{I}_i^{-1} ((\mathbf{p}_i - \mathbf{x}_i) \times \mathbf{n})$$

$$\lambda \leftarrow -\Delta v \cdot (w_1 + w_2)^{-1}$$

$$\mathbf{v}_i \leftarrow \mathbf{v}_i \pm \lambda \mathbf{n}\, m_i^{-1}$$

$$\boldsymbol{\omega}_i \leftarrow \boldsymbol{\omega}_i \pm \lambda \mathbf{I}_i^{-1} (\mathbf{r}_i \times \mathbf{n})$$

Angular Velocity Correction

ApplyAngularVelocityCorrection($\Delta\omega$)

$$\Delta\omega \leftarrow |\Delta\boldsymbol{\omega}|$$

$$\mathbf{n} \leftarrow |\Delta\boldsymbol{\omega}| \big/ |\Delta\boldsymbol{\omega}|$$

$$w_i \leftarrow \mathbf{n}^T \mathbf{I}_i^{-1} \mathbf{n}$$

$$\lambda \leftarrow \Delta\omega \cdot (w_1 + w_2)^{-1}$$

$$\boldsymbol{\omega}_i \leftarrow \boldsymbol{\omega}_i \pm \lambda \mathbf{I}_i^{-1} \mathbf{n}$$

Linear Damping

DampLinear$(\mathbf{p}_1, \mathbf{p}_2, \mathbf{n}, c_{linear})$

$$\Delta\mathbf{v} \leftarrow \mathbf{v}_2 + (\mathbf{p}_2 - \mathbf{x}_2) \times \boldsymbol{\omega}_2 - \mathbf{v}_1 - (\mathbf{p}_1 - \mathbf{x}_1) \times \boldsymbol{\omega}_1$$

$$\Delta v \leftarrow \mathbf{n} \cdot \Delta\mathbf{v}$$

$$\Delta v \leftarrow \Delta v \min(\Delta t\, c_{linear}, 1)$$

ApplyLinearVelocityCorrection$(\mathbf{p}_1, \mathbf{p}_2, -\Delta v\, \mathbf{n})$

Angular Damping

DampAngular$(\mathbf{n}, c_{angular})$

$$\Delta\boldsymbol{\omega} \leftarrow \boldsymbol{\omega}_2 - \boldsymbol{\omega}_1$$

$$\Delta\omega \leftarrow \mathbf{n} \cdot \Delta\boldsymbol{\omega}$$

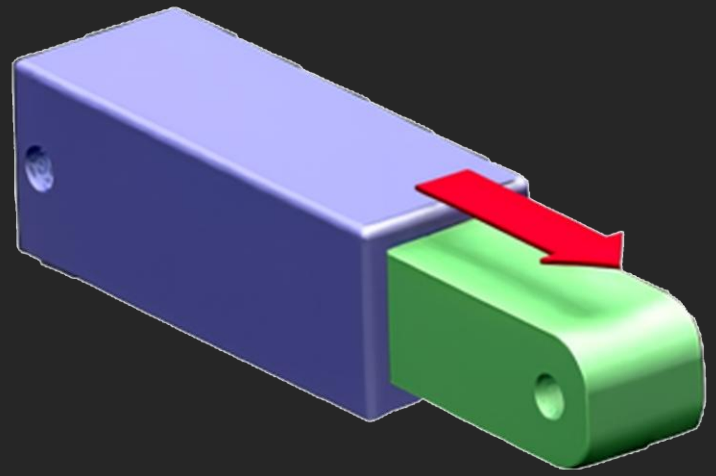
$$\Delta\omega \leftarrow \Delta\omega \min(\Delta t c_{angular}, 1)$$

ApplyAngularVelocityCorrection$(-\Delta\omega\mathbf{n})$

Apply a Cylinder Force

ApplyForce(f)

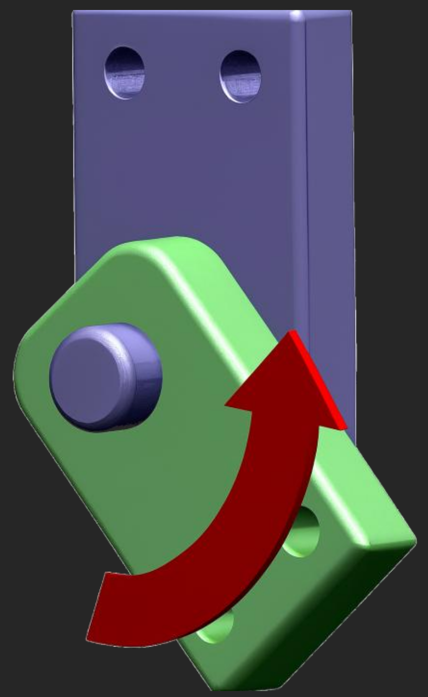
$\text{ApplyLinearVelocityCorrection}(\mathbf{p}_1, \mathbf{p}_2, \frac{f}{\Delta t}\mathbf{a})$



Apply a Motor Torque

ApplyTorque(τ)

ApplyAngularVelocityCorrection$(\frac{\tau}{\Delta t}\mathbf{a})$



See you in the next tutorial…