力覚計算と更新周期

電気通信大学知能機械工学科
長谷川晶一

必要な更新周期は？

1kHz for 70kN/m,

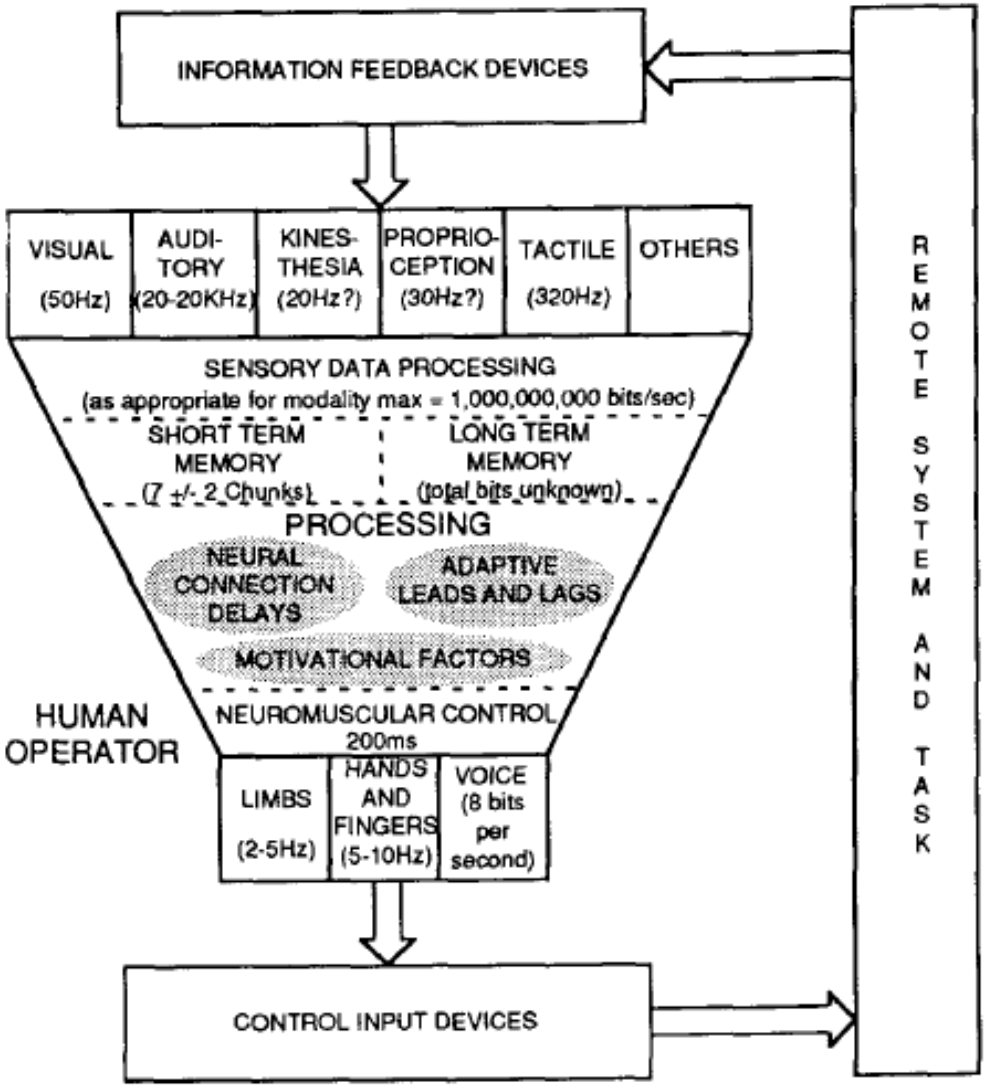
100Hz for 6.7kN/m

[Love and Book 1995]

1kHz:　1kg重 で 0.14mmの侵入

100Hz: 1kg重 で 1.5mmの侵入

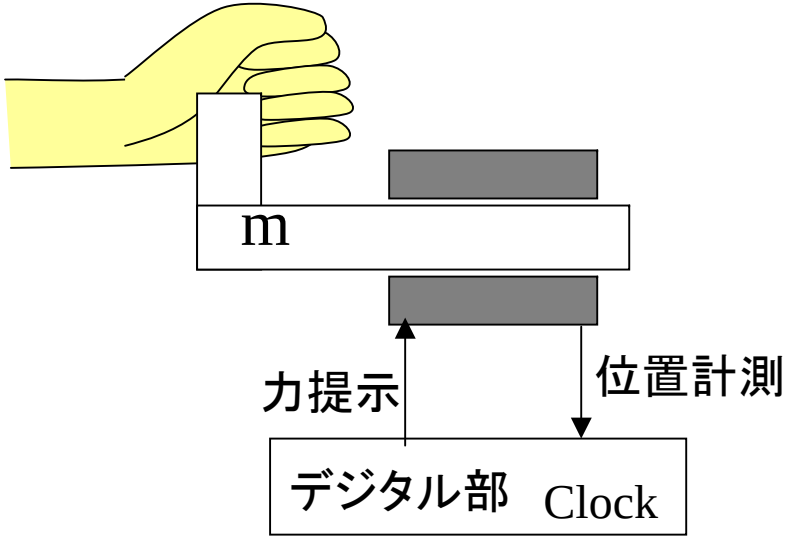
私の感覚@SPIDARとはちょっと合わない。
おそらく、機構の粘性や摩擦が大きい。



[T.L.Brooks 1990]

安定になる更新周期 [J.E.Colgate and G.Schenkel 1997] を紹介

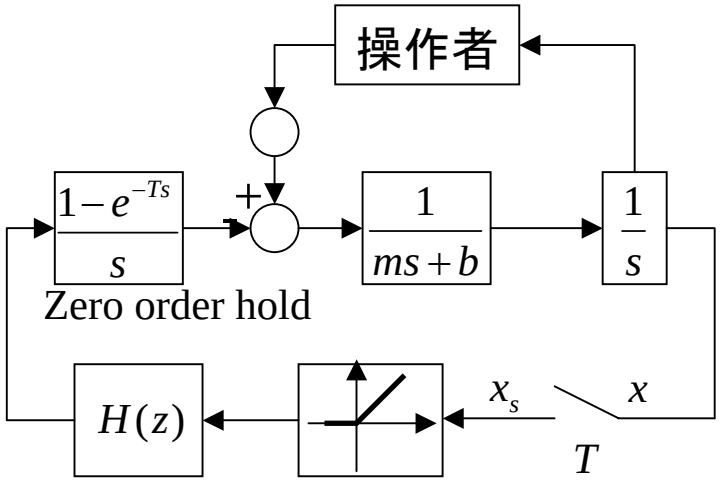
- 1自由度
- 位置計測・力出力



デジタル部分

- $H(z) = K + B\frac{z-1}{Tz}$
- T: 制御周期
- K:バーチャル壁のバネK
- B:ダンパ B

- 実デバイス部分
- m: デバイスの質量
- b: デバイスの粘性



安定になる更新周期

- 証明が論文　J.E.Colgate and G.Schenkel:
  Passivity of a Class of Sampled-Data Systems: Application to Haptic Interfaces 1997,
  Journal of Robotic Systems 14(1), 37-47 (1997)

にありますが、 $b > \frac{T}{2}\frac{1}{1-\cos\omega T}\mathrm{Re}\{(1-e^{-j\omega T})H(e^{j\omega T})\}$

$$0 \le \omega \le \omega_N = \pi/T$$

$$b > \frac{KT}{2} - B\cos\omega T \quad 0 \le \omega \le \omega_N$$

$$b > \frac{KT}{2} + B$$

と出てきます。

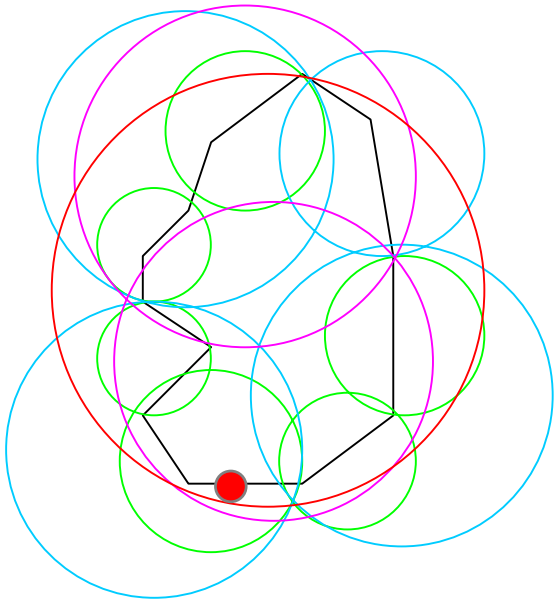
  - 機構にダンパ性 b が必須。
  - Tが大きいとKを大きく出来ない。
  - ところで、B（壁のダンパ）は逆効果？
    ⇒ 普通、b は高域ほど大きいので、効果がある
    （$\omega < \omega_N$ が小さいときは、$b > \frac{KT}{2} - B$ に近くなる）
- 1kHzは制御のため（だけ）に必要

目次

- 1kHzを実現するための力覚計算
  - 衝突判定の高速化
    - Bounding Volumeの階層化による高速化
  - 力覚ポインタが形状を持つ場合の計算
    - Point Cloud モデル
    - 凸多面体モデル
- 低速な物理シミュレータとの接続
  - 1ステップが低速な物理シミュレータを利用するには
    - 拘束解消法（LCPなど）の物理シミュレータとの接続法

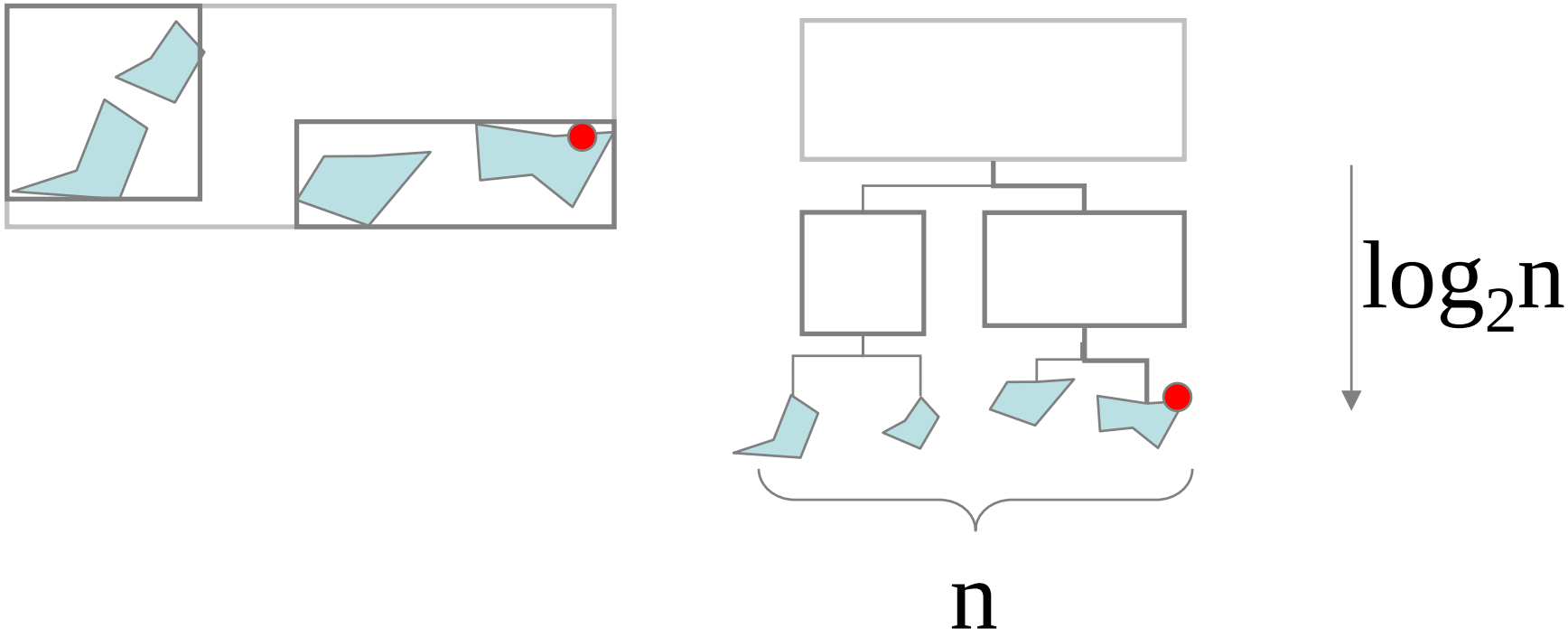
Bounding Volumeの階層化による高速化

- 階層的にBouding Volume
  （Ruspini 97では球）[Ruspini et.al. SIGGAPH 97]
  でポリゴンを覆う。
  - ポリゴン数が多いときに有効
  - 力覚ポインタの接触判定は、
    まず、一番外側の球（赤）と判定。
    もし当たっていたら、紫と判定。
    紫と当たっていたら…最後に
    ポリゴン(黒)と判定



Bouding Volume自体の階層化

- 物体数が多い場合に有効
- 物体数nに対して、$\log_2 n$ 回の判定で済む

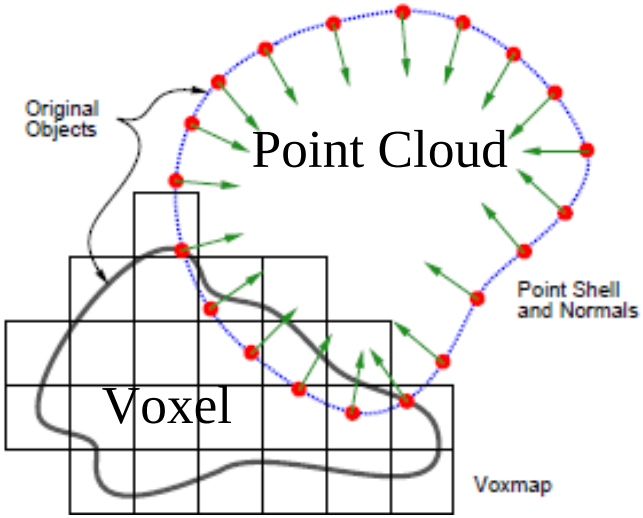


6自由度力覚レンダリング

- 3自由度力覚レンダリング
  - 力覚ポインタは点
  - 接触対象の形状とポインタの点の距離を計算
- 6自由度力覚レンダリング
  - 力覚ポインタが形状を持つ
  - 接触対象の形状とポインタの形状の距離を計算

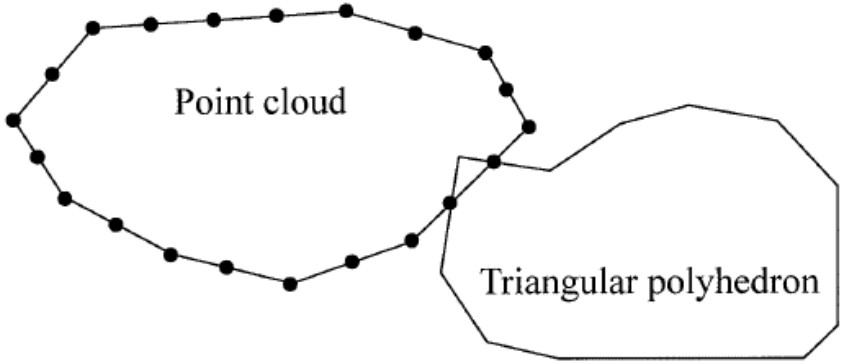
力覚ポインタにPoint Cloudを用いる方法

- 力覚ポインタをPoint Cloud、静止物体をVolxelやポリゴンでモデル化
- 判定が単純で高速



[McNeely et.al. SIGGAPH 99]

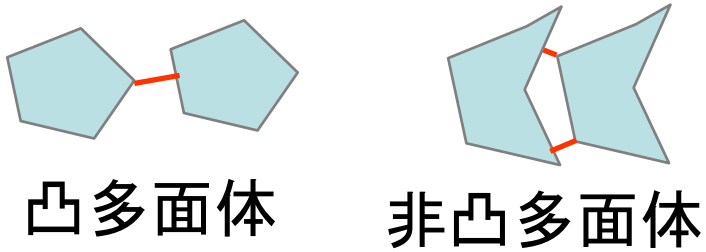
- 欠点
  - 特殊なモデルが必要
  - 静止物体が多い場合専用



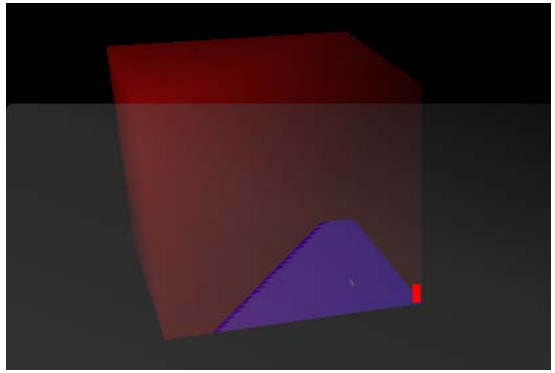
[足立ら JRSJ 03]

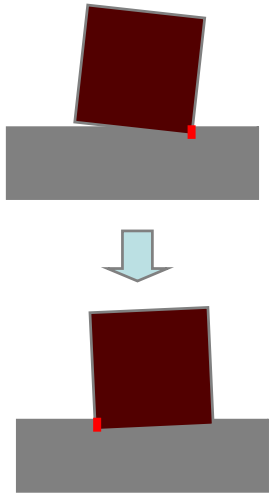
形状に、凸多面体を用いる方法

- 凸多面体



- 最侵入点でのペナルティ法はだめ

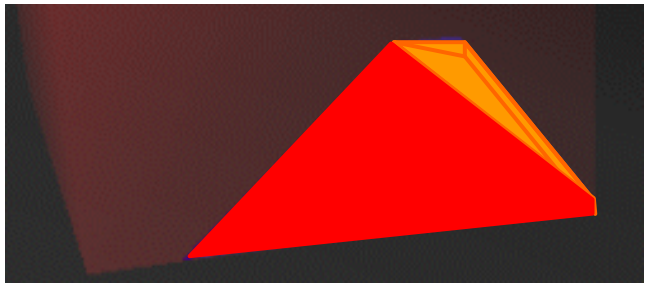




- 体積を計算すればOK

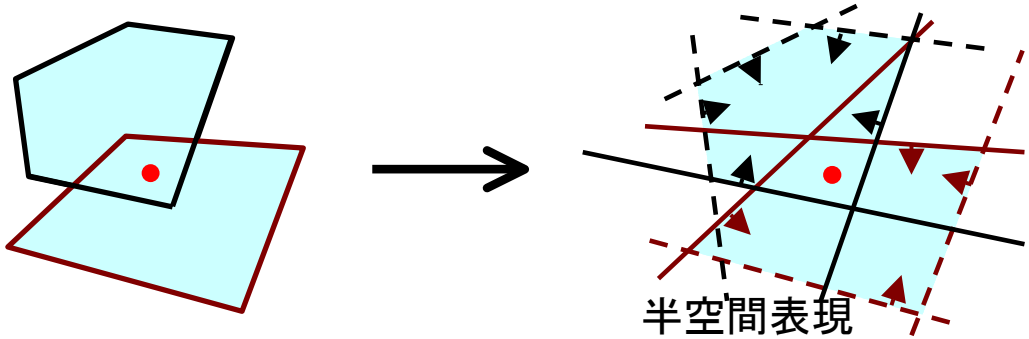
高速計算法

1. 物体は凸多角形でモデル化
2. GJKで共有点を1つ見つける
3. [Muller and Preparata 1978]で
   共有部分の凸多角形を高速計算

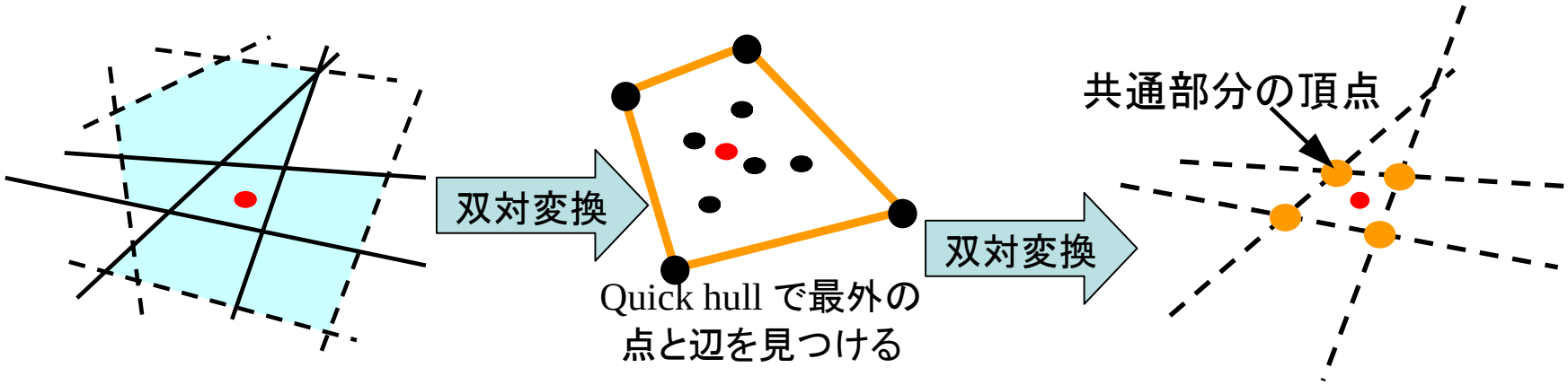


[Muller and Preparata 1978] を簡単に紹介

- 2つの凸多面体の共通部分は、半空間表現のアンド



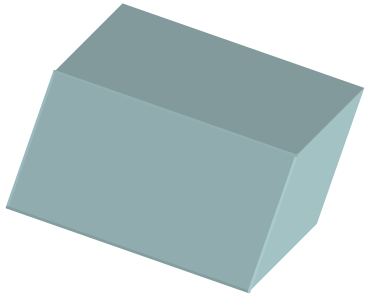
- 双対変換とQuick Hullで欲しい情報を取り出す



- 双対変換とは？
  - 距離を逆数に
    点を線に、線を点に

物理シミュレーションと力覚レンダリング

■ 剛体の運動



v: 速度
ω:角速度

m: 質量
I: 慣性テンソル
f: 外力

運動方程式

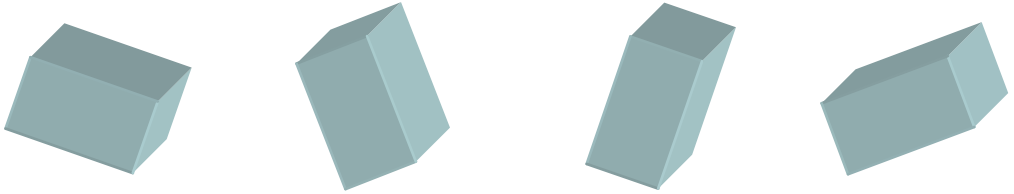
$$m\dot{\mathbf{v}} = \mathbf{f}$$

$$\mathbf{I}\dot{\boldsymbol{\omega}} = \boldsymbol{\tau}$$

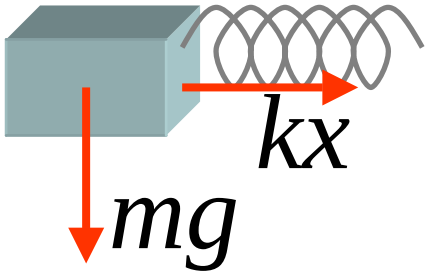
$$\mathbf{v}(t+\Delta t) = \mathbf{v}(t) + \mathbf{f}\Delta t / m$$

$$\mathbf{I}(t+\Delta t)\boldsymbol{\omega}(t+\Delta t) = \mathbf{I}(t)\boldsymbol{\omega}(t) + \boldsymbol{\tau}\Delta t$$

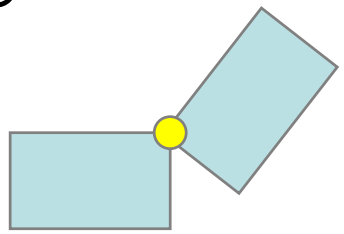
$\mathbf{f} = 0, \boldsymbol{\tau} = 0$ ならば，速度一定・角運動量一定



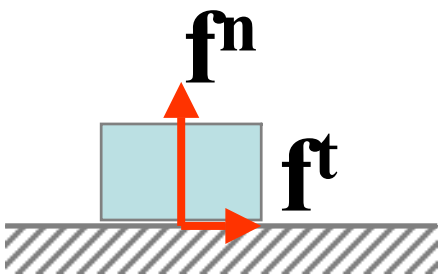
剛体に働く力



- 重力→ f=mg… 定数
- バネ→ f=kx… 位置に比例
- 拘束力
  - 力の大きさは不明
  - 剛体同士の位置・速度関係が決まっている
    - 蝶番：2物体の相対位置が一定
    - 抗力：2物体が互いに侵入しない
    - 静止摩擦力：物体が滑らない

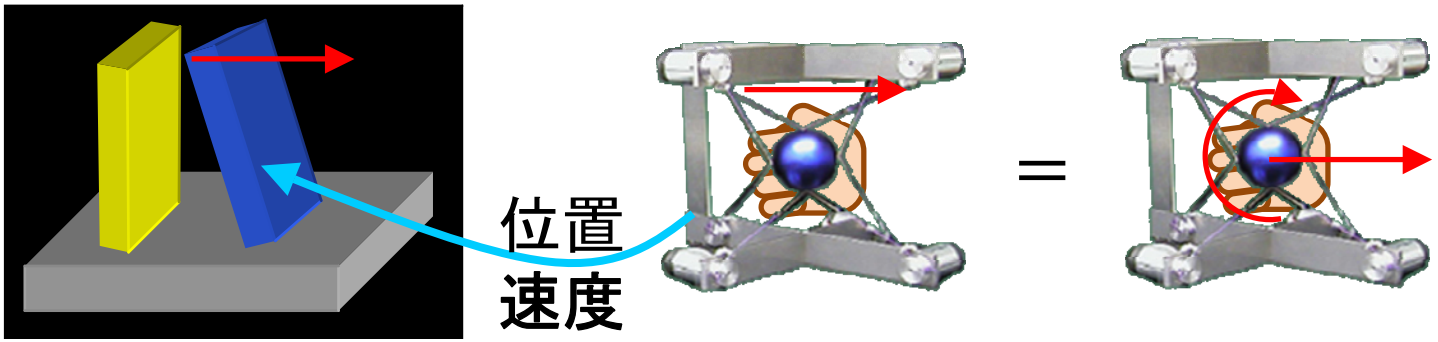


- 拘束力（接触力も拘束力の一種）の計算が難しい
  ⇒ 物理シミュレーションの主な仕事

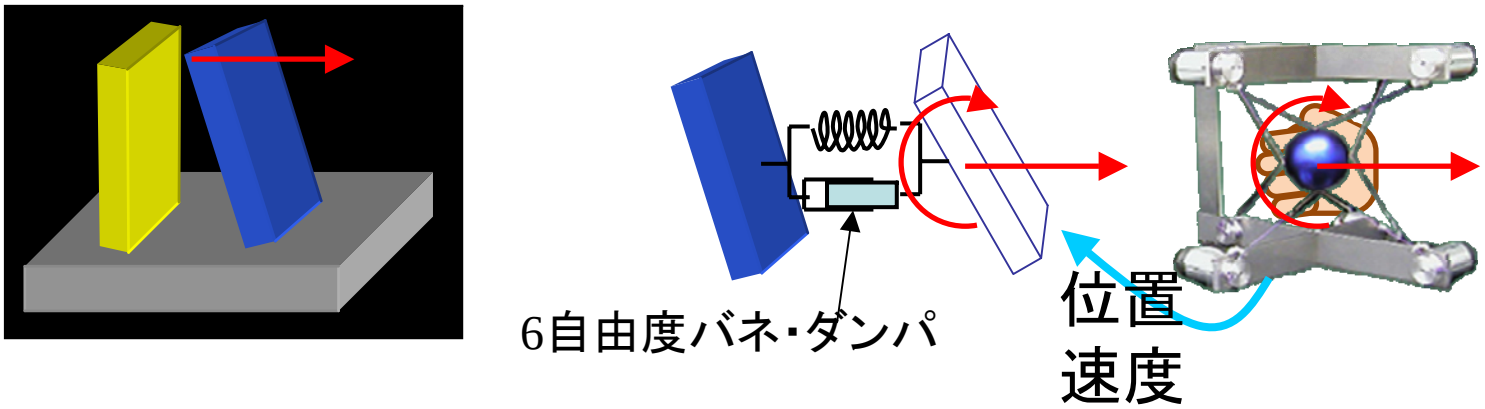


物理シミュレータと力覚レンダリングの関係

- 物理シミュレータ ⇒ 任意形状同士の接触力を計算
  ⇒ 6自由度力覚レンダリング
  - シミュレータ内で加わった接触力をそのまま提示



  - バーチャルカップリング：バネダンパを仮定



1kHzは本当に必要か？

- 制御には必要

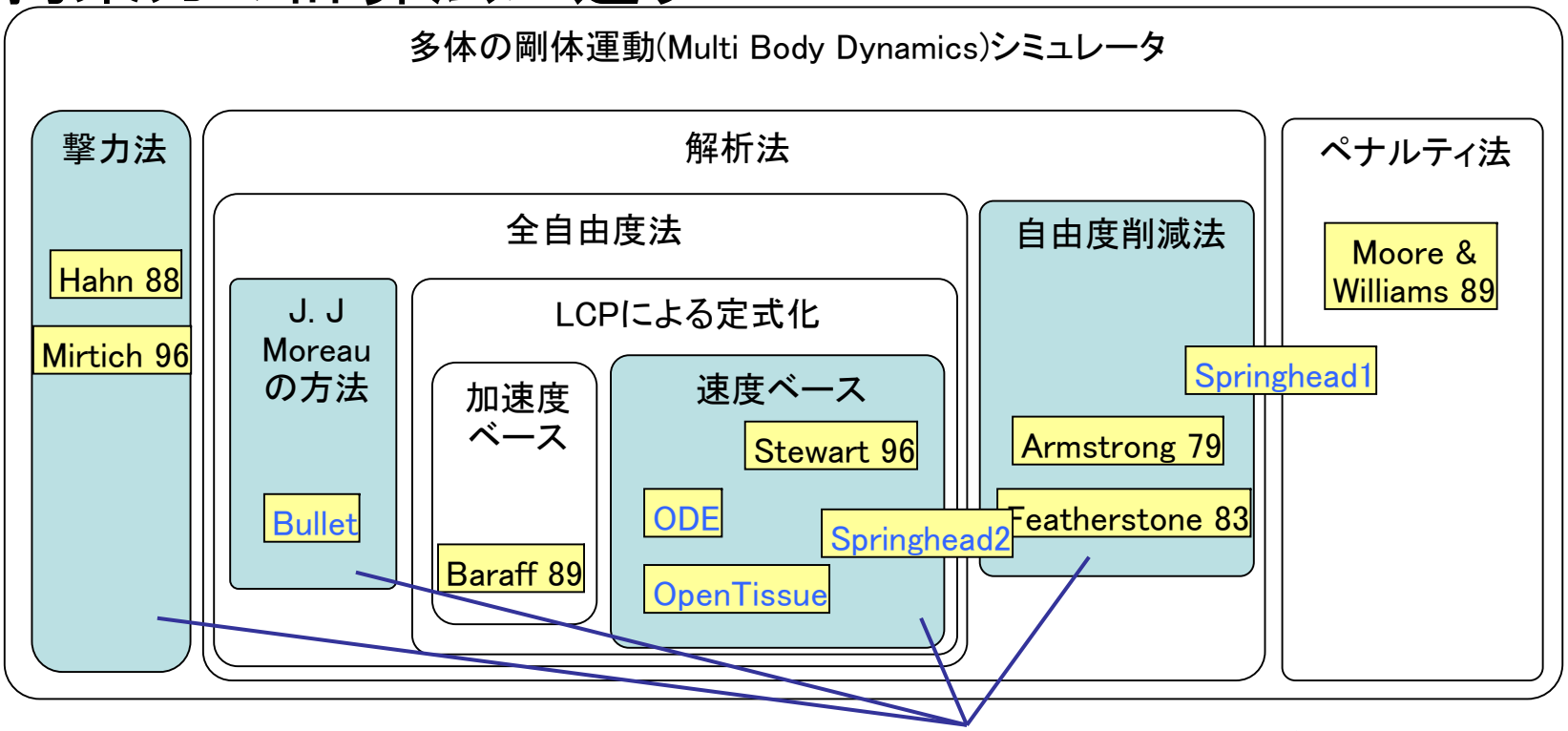
- 人の感覚としては必要ない
  - テレオペレーションシステムで、スレーブロボットは10~20Hzの帯域幅でよい
  - スレーブからマスターへのフィードバックは20~320Hzあると良い

[T.L.Brooks 90]

- 人間には、1kHzのループは不要なのでは？

物理シミュレータの種類

- 拘束力の計算法が違う

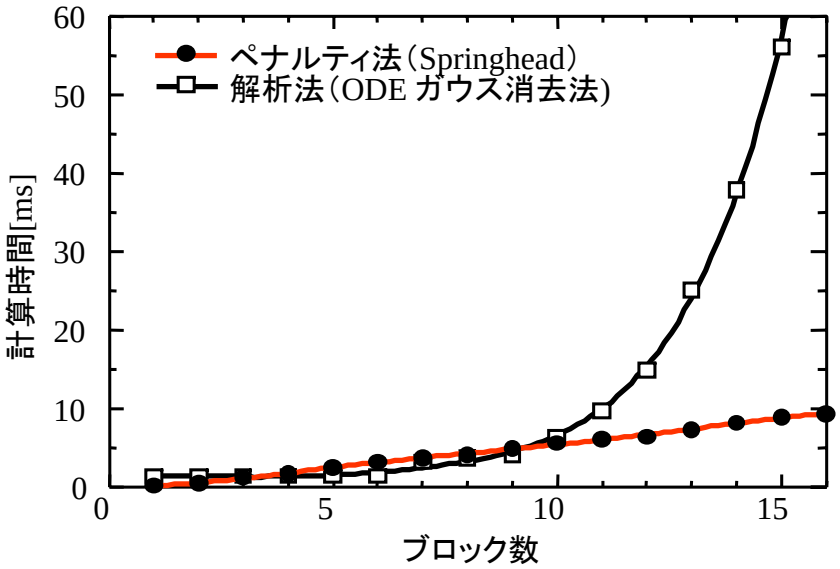


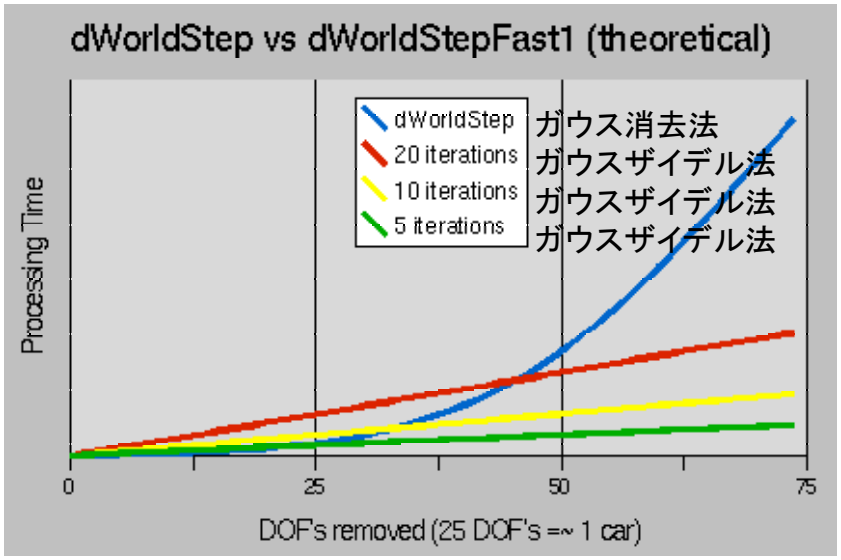
後退積分(が使いやすい)

- 速度ベース・加速度ベース:拘束条件の書き方
  - 速度ベース：侵入しない＝相対速度 ≧ 0
  - 加速度ベース：侵入しない＝相対加速度 ≧ 0

更新周期と計算量

- 解析法のシミュレータ
  - 後退積分だから安定 ⇒ Δtを大きくできる
  - シミュレーション1ステップには時間がかかる。
    ?
- 処理時間の比較

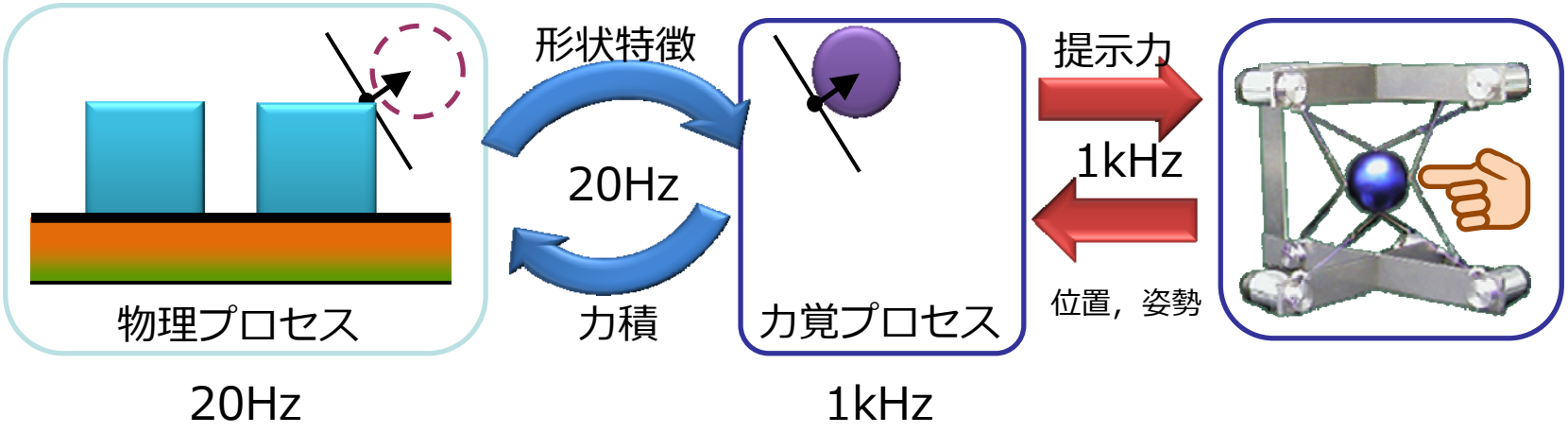




- ガウスザイデル法(近似解法)を使うとそこまで遅くない

ポインタ付近だけ1kHz

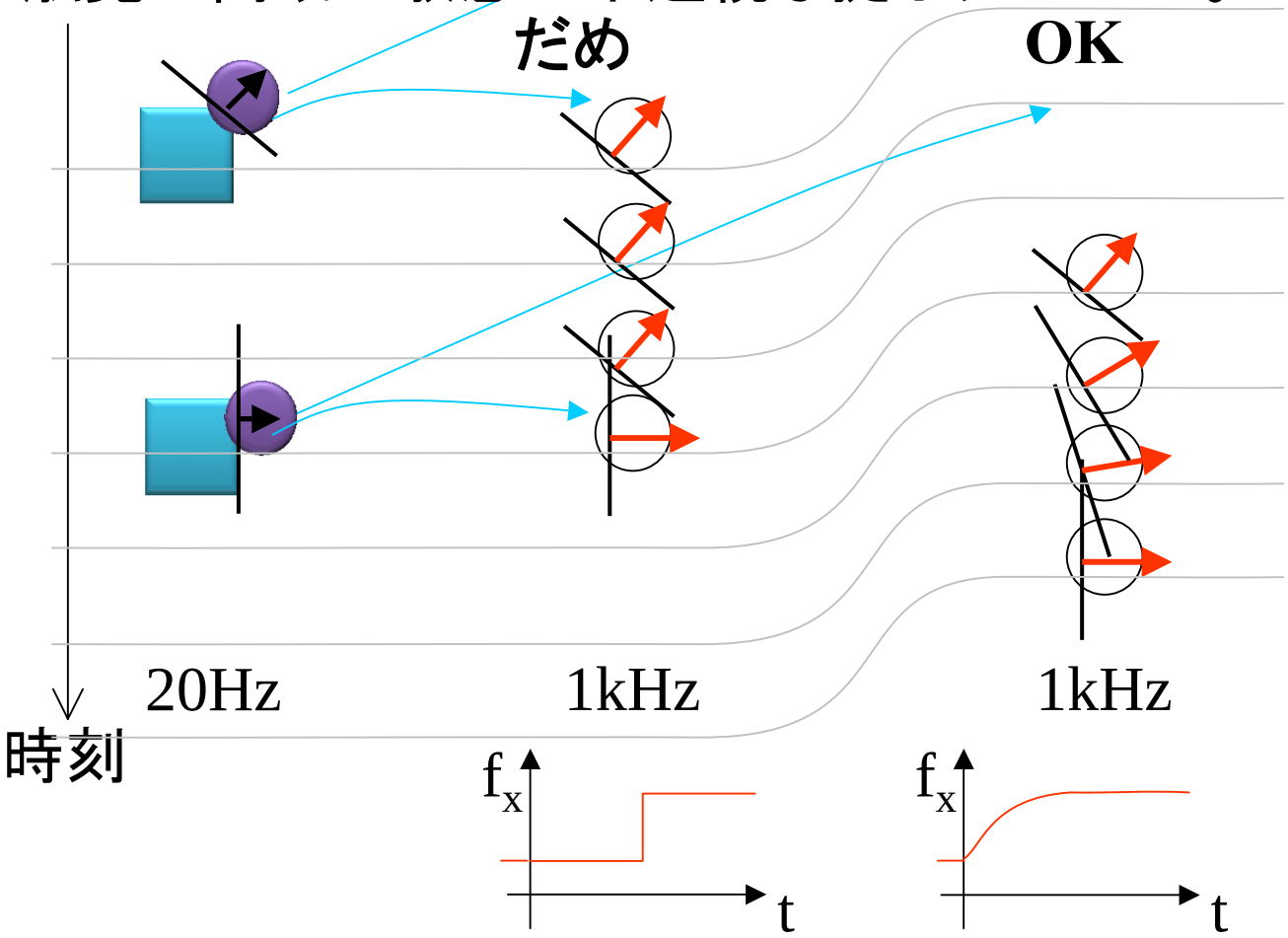
- 物理は20Hz　力覚は1kHz
- 力覚プロセスで何をすればよいか？



[Adacheら 1995：“Intermediate Representation for Stiff Virtual Objects”]

力覚プロセスの処理での注意

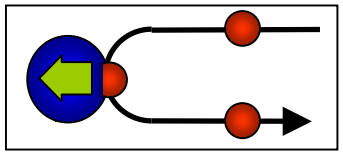
- 提示力は連続に
  - 力触覚は高域で敏感⇒不連続な提示力はだめ。



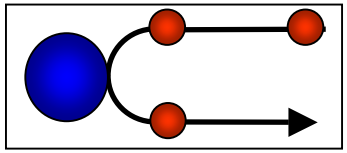
力覚プロセスの処理での注意点

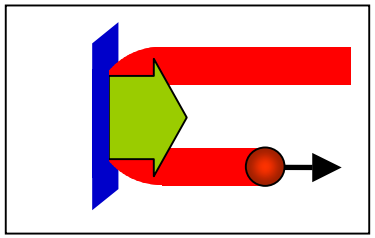
- 物理プロセスで接触判定をするとだめ

20Hz（運が良い場合）



20Hz（間が悪いと）



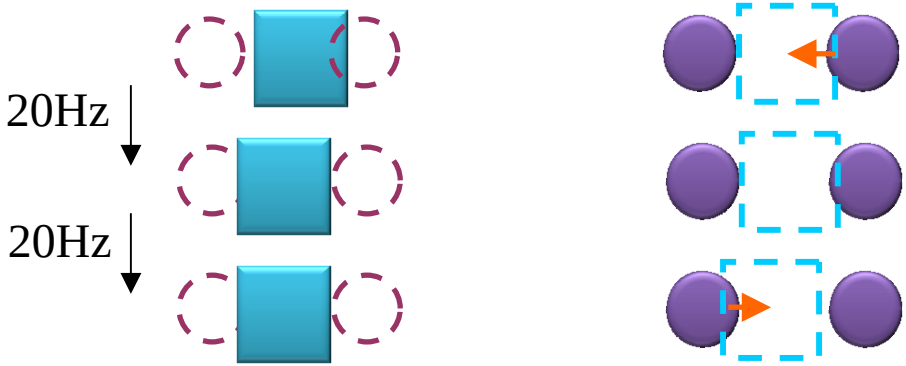


1kHz（どちらも同じ）

  - 1kHzで力積を計算して、20Hz（物理シミュ）に返せばOK

[長谷川ら 1999]

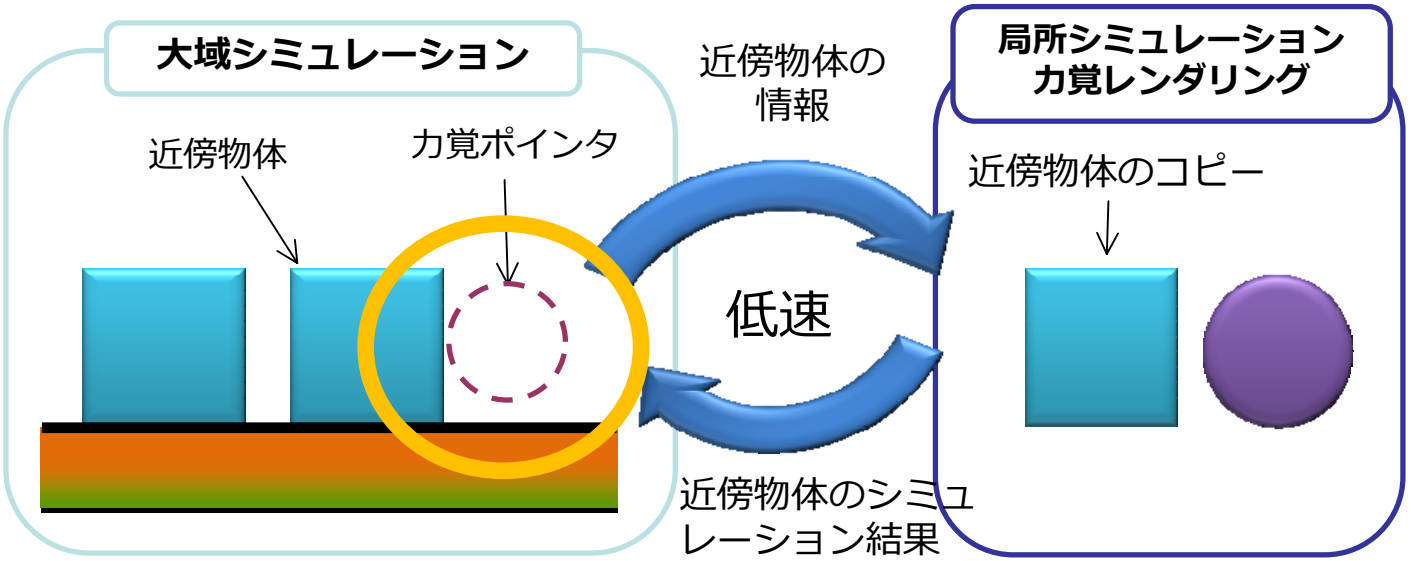
- 制御のループに注意：把持が発振する



  - 力が遅れて伝わる ⇒ 発振

力覚プロセスの処理

- 遅れをなくすには？
  - 力覚ポインタ近傍のシミュレーションも行う



[技術展示：T4

口頭発表：さきほど 2A2-1

大規模バーチャル世界のための高品質力覚提示]

力覚レンダリングでの工夫

- イベントベース：制御しないのも手です
  - フィードバック制御 ⇒ 発振
    オープンループで提示：Event-based haptic
    - 接触した瞬間に撃力を加える
      Braking force pulse
      [S.E.Salcudean and T.D.Vlaar 1994]
    - 接触時の振動を再現する
      [P. Wellman and R. D. Howe 1995]

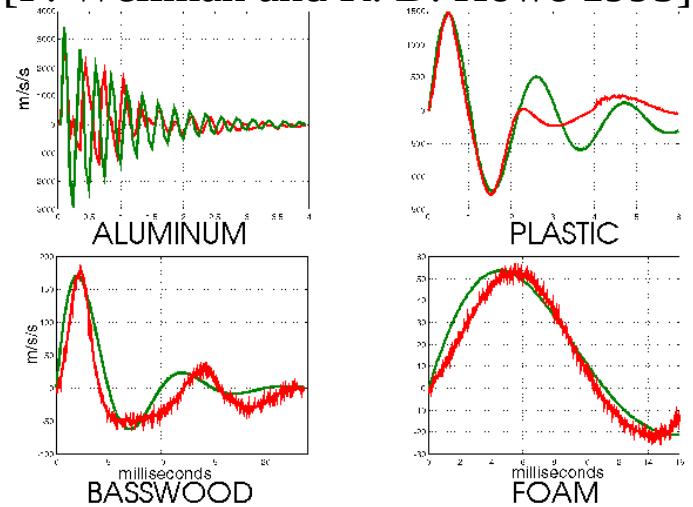
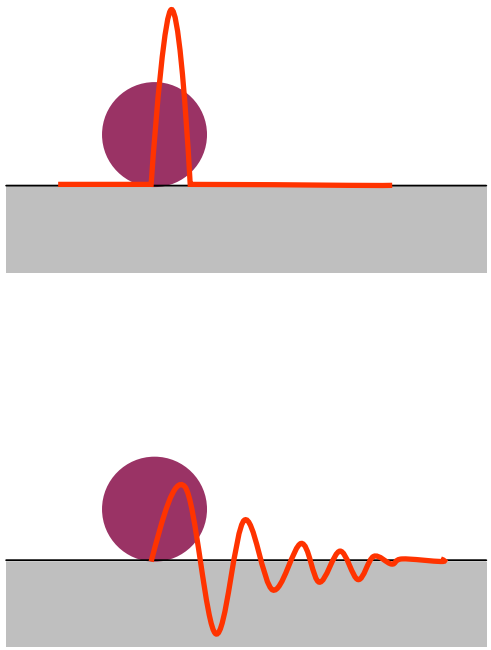


Figure 4 - Typical waveforms played during virtual reality testing (green line) and a recorded waveform at 0.289 m/s impact velocity (red line).



力覚インタフェースで再現

まとめ

- 安定な制御のために、1kHz以上の更新周期が必要
  - 人間の感覚のために必要なわけではない
    - 物理シミュレータ自体は20Hz~50Hzで十分。
    - ただし、力触覚は高域で敏感⇒不連続な提示力はだめ。
    - 把持の問題：制御のループに注意
    - Event-based haptic ＝ 制御しないのも手
- 物理シミュレーションは遅くても良い
  - 剛体シミュレーションは、後退積分が可能な、解析法のシミュレータで、Δtを大きくすると高速
  - 力覚プロセスでの処理に工夫が必要
    - 提示力を連続に
    - 遅れのあるループが出来ていないか ⇒ 把持の問題
    - 近傍だけを高速シミュレーションすれば、遅れません