



KYUSHU UNIVERSITY 100th 2011  
知の新世紀を拓く

# 力学系の制御とシミュレーションに おける 微分包含式の取り扱いについて

九州大学大学院  
工学研究院機械工学部門  
准教授

きく    うえ    りょう  
菊   植   亮

<http://rk.mech.kyushu-u.ac.jp/>

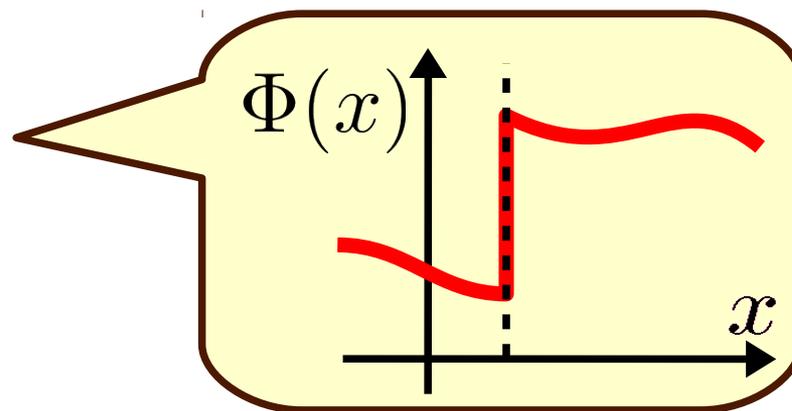


九州大学

# 微分包含式(Differential Inclusions)

## ◆ 不連続性を含む微分方程式

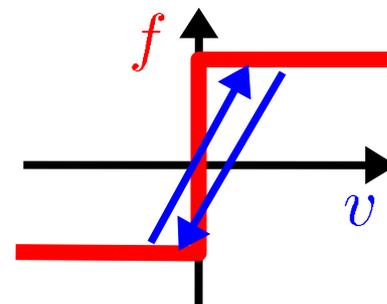
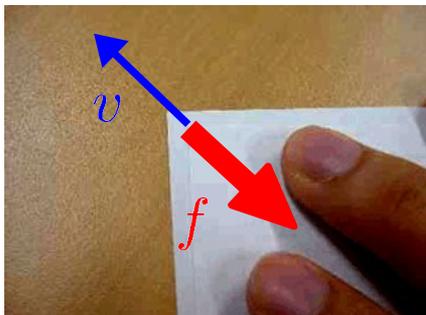
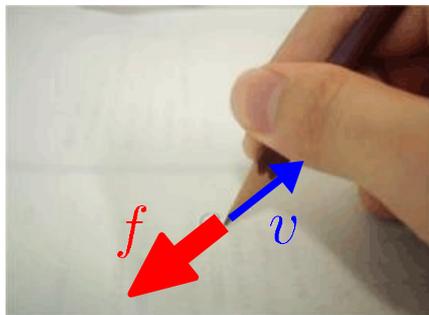
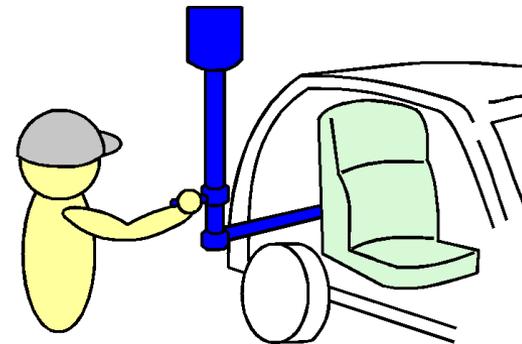
$$\dot{x} \in \Phi(x)$$



- 微分包含式の性質
- 数値計算のための便利な公式
- 陰的積分
- 「微分代数緩和」
- スライディングモード制御と「微分代数緩和」
- 連続時間領域での解析

# 動機：自動車産業からのニーズ (2003年)

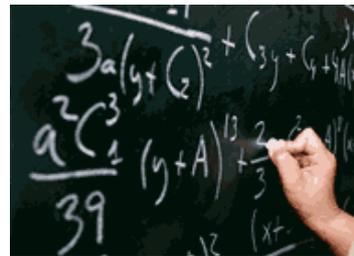
- ◆【ニーズ】重量物の精密位置決め作業をアシストするロボットが欲しい。
- ◆【課題】どのように反力を与えると位置決め効率がアップするか？
- ◆【アイデア】ロボットで擬似的な**摩擦力**を発生すればよいのでは？
- ◆【問題】制御で**摩擦力**を作り出すのが難しい。
- ◆⇒ **不連続性**と**微分包含式**に興味がる。



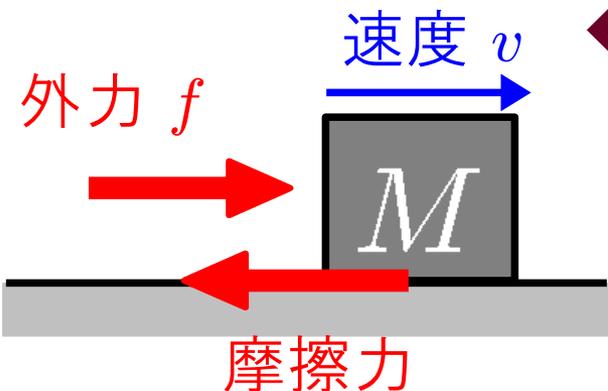


[1]

# 微分包含式 (Differential Inclusions)



# クーロン摩擦力

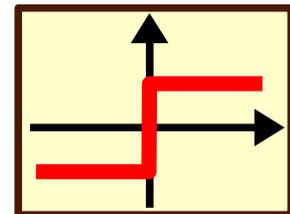


- ◆ 動摩擦では速度と反対向き
- ◆ 静止摩擦状態では外力と釣り合う

$$M\dot{v} \in -F \operatorname{sgn}(v) + f$$

ただし

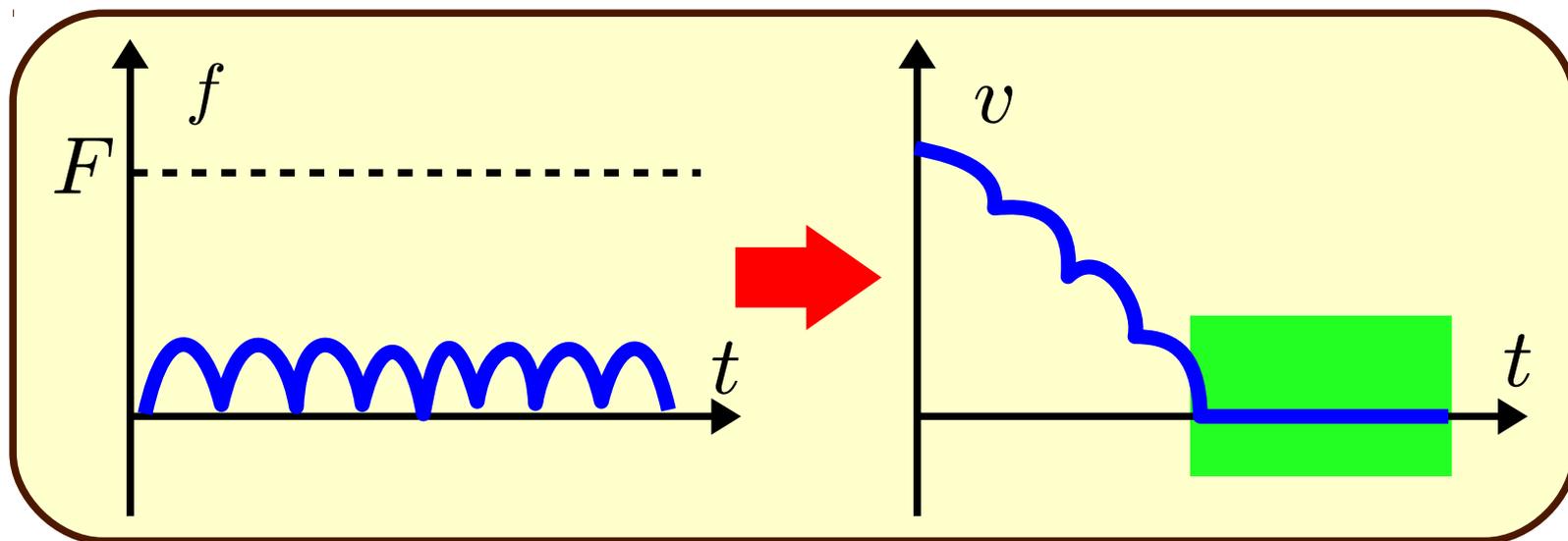
$$\operatorname{sgn}(v) \triangleq \begin{cases} 1 & \text{if } v > 0 \\ [-1, 1] & \text{if } v = 0 \\ -1 & \text{if } v < 0 \end{cases} =$$



- ◆ 上式のFilippov解:  $v = 0$  かつ  $|f| < F$  のとき,  
 $v = 0$  が維持されるように左辺が選択される。  
⇒ 「スライディングモード」という状態。
  - 物理的にスライドしていない状態(静止摩擦状態)が、  
数学的には「スライディングモード」!

# 微分包含式の「解析的な」性質

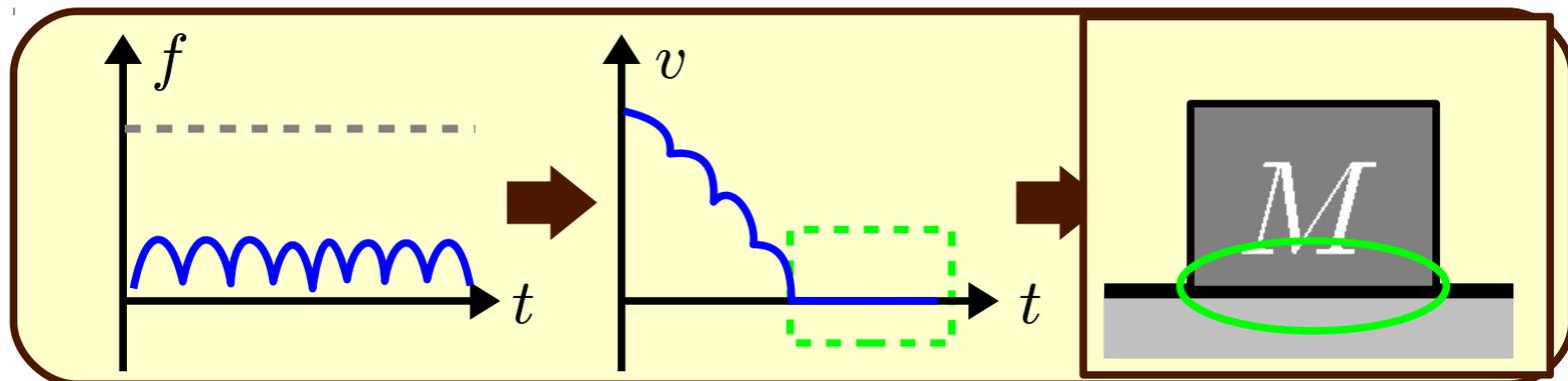
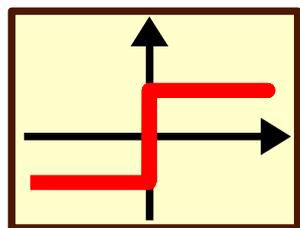
$$M\dot{v} \in -F\text{sgn}(v) + f$$



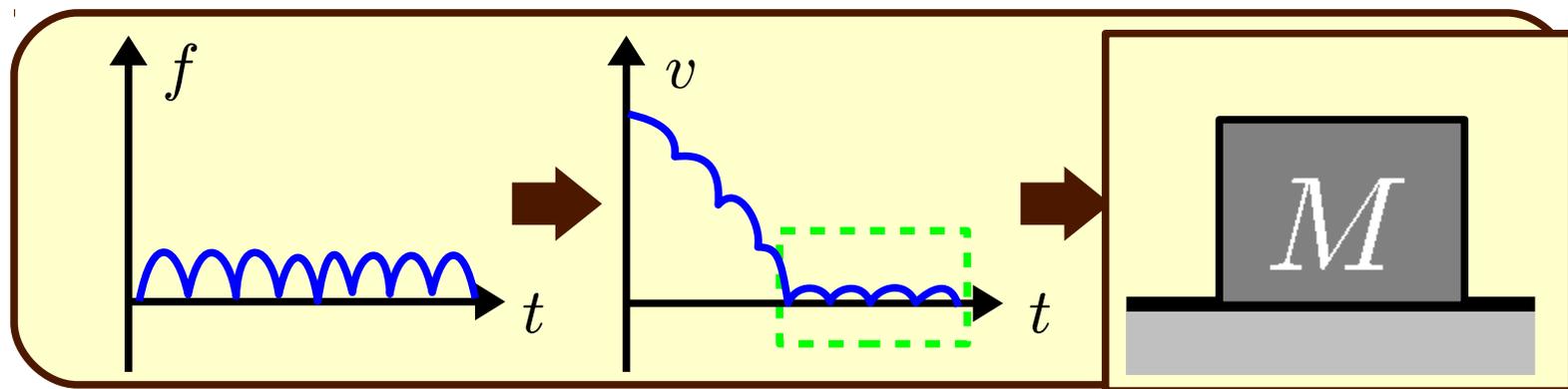
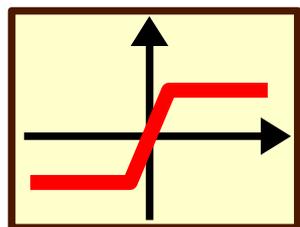
- ◆  $\text{sgn}$ 内が零 ( $v = 0$ ) で  $|f| < F$  のとき,  
系は  $f$  の影響を受けない.  
 ⇒ これは解析的な性質. 無限に高速な切替が必要.  
 ⇒ 数値計算では再現しにくい.

# 数値計算⇒単純な平滑化はダメ

$$M\dot{v} \in -F \operatorname{sgn}(v) + f$$



~~X~~  $\cdot \|\cdot\|$



- ◆ シミュレーションや制御のためには、この解析的性質を温存できる数値計算手法が必要.



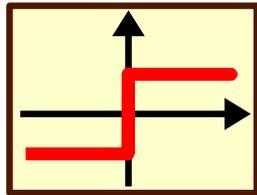
# [2] 便利な公式

# 例題:「代数包含式」

◆ 問1: 下式を  $x$  について解け.

$$x \in \text{sgn}(a - x) \cdots (1)$$

(符号関数)

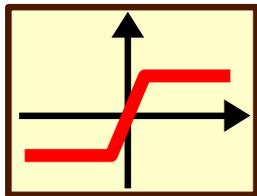
ただし,  $\text{sgn}(z) \triangleq$    $= \begin{cases} 1 & \text{if } z > 0 \\ [-1, 1] & \text{if } z = 0 \\ -1 & \text{if } z < 0 \end{cases}$

◆ 解答:

$$(1) \iff \begin{aligned} &(x = 1 \wedge a - x > 0) \\ &\vee (-1 \leq x \leq 1 \wedge a - x = 0) \\ &\vee (x = -1 \wedge a - x < 0) \end{aligned}$$

$$\iff x = \text{sat}(a)$$

(単位飽和関数)

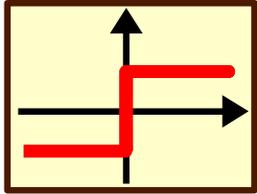
ただし,  $\text{sat}(z) \triangleq$    $= \begin{cases} 1 & \text{if } z > 1 \\ z & \text{if } |z| \leq 1 \\ -1 & \text{if } z < -1 \end{cases}$

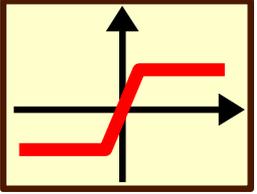
# 定理1

[Kikuuwe et al.: IEEE-TRO, 2006]

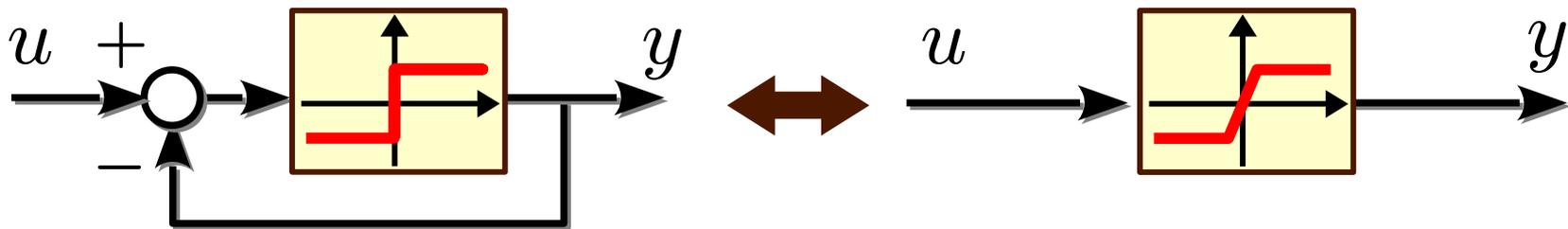
[Kikuuwe et al.: ICRA 2006]

$$y \in \text{sgn}(u - y) \iff y = \text{sat}(u)$$

ただしここで,  $\text{sgn}(x) \triangleq$    $= \begin{cases} 1 & \text{if } x > 0 \\ [-1, 1] & \text{if } x = 0 \\ -1 & \text{if } x < 0 \end{cases}$

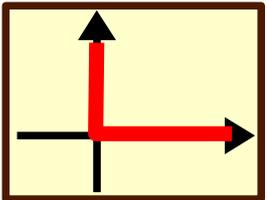
$\text{sat}(x) \triangleq$    $= \begin{cases} 1 & \text{if } x > 1 \\ x & \text{if } |x| \leq 1 \\ -1 & \text{if } x < -1 \end{cases}$

◆ つまり, **代数ループ**で不連続性を除去できる.



# 定理2

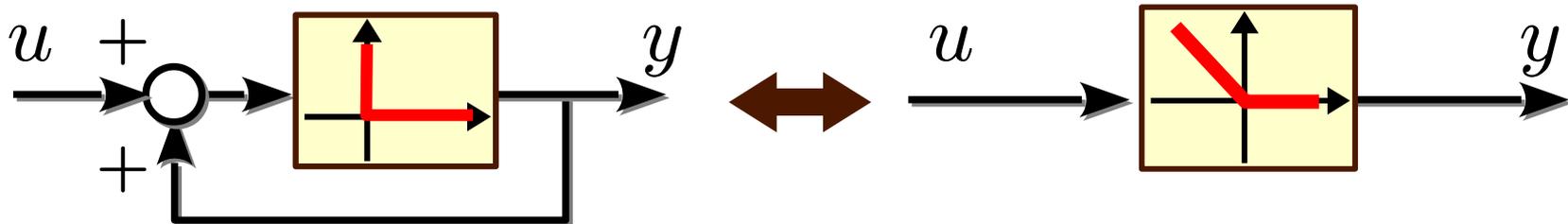
$$y \in \text{dio}(u + y) \iff y = \max(-u, 0)$$

ただしここで,  $\text{dio}(x) \triangleq$    $= \begin{cases} 0 & \text{if } x > 0 \\ [0, \infty) & \text{if } x = 0 \\ \emptyset & \text{if } x < 0 \end{cases}$

(理想ダイオード関数)

■ この関数は, 片側拘束の表現に便利.

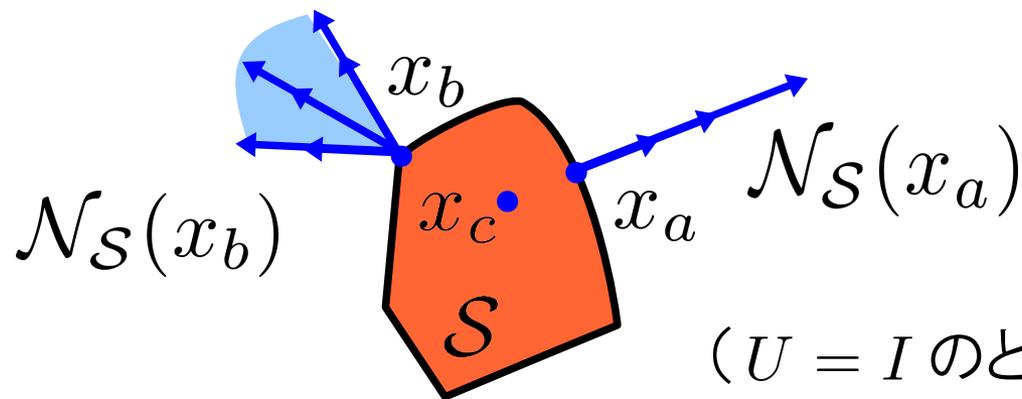
◆ これも, 代数ループで不連続性を除去できる.



# 上位概念：法錐 (Normal Cone)

◆ 点  $x$  における集合  $S$  のメトリック  $U$  での法錐

$$\mathcal{N}_S^U(x) \triangleq \begin{cases} \{\xi \in \mathbb{R}^n \mid \xi^T U(x^* - x) \leq 0 \ \forall x^* \in S\} & \text{if } x \in S \\ \emptyset & \text{if } x \notin S \end{cases}$$



- ◆  $x$  が集合  $S$  の表面  $\Rightarrow$  法錐は法線ベクトルの集合
- ◆  $x$  が集合  $S$  の内部  $\Rightarrow$  法錐はゼロベクトル
- ◆  $x$  が集合  $S$  の外部  $\Rightarrow$  法錐は空集合

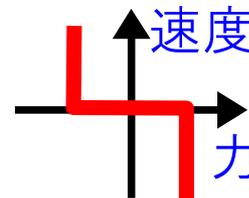
# 上位概念：法錐 (Normal Cone)

- ◆ 点  $x$  における集合  $S$  のメトリック  $U$  での法錐

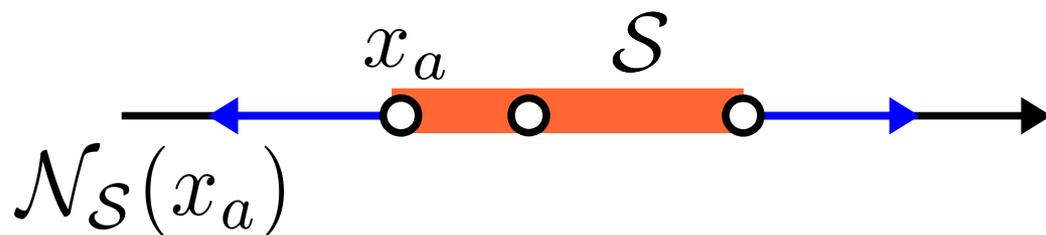
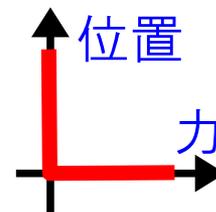
$$\mathcal{N}_S^U(x) \triangleq \begin{cases} \{\xi \in \mathbb{R}^n \mid \xi^T U(x^* - x) \leq 0 \ \forall x^* \in S\} & \text{if } x \in S \\ \emptyset & \text{if } x \notin S \end{cases}$$

- ◆ 符号関数とダイオード関数は，法錐の逆写像

$$f \in -F \text{sgn}(v) \iff -\mathcal{N}_{[-F, F]}(f) \ni v$$

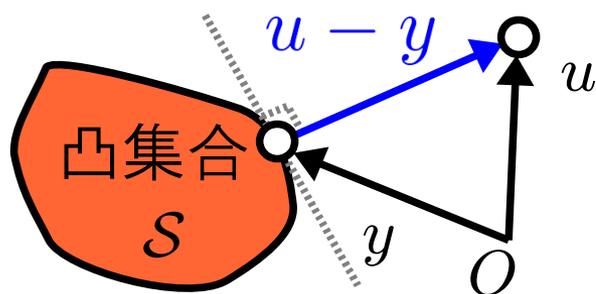


$$f \in \text{dio}(p) \iff -\mathcal{N}_{\mathbb{R}^+}(f) \ni p$$



# 法錐は射影と裏表一体の関係

$$\mathcal{N}_S(y) \ni u - y \iff y = \text{proj}_S(u)$$



◆ 「 $u$  から  $S$  に下ろした垂線の足が  $y$  」  
 という状況の二つの等価な表現

◆ 前述の定理は上記の関係の特殊ケース

$$S = [-1, 1] \implies y \in \text{sgn}(u - y) \iff y = \text{sat}(u)$$

$$S = [0, \infty) \implies y \in \text{dio}(u + y) \iff y = \max(-u, 0)$$

◆ 新しい知見ではないが、これらの単純な形式は前例なし

◆ これによって、様々な技術の導出が可能に

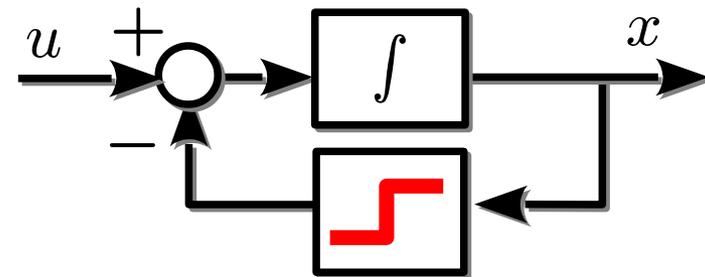


**[3]**

**技巧【1】:陰的積分**

# 微分包含式の数値積分

$$\dot{x} \in -\text{sgn}(x) + u$$



## ◆ 前進(陽的)オイラー法

$$\frac{x_k - x_{k-1}}{T} \in -\text{sgn}(x_{k-1}) - u_k$$

- 左辺によって右辺を決定

未知 既知

## ◆ 後退(陰的)オイラー法

$$\frac{x_k - x_{k-1}}{T} \in -\text{sgn}(x_k) - u_k$$

- 左辺と右辺の辻褃があうように  $x_k$  を決定

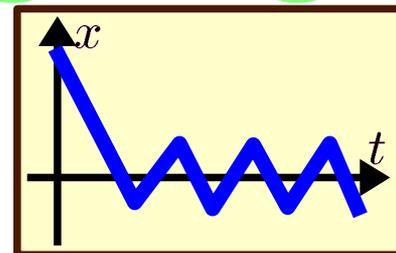
# 数値積分アルゴリズムの導出

未知 既知

◆ 前進オイラー法:  $\frac{x_k - x_{k-1}}{T} \in -\text{sgn}(x_{k-1}) - u_k$

$$\iff x_k \in x_{k-1} - T(\text{sgn}(x_{k-1}) + u_k)$$

➔  $x_k \leftarrow x_{k-1} - T(\text{sgn}(x_{k-1}) + u_k)$



◆ 後退オイラー法:  $\frac{x_k - x_{k-1}}{T} \in -\text{sgn}(x_k) - u_k$

$$\iff \frac{x_k - x_{k-1}}{T} + u_k \in \text{sgn} \left( -\frac{x_{k-1}}{T} + u_k - \left( \frac{x_k - x_{k-1}}{T} + u_k \right) \right)$$

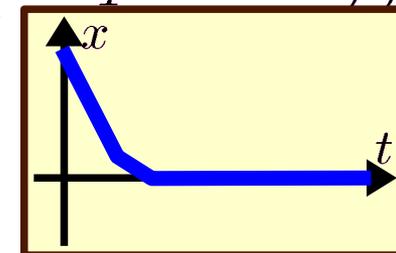
$$\iff \frac{x_k - x_{k-1}}{T} + u_k = \text{sat} \left( -\frac{x_{k-1}}{T} + u_k \right)$$

$$y \in \text{sgn}(u - y)$$

$$\iff y = \text{sat}(u)$$

$$\iff x_k = x_{k-1} - T u_k - T \text{sat} \left( \frac{x_{k-1} - T u_k}{T} \right)$$

➔  $x_k \leftarrow x_{k-1} - T u_k - T \text{sat} \left( \frac{x_{k-1} - T u_k}{T} \right)$



# 摩擦力を受ける質点のシミュレーション

## ◆ 連続時間表現(微分包含式)

[Kikuuwe et al.: IEEE-TRO, 2006]

$$M\dot{v} = h - f$$

$$f \in F\text{sgn}(v)$$

## ◆ 離散時間表現(代数包含式)

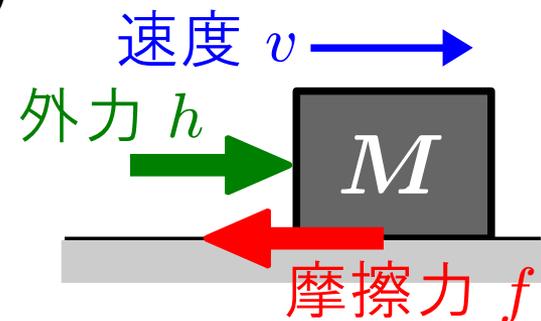
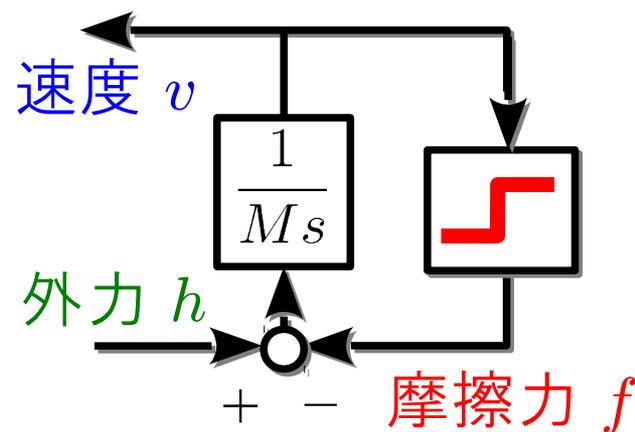
$$M(v_k - v_{k-1})/T = h_k - f_k$$

$$f_k \in F\text{sgn}(v_k)$$

## ◆ アルゴリズム(上式を解く計算手順)

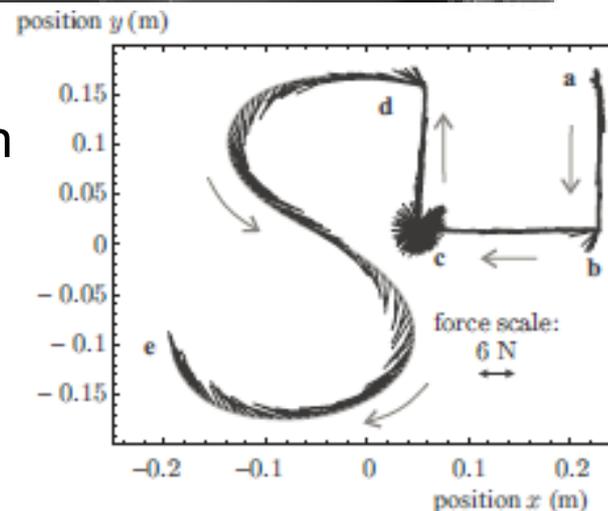
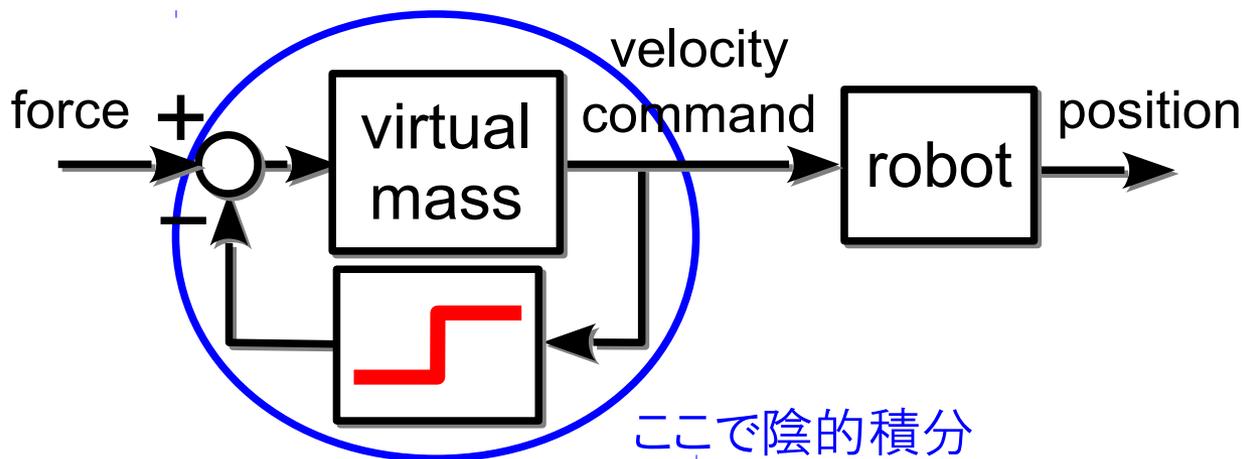
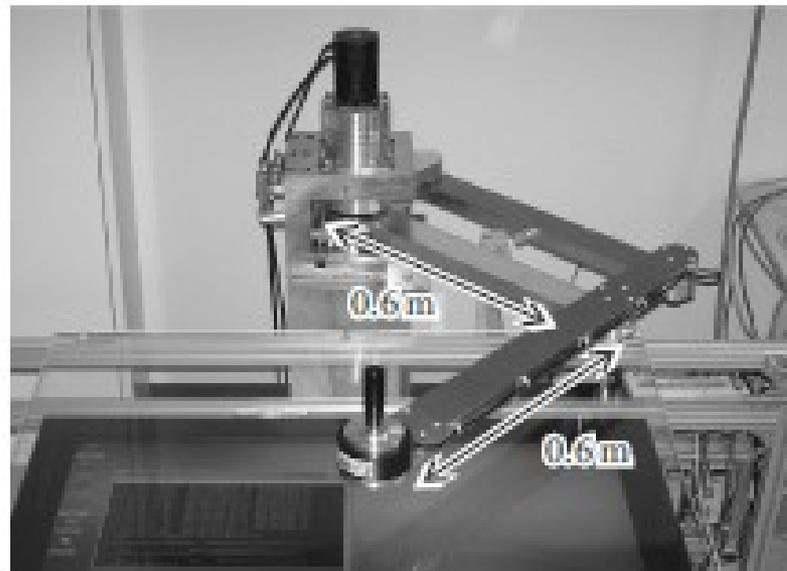
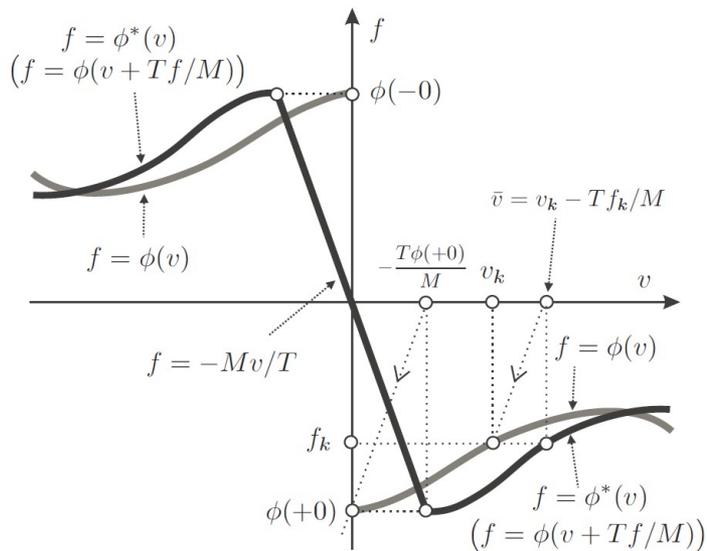
$$f_k := F\text{sat}((Mv_k/T + h_k)/F)$$

$$v_k := v_{k-1} + T(h_k - f_k)/M$$



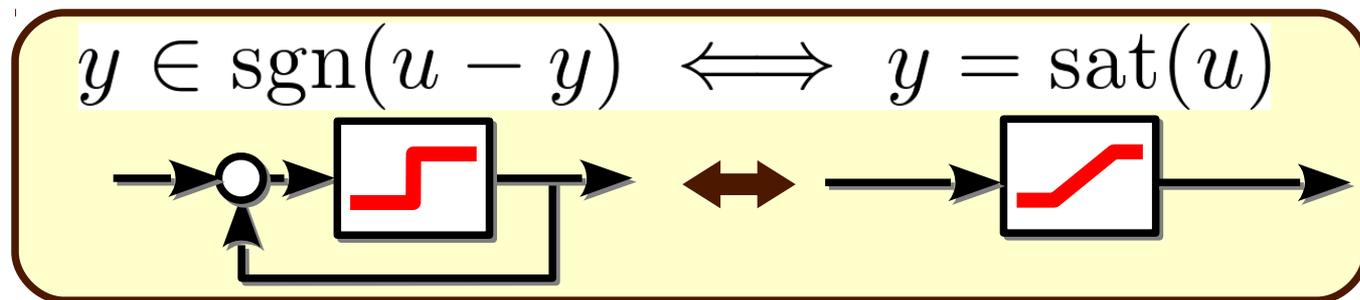
# クーロン摩擦を含むアドミッタンス制御

[Kikuuwe et al.: IROS2005]



# 業界の反応

- ◆ Acary & Brogliato: “Numerical Methods for Nonsmooth Dynamical Systems”, Springer, 2008.
- ◆ ... An interesting contribution is in **Kikuuwe et al. (2005)** where the implicit discretization of Coulomb friction is **rediscovered** and extended to more sophisticated multivalued nonsmooth models (equations (14a)-(14c) in Kikuuwe et al., 2005 exactly correspond to the procedure described in Fig. 1.17) ...



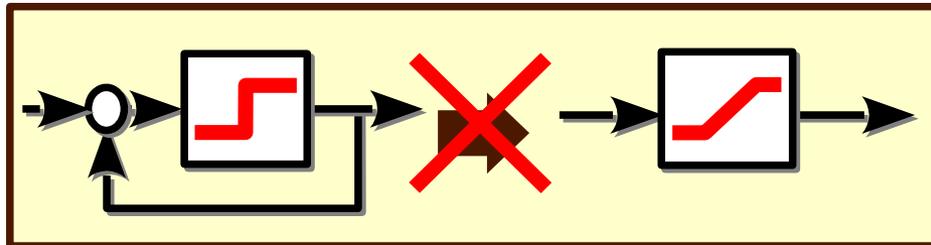
- ◆ 数学的には、それほど新しくない。結局は特殊ケース。
- ◆ しかし、Normal Coneなどの難しい概念を経由しないので単純。
- ◆ 「代数ループ」という解釈は制御屋特有？



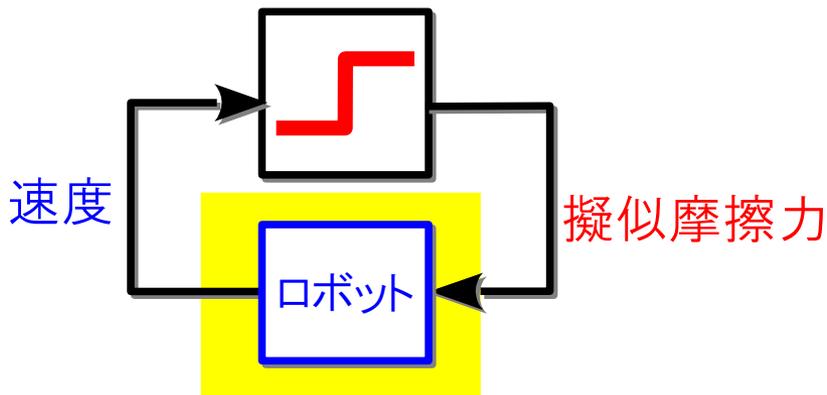
[4]

**技巧【2】:「微分代数緩和」**

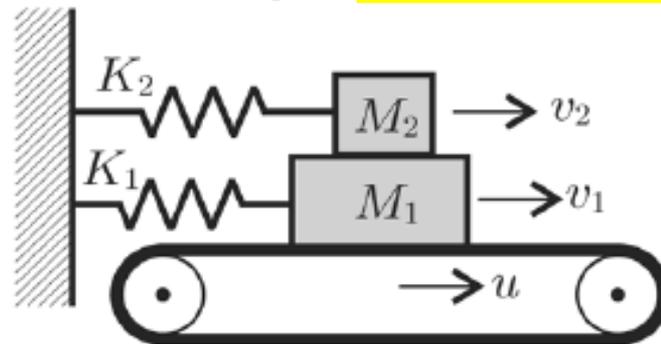
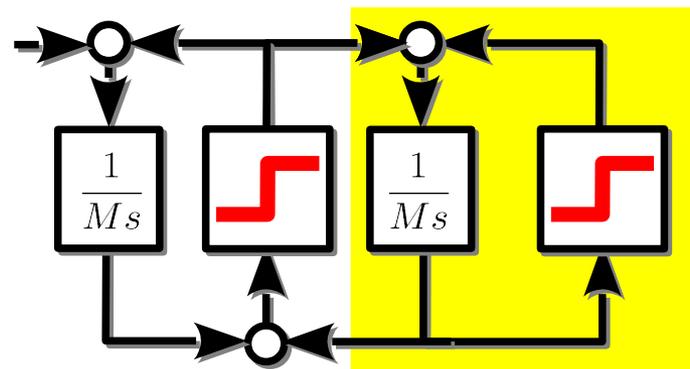
# 陰的積分のみで対処できない場合



- ◆ 符号関数を囲むループが実世界に露出している場合



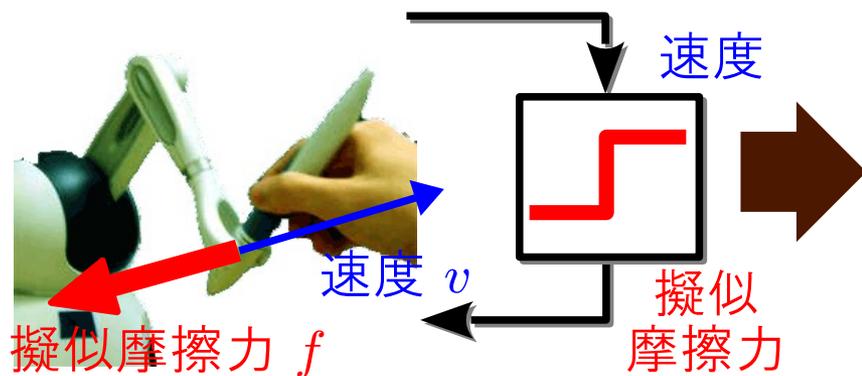
- ◆ 符号関数を囲むループが複雑な場合
  - e.g., 符号関数を含むループなど



# 「微分代数緩和」の物理的意味

[Kikuuwe et al.: IEEE-TRO, 2006]

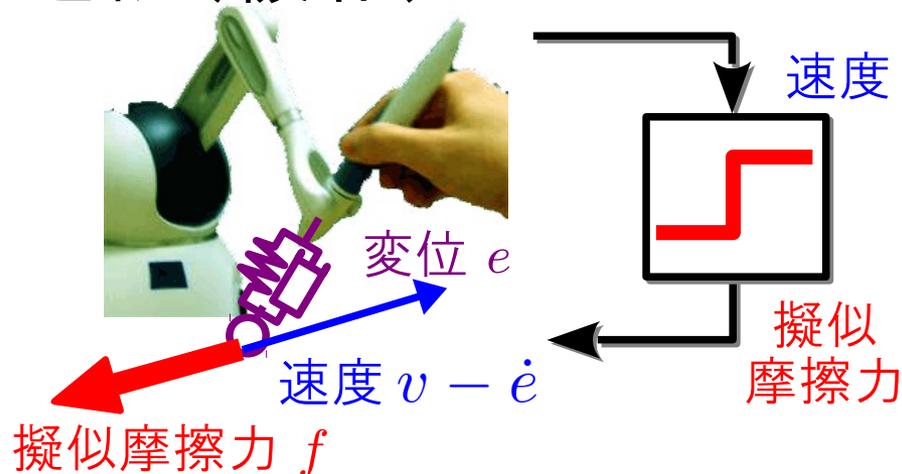
## ◆ 元のシステム



$$f \in F \operatorname{sgn}(v)$$

- ◆ 摩擦力はデバイスに直接働く。

## ◆ 近似(緩和)したシステム

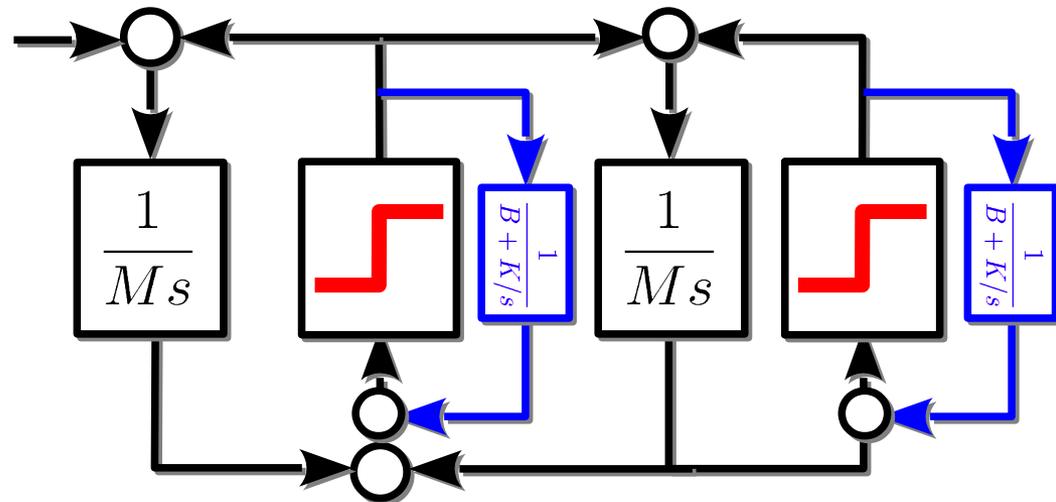
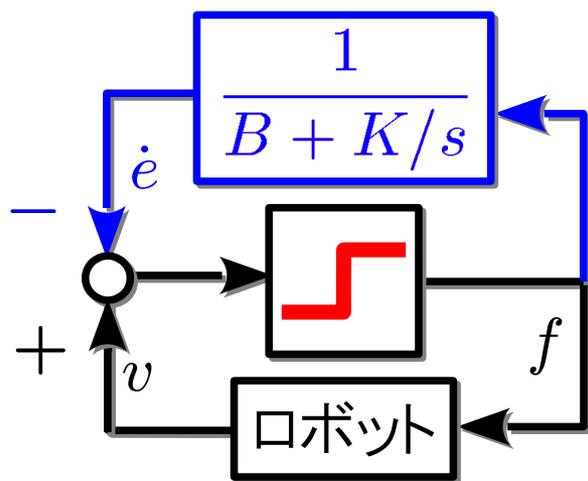


$$\begin{cases} f \in F \operatorname{sgn}(v - \dot{e}) \\ f = Ke + B\dot{e} \end{cases}$$

- ◆ 質量の無い仮想物体(プロクシ)と, 高剛性のバネ・ダンパ要素を想定。
- ◆ 摩擦力は仮想物体に働く。

# 対処法:「微分代数緩和」

- ◆ 符号関数周りに高ゲインのフィードバックを追加



$$f \in F \operatorname{sgn}(v)$$

$$f \in F \operatorname{sgn}(v - \dot{e})$$

$$f = Ke + B\dot{e}$$

← 微分代数包含式  
 $K$ と $B$ はできるだけ大きく

- ◆ 入出力間の拘束を緩和することに相当
- ◆ ペナルティベース手法の一種ともみなせる

# 数値計算アルゴリズムの導出

- ◆ 実現したい入出力関係(連続時間)

$$f \in F \operatorname{sgn}(v)$$

- ◆ 緩和した入出力関係(微分代数包含式)

$$f \in F \operatorname{sgn}(v - \dot{e})$$

$$f = Ke + B\dot{e}$$

- ◆ 上記の離散時間表現(代数包含式)

$$f_k \in F \operatorname{sgn}(v_k - (e_k - e_{k-1})/T)$$

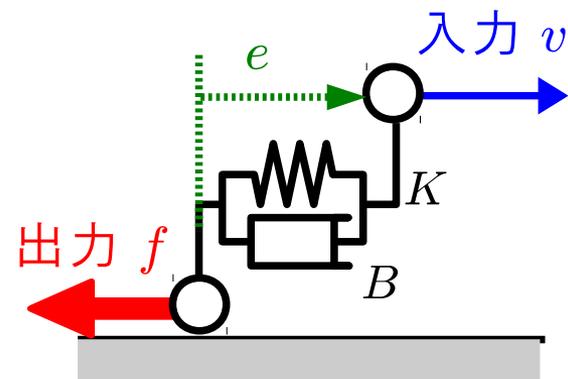
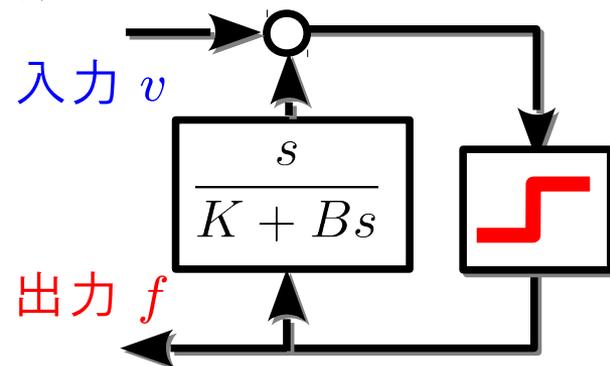
$$f_k = Ke_k + B(e_k - e_{k-1})/T$$

- ◆ アルゴリズム(上式を解く計算手順)

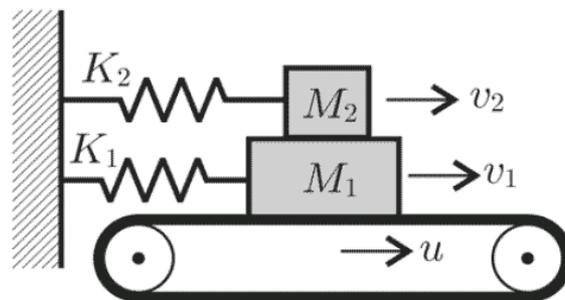
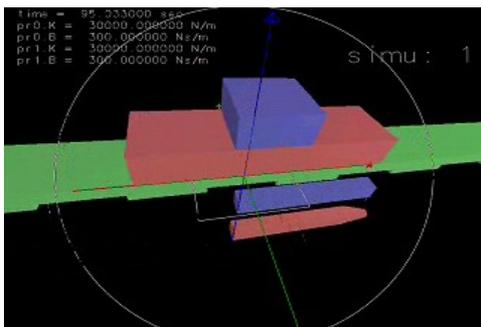
$$f_k := F \operatorname{sat}(((KT + B)u_k + Ke_{k-1})/F)$$

$$e_k := (Be_{k-1} + Tf_k)/(KT + B)$$

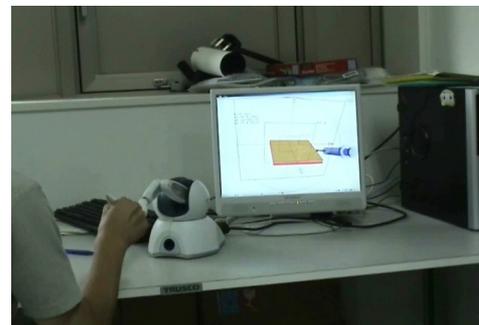
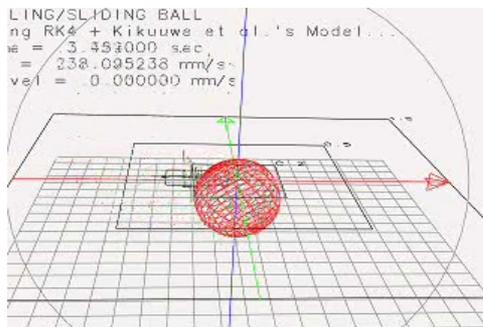
[Kikuuwe et al.: IEEE-TRO, 2006]



# 摩擦力のシミュレーション・力覚提示



- ◆ 複数の摩擦力が働く系でスティックスリップを再現

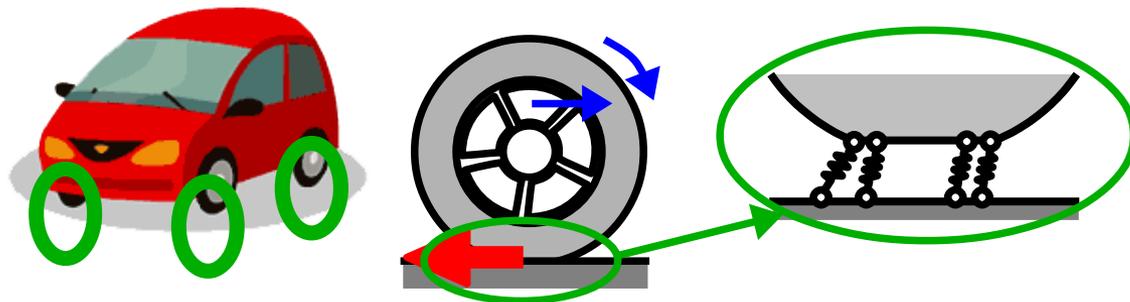


- ◆ 滑り⇒純転がりへの遷移も再現

- ◆ 注射針で皮膚を突き刺すときの摩擦力を提示

# 車両系のシミュレーション

[菊植ら,SI2007]



微小部位のバネ・ダンパ力      微小部分の変形速度      微小部位の摩擦力

$$0 \in Ke + B \left( \frac{\partial e}{\partial t} + R\omega \frac{\partial e}{\partial \xi} \right) - \mu F_n \operatorname{sgn} \left( V - R\omega - \frac{\partial e}{\partial t} - R\omega \frac{\partial e}{\partial \xi} \right)$$

微小部位の対地速度

$$f = \int_0^L \left( Ke + B \left( \frac{\partial e}{\partial t} + R\omega \frac{\partial e}{\partial \xi} \right) \right) d\xi$$

タイヤにかかる総力

- ◆ 偏微分代数包含式 (Partial Differential Algebraic Inclusion)
- ◆ タイヤの粘弾性を緩和のペナルティとして利用

# アルゴリズムの導出

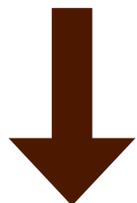
[菊植ら, SI2007]

- ◆ 偏微分代数包含式  
(連続時間)



時間・空間で離散化

- ◆ 代数包含式  
(離散時間)



解析解(定理1を利用)

$$y \in \text{sgn}(u - y)$$

$$\iff y = \text{sat}(u)$$

- ◆ アルゴリズム

$$f(\xi, t) = \mu F_n(\xi) \Phi \left( V(t) - \dot{\psi}(t)(\xi - \xi_c) - \frac{D}{Dt} e(\xi, t) \right)$$

$$f(\xi, t) = K e(\xi, t) + B \frac{D}{Dt} e(\xi, t)$$

$$\frac{D}{Dt} e(\xi, t) = \frac{\partial}{\partial t} e(\xi, t) + R\omega \frac{\partial}{\partial \xi} e(\xi, t)$$

$$\frac{D}{Dt} e(\xi, t) \approx \frac{e(\xi, t) - \hat{e}(\xi, t)}{\hat{T}(t)}$$

$$\hat{e}(\xi, t) = \frac{Z e(\xi, t - T) + TR|\omega(t)| e(\xi - \text{sgn}(\omega)Z, t)}{Z + TR|\omega(t)|}$$

$$\hat{T}(t) = TZ / (Z + TR|\omega(t)|)$$

$$\hat{T} := TZ / (Z + TR|\omega(t)|)$$

$$\xi := (1 - \text{sgn}(\omega))L/2$$

$$e(\xi, t) := 0$$

WHILE  $\xi \in [0, L]$

$$\xi += \text{sgn}(\omega)Z$$

$$\hat{e} := \frac{Z e(\xi, t - T) + TR|\omega(t)| e(\xi - \text{sgn}(\omega)Z, t)}{Z + TR|\omega(t)|}$$

$$v^* := V(t) - \dot{\psi}(t)(\xi - \xi_c)$$

$$f(\xi, t) := \mu F_n(\xi) \hat{\Phi} \left( v^* + \frac{K \hat{e}}{B + \hat{T}K}; \frac{1}{B + \hat{T}K} \right)$$

$$e(\xi, t) := (K \hat{e} + B f(\xi, t)) / (K \hat{T} + B)$$

END WHILE

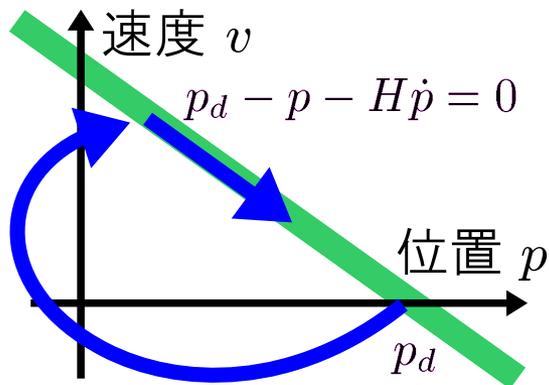


[5]

スライディングモード制御  
+  
微分代数緩和

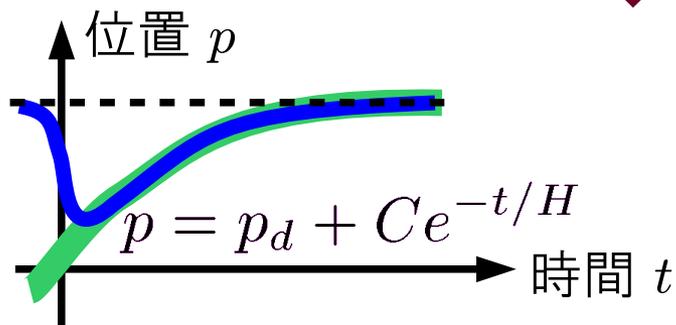
# スライディングモード制御で制御される系

- ◆ 状態空間内に設定した「切替面」を境界にして、操作量(アクチュエータの力)を不連続に切り替える制御則.



制御対象:  $M\ddot{p} = f + h$

制御則:  $f \in F \text{sgn}(p_d - p - H\dot{p})$

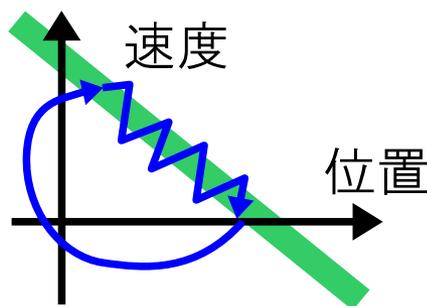
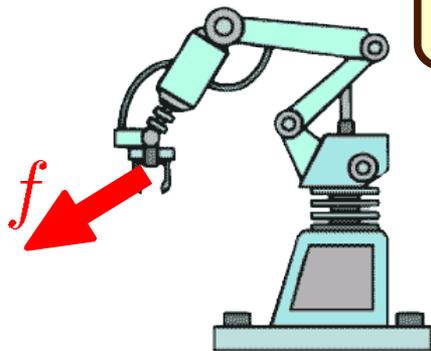


- ◆ **スライディングモード** = 状態が切替面に拘束されている状態.
  - $|f| < F$  で切り替えが無限に高速であれば, 拘束は**完全**.

# スライディングモード制御(SMC)

- ◆ ただし, そのまま実装すると, チャタリングが発生

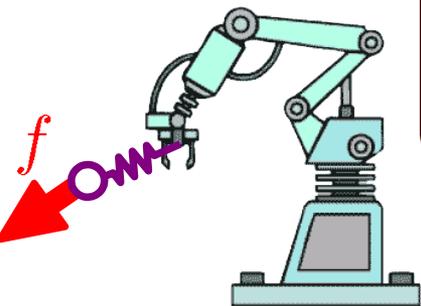
$$f = F \operatorname{sgn}(p_d - p + H(\dot{p}_d - \dot{p}))$$



- ◆ 「微分代数緩和」で実装 ⇒ チャタリングなし！！

$$f \in F \operatorname{sgn}(p_d - p - \dot{a} + H(\dot{p}_d - \dot{p} - \dot{a}))$$

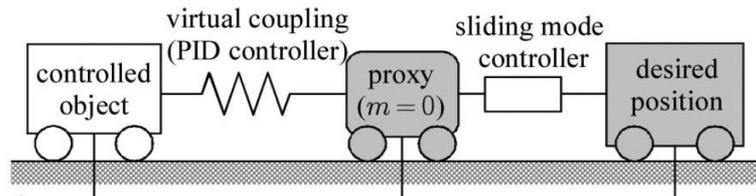
$$f = La + K\dot{a} + B\ddot{a}$$



- 質量の無い仮想物体(プロキシ)に対してSMC. プロキシとロボットはPID制御で接続
- 「プロキシベースト・スライディングモード制御」と命名

# “Proxy-based Sliding Mode Control”

[Kikuuwe and Fujimoto: ICRA2006]



- ◆ 微分代数包含式 (連続時間)  $f \in F \operatorname{sgn}(p_d - p - \dot{a} + H(\dot{p}_d - \dot{p} - \ddot{a}))$   
 $f = La + K\dot{a} + B\ddot{a}$



- ◆ 代数包含式 (離散時間)  $f_k \in F \operatorname{sgn}(p_{d,k} - q_k + H(\nabla p_{d,k} - \nabla q_k))$   
 $f_k = La_k + K\nabla a_k/T + B\nabla^2 a_k/T^2$



解析解(定理1を利用):  $y \in \operatorname{sgn}(u - y) \iff y = \operatorname{sat}(u)$

- ◆ アルゴリズム

$$\sigma_k := p_{d,k} - p_k + H(\nabla p_{d,k}/T - \nabla p_k/T)$$

$$f_k^* := \frac{B+KT+LT^2}{H+T} \sigma_k + \frac{KH-B+LT(2H+T)}{(H+T)T} a_{k-1} - \frac{KH-B+LTH}{(H+T)T} a_{k-2}$$

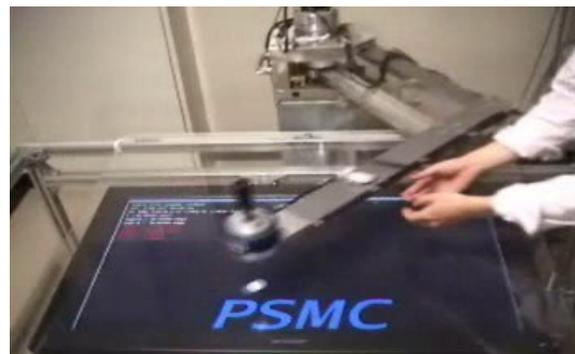
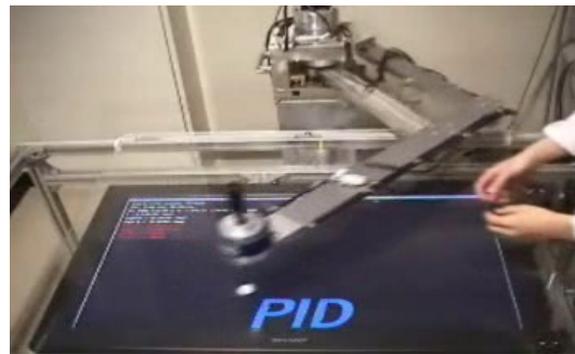
$$f_k := F \operatorname{sat}(f_k^*/F)$$

$$a_k := \frac{(2B+KT)a_{k-1} - Ba_{k-2} + T^2 f_k}{B+KT+LT^2}$$

# PSMCの利点

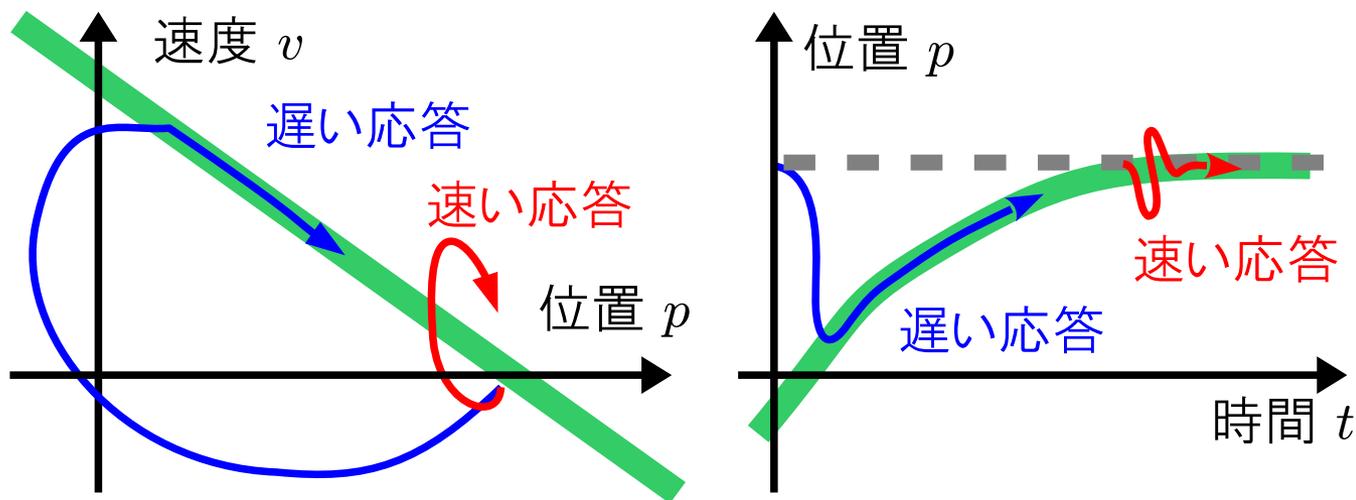
## ◆ PID制御とPSMCの比較

- ステップ状の目標値入力に対する応答
- 大きい外乱に対する応答



# PSMCの利点

- ◆ PSMCは、PID制御と同等に正確で、より安全。
  - 小さい位置誤差からは素早く復帰(正確=PIDの効果)
  - 大きい位置誤差からは緩やかに復帰(安全=SMCの効果)
- ◆ PID制御 or SMCのみで同様の遅い特性を作ろうとすると速度計測のノイズの影響を受けてしまう。



- ◆ SMCのあまり指摘されてこなかった理論上の利点を生かし、実用的な制御則を実現。

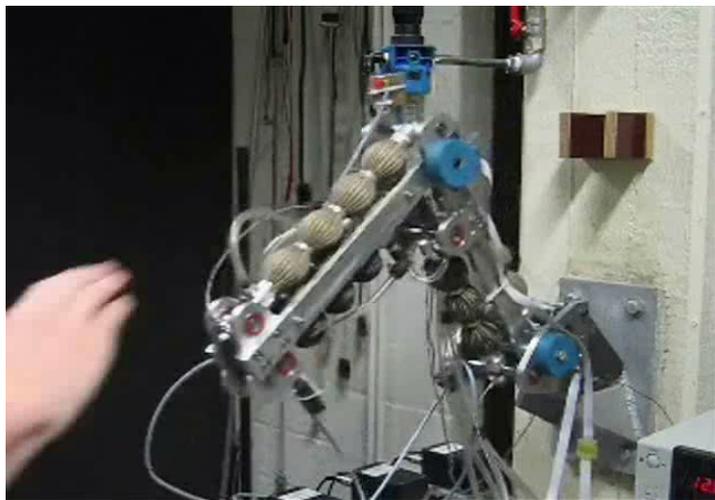
# ユニラテラル・テレオペレーション

- ◆ スライディング面の設計により速度制限
- ◆ トルクは安全なレベルで飽和
  - (このレベルがスライディングモード則のゲインに相当)
- ◆ スライディング状態(飽和前状態)では高ゲインPIDと等価

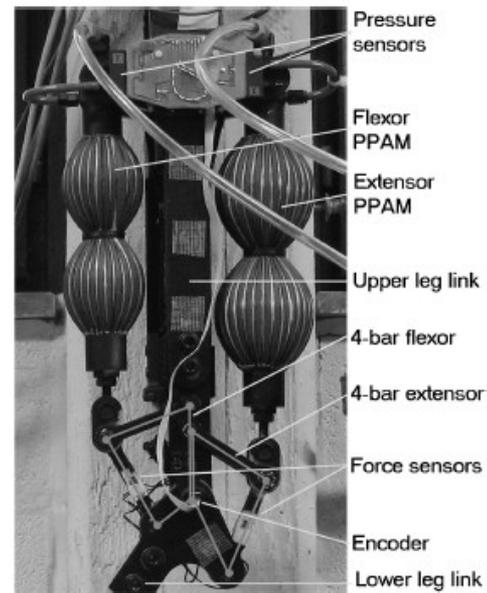


# 利用者の声

- ◆ 空気圧ロボットに実装されたいらしい
- ◆ Van Damme et al., “Proxy-based sliding mode control of a Planar Pneumatic Manipulator,” IJRR, 2009.
  - @ Vrije Univ. Brussel, Belgium
  - リハビリテーションロボットへの応用が主目的



(a)



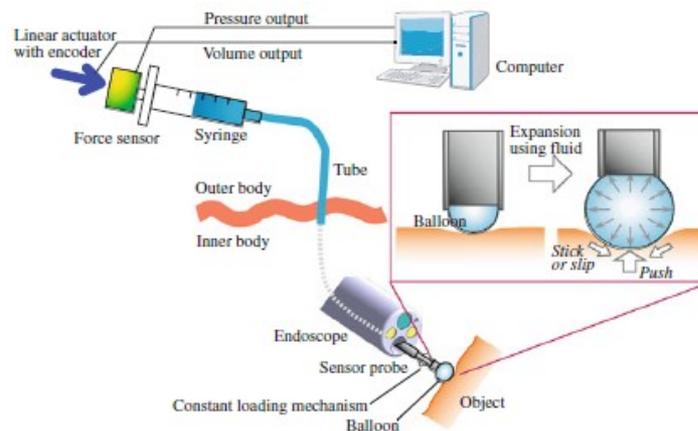
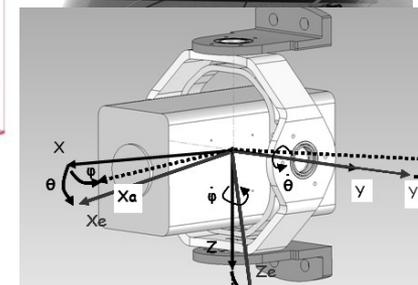
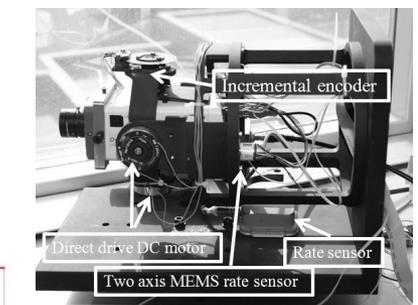
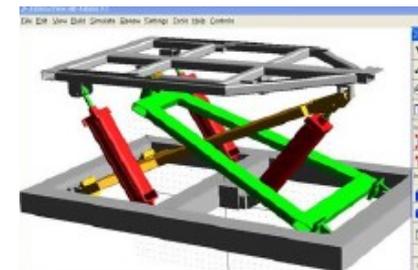
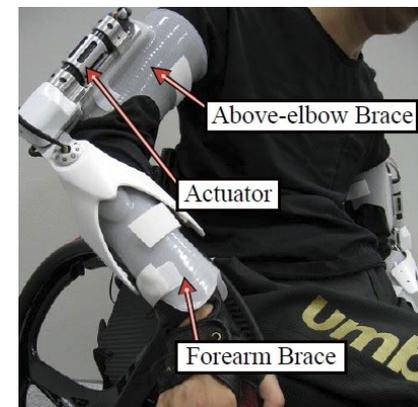
(b)

[Van Damme et al., ICRA2007]

[Beyl et al., 2011, Advanced Robotics]

# 他の導入実績

- ◆ Watanabe et al., "Extension motion assistance for upper limb using **Proxy-Based Sliding Mode Control**," Proc. SMC, 2011,
  - リハビリロボット @ 三重大学
- ◆ Prieto et al.: **Proxy-based sliding mode control** on platform of 3 degree of freedom," Advanced Robotics, 2013.
  - パラレルリンクのプラットフォーム@キューバ.
- ◆ Hastürk et al., "**Proxy-Based Sliding Mode** Stabilization of a Two-Axis Gimbaled Platform" Proc. World Congress on Engineering and Computer Science, 2011.
  - カメラのためのジンバルプラットフォーム @トルコ.
- ◆ Tanaka et al., "Development of a real-time tactile sensing system for brain tumor diagnosis," Int. J. CARS, 2010.
  - 触覚センサ@ 名工大



# パラダイムシフトの予感??

- ◆ Calanca et al. “Improving continuous approximation of sliding mode control,” ICAR2013.
- ◆ Among the existing methodologies, the Sliding Mode Control (SMC) technique is characterized by high simplicity and robustness. Also, despite not in original purpose, has been recently outlined that SMC in human-robot interaction can effectively improve safety, thanks to its intrinsic saturation of control action [1], [2]. Combining robustness with safety, control laws based on SMC have been recently applied to rehabilitation robotics [3], [4], being one of the favored control concepts in the area.

## REFERENCES

- [1] R. Kikuuwe and H. Fujimoto, “Proxy-based sliding mode control for accurate and safe position control,” in *IEEE International Conference on Robotics and Automation*, 2006, pp. 25–30. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1641156>
- [2] M. V. Damme and B. Vanderborght, “Proxy-based sliding mode control of a planar pneumatic manipulator,” *International Journal of Robotics Research*, vol. 28, no. 2, pp. 266–284, 2009. [Online]. Available: <http://ijr.sagepub.com/content/28/2/266.short>
- [3] P. Beyl, M. Van Damme, R. Van Ham, B. Vanderborght, and D. Lefeber, “Design and control of a lower limb exoskeleton



# 理論分野からの反応

◆ まったくなし！！！！



**[6]**

**連続時間領域で扱う  
摩擦あり片側拘束**

# 基本: 拘束条件付きの運動方程式

## ◆ 運動方程式

$$M(q)\ddot{q} + \Phi(q, \dot{q}, t) = J(q)^T \lambda$$

## ◆ 等式拘束

$$\begin{bmatrix} J_v(q)\dot{q} \\ \Psi_p(q) \end{bmatrix} = 0$$

位置・速度

ここで  $J(q) \triangleq \begin{bmatrix} J_v(q) \\ \partial\Psi_p(q)/\partial q \end{bmatrix}$

- 速度拘束(車両などのノンホロ拘束)
- 位置拘束(ホロノミック拘束)

下半分を微分  $\xrightarrow{\hspace{1cm}}$   $J(q)\dot{q} = 0$   $\xrightarrow{\hspace{1cm}}$   $J(q)\ddot{q} + \dot{J}(q)\dot{q} = 0$

速度 加速度

- ◆ 加速度  $\ddot{q}$  を消去して  $\lambda = \dots$  を得て, それを運動方程式に代入すると計算できる! 簡単!

# 片側拘束のときは？ 摩擦があったら？

$$M(q)\ddot{q} + \Phi(q, \dot{q}, t) = J(q)^T \underline{\lambda}_{\text{力}}$$

## ◆ 片側拘束

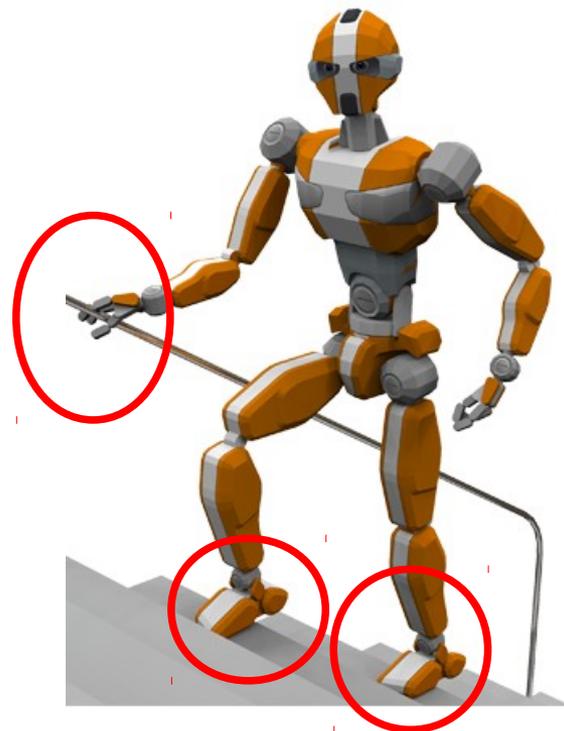
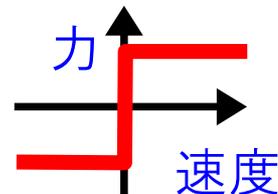
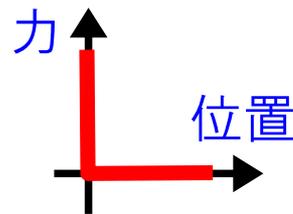
$$0 \leq \underline{\Psi}_p(q) \perp \underline{\lambda}_n \geq 0$$

位置
力

## ◆ 摩擦

$$\underline{\lambda}_t \in -\mu \underline{\lambda}_n \text{sgn}(\underline{J}_v(q)\dot{q})$$

力
速度



JVRC (主催: NEDO) のサイトより

- ◆ 等式拘束の場合と見た目が違いすぎる
- ◆ 美しくない！ 扱いづらい！
- ◆ 離散化しないと計算できない

# この研究テーマのポイント

## ◆ 摩擦あり片側拘束の新しい表現

$$\begin{cases} M(q)\ddot{q} + \Phi(q, \dot{q}, t) = J(q)^T \lambda \\ \begin{bmatrix} J_v(q)\dot{q} \\ \Psi_p(q) \end{bmatrix} \in - \lim_{\varepsilon \rightarrow +0} \mathcal{N}_{\mathcal{F}^\varepsilon}^{U^\varepsilon}(\lambda) \end{cases}$$

- 等式拘束の場合の自然な一般化

## ◆ それを計算／解析するための近似(緩和)方法

$$\begin{cases} M(q)\ddot{q} + \Phi(q, \dot{q}, t) = J(q)^T \lambda \\ J(q)\ddot{q} + \dot{J}(q)\dot{q} + \frac{2\zeta}{\delta} J(q)\dot{q} + \frac{1}{\delta^2} \begin{bmatrix} 0 \\ \Psi_p(q) \end{bmatrix} \in - \lim_{\varepsilon \rightarrow +0} \mathcal{N}_{\mathcal{F}^\varepsilon}^{U^\varepsilon}(\lambda) \end{cases}$$

- (奇しくも?) Baumgarte 安定化法(1920)の一般化

## ◆ 摩擦接触特有の代数問題について

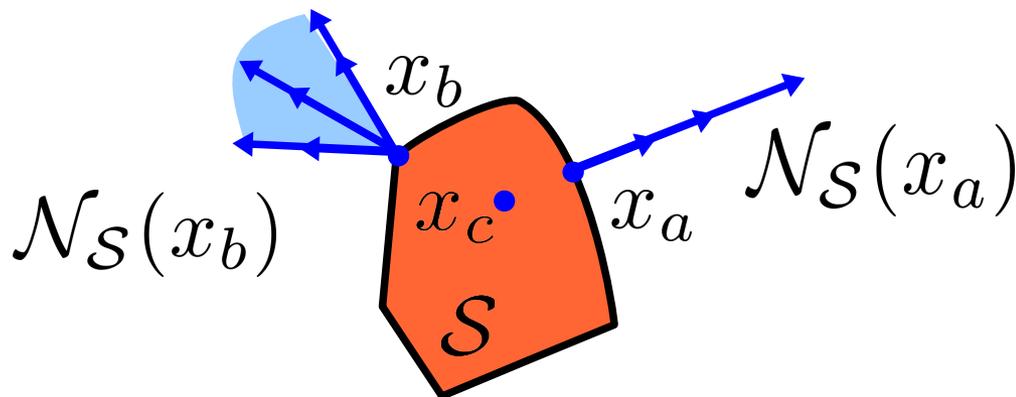
- 線形相補性問題(LCP)でも二次錐相補性問題でもない

$$\lambda - A^{-1}b \in - \lim_{\varepsilon \rightarrow +0} A^{-1} \mathcal{N}_{\mathcal{F}^\varepsilon}^{U^\varepsilon}(\lambda)$$

# 便利なツール: 法錐 (Normal Cone)

◆ 点  $x$  における集合  $S$  のメトリック  $U$  での法錐

$$\mathcal{N}_S^U(x) \triangleq \begin{cases} \{\xi \in \mathbb{R}^n \mid \xi^T U(x^* - x) \leq 0 \ \forall x^* \in S\} & \text{if } x \in S \\ \emptyset & \text{if } x \notin S \end{cases}$$



- ◆  $x$  が集合  $S$  の表面  $\Rightarrow$  法錐は法線ベクトルの集合
- ◆  $x$  が集合  $S$  の内部  $\Rightarrow$  法錐はゼロベクトル
- ◆  $x$  が集合  $S$  の外部  $\Rightarrow$  法錐は空集合

# 便利なツール: 法錐 (Normal Cone)

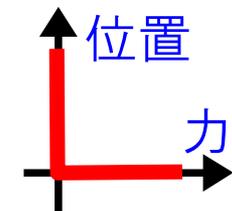
◆ 点  $x$  における集合  $S$  のメトリック  $U$  での法錐

$$\mathcal{N}_S^U(x) \triangleq \begin{cases} \{\xi \in \mathbb{R}^n \mid \xi^T U(x^* - x) \leq 0 \ \forall x^* \in S\} & \text{if } x \in S \\ \emptyset & \text{if } x \notin S \end{cases}$$

◆ 一次元の {位置・速度} vs 力の表現には使える!

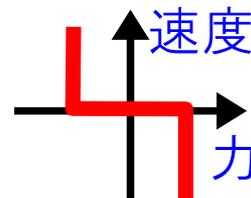
■ 片側拘束:

$$0 \leq f \perp p \geq 0 \quad \Rightarrow \quad p \in -\mathcal{N}_{\mathbb{R}^+}^I(f)$$



■ クーロン摩擦:

$$f \in -F \text{sgn}(v) \quad \Rightarrow \quad v \in -\mathcal{N}_{[-F, F]}^I(f)$$



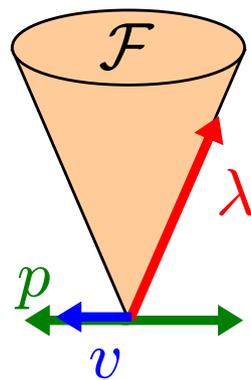
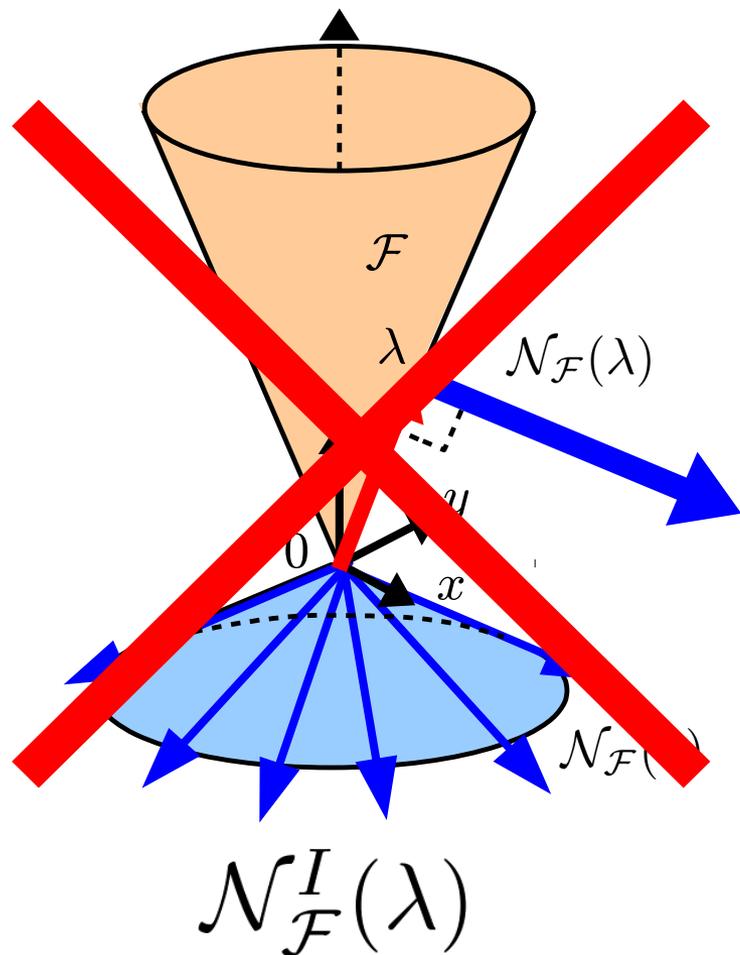
◆ 基本形:

$$\{\text{位置 or 速度}\} \in -\mathcal{N}_{\mathcal{F}}(\text{力})$$

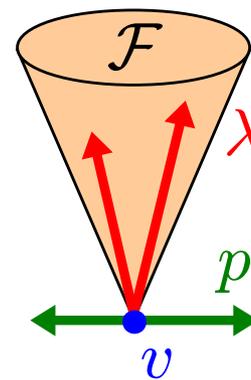
許容される力の集合 ↗

# 摩擦あり片側接触

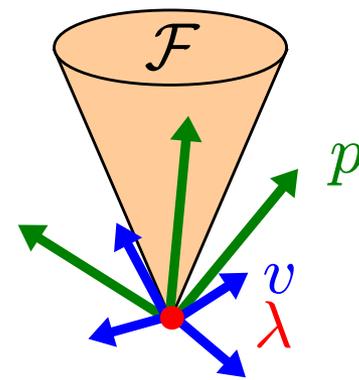
◆ 摩擦円錐の法錐：位置・速度と関連付けられる？



動摩擦



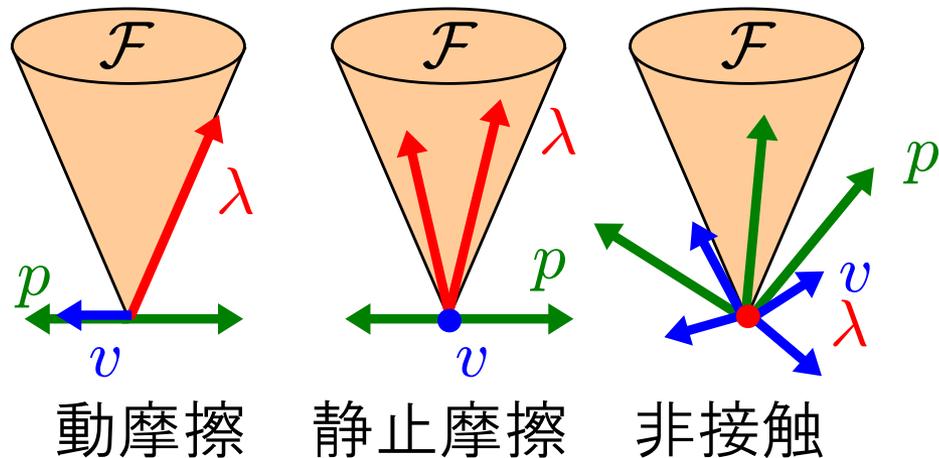
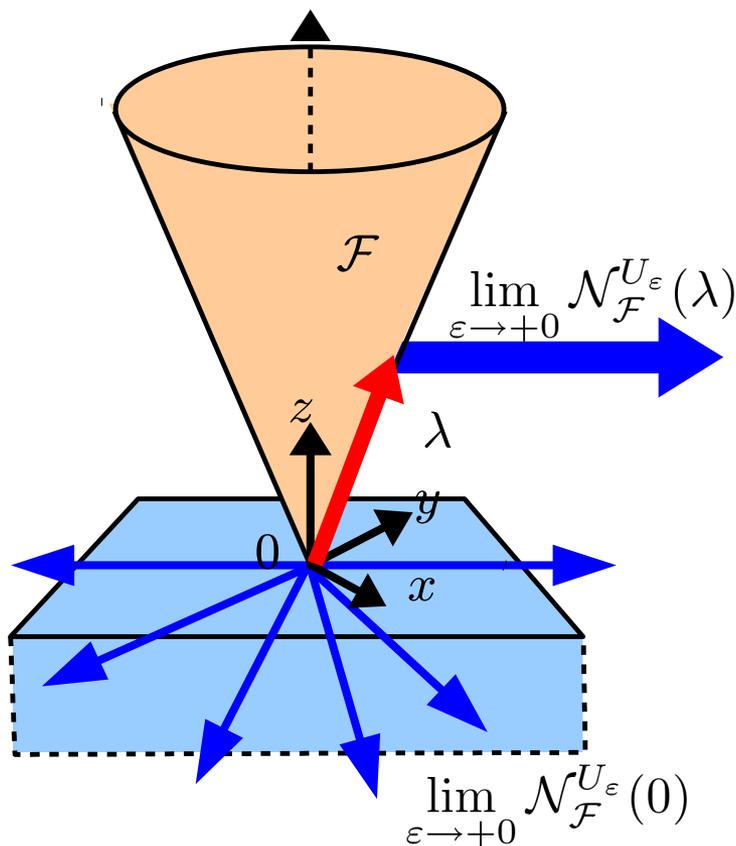
静止摩擦



非接触

# 摩擦あり片側接触

◆ 摩擦円錐の法錐：位置・速度と関連付けられる



$$\begin{bmatrix} v_x \\ v_y \\ p_z \end{bmatrix} \in - \lim_{\epsilon \rightarrow +0} \mathcal{N}_{\mathcal{F}}^{U_\epsilon}(\lambda)$$

ただし  $U_\epsilon = \text{diag}[1, 1, 1/\epsilon]$

**NEW!!!**

# 例

$$M(q)\ddot{q} + \Phi(q, \dot{q}, t) = J(q)^T \lambda$$

$$\begin{bmatrix} J_v(q)\dot{q} \\ \Psi_p(q) \end{bmatrix} \in - \lim_{\varepsilon \rightarrow +0} \mathcal{N}_{\mathcal{F}}^{U_\varepsilon}(\lambda)$$

- ◆ 等式拘束  $\mathcal{F} = \mathbb{R}^n$   $U_\varepsilon = I_n$
- ◆ 片側拘束  $\mathcal{F} = \mathbb{R}_+^n$   $U_\varepsilon = I_n$
- ◆ クーロン摩擦  $\mathcal{F} = [-F, F]$   $U_\varepsilon = 1$
- ◆ 摩擦接触  $\mathcal{F} = \text{cone}$   $U_\varepsilon = \text{diag}[1, 1, \frac{1}{\varepsilon}]$
- ◆ 複数の摩擦接触  $\mathcal{F} = \text{cone} \times \text{cone} \times \dots$   $U_\varepsilon = \text{diag}[1, 1, \frac{1}{\varepsilon}, 1, 1, \frac{1}{\varepsilon}, \dots]$

# どうやって計算する？

- ◆ 等式拘束  $\Rightarrow$  微分して連立して  $\ddot{q}$  を消去 .

$$M(q)\ddot{q} + \Phi(q, \dot{q}, t) = J(q)^T \lambda$$

$$\begin{bmatrix} J_v(q)\dot{q} \\ \Psi_p(q) \end{bmatrix} = 0 \quad \longrightarrow \quad J(q)\ddot{q} + \dot{J}(q)\dot{q} = 0$$

- ◆ 不等式拘束から加速度を得るには？

$$x \geq 0 \quad \xrightarrow{\text{ほぼ等価}} \quad \lim_{\delta \rightarrow +0} \left( \ddot{x} + \frac{2\zeta\dot{x}}{\delta} + \frac{x}{\delta^2} \right) \geq 0$$

$$\xrightarrow{\text{緩和}} \quad \ddot{x} + \frac{2\zeta\dot{x}}{\delta} + \frac{x}{\delta^2} \geq 0 \quad \begin{array}{l} \delta: \text{微小} \\ \zeta: \text{任意} \end{array}$$

**NEW!!!**

# 提案手法: Baumgarte-like 緩和

$$M(q)\ddot{q} + \Phi(q, \dot{q}, t) = J(q)^T \lambda$$

$$\begin{bmatrix} J_v(q)\dot{q} \\ \Psi_p(q) \end{bmatrix} \in - \lim_{\varepsilon \rightarrow +0} \mathcal{N}_{\mathcal{F}}^{U_\varepsilon}(\lambda)$$

緩和

$$J(q)\ddot{q} + \dot{J}(q)\dot{q} + \frac{2\zeta}{\delta} J(q)\dot{q} + \frac{1}{\delta^2} \begin{bmatrix} 0 \\ \Psi_p(q) \end{bmatrix} \in - \lim_{\varepsilon \rightarrow +0} \mathcal{N}_{\mathcal{F}}^{U_\varepsilon}(\lambda)$$

$\ddot{q}$  を消去すると……

$$\lambda - A^{-1}b \in - \lim_{\varepsilon \rightarrow +0} A^{-1} \mathcal{N}_{\mathcal{F}}^{U_\varepsilon}(\lambda)$$

ただし  $A(q) \triangleq J(q)M(q)^{-1}J(q)^T$

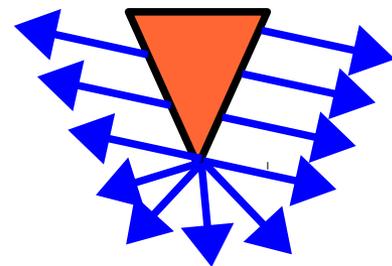
$$b(\delta, q, \dot{q}) \triangleq J(q)M(q)^{-1}\Phi(q, \dot{q}, t) - H(q, \dot{q})\dot{q} - \frac{2\zeta}{\delta} J(q)\dot{q} - \frac{1}{\delta^2} \begin{bmatrix} 0 \\ \Psi_p(q) \end{bmatrix}$$

- ◆ 上記の代数問題が  $\lambda$  に関して唯一解を持てば、それを運動方程式に代入して計算できる！！

# 摩擦あり片側接触に特有の代数問題

$A, b, U_\varepsilon, \mathcal{F}$  が与えられたとき,  

$$\lambda - A^{-1}b \in - \lim_{\varepsilon \rightarrow +0} A^{-1} \mathcal{N}_{\mathcal{F}}^{U_\varepsilon}(\lambda)$$
 を満たす  $\lambda$  を求めよ.



- ◆ 従来研究では、後退オイラー法での離散化によって、等価な式に到達。解き方も提案されている。
- ◆ しかし、表現はまったく異なる。

- ◆ Acary et al., 2011 ZAMM:

- ◆ Bertails–Descoubes et al., 2011 TOG:

- ◆ Bonnefon & Daviet, 2011 INRIA–TR:

$$\begin{cases} Mv + f = H^\top r \\ \tilde{u} = Hv + w + Es \\ L^* \ni \tilde{u} \perp r \in L, \end{cases}$$

$$f^{AC}(\mathbf{W}\lambda + \mathbf{b}, \lambda) = \mathbf{0}.$$

$$\begin{aligned} U &= MR + q, \\ (U, R) &\in C(\mu, e) \end{aligned}$$

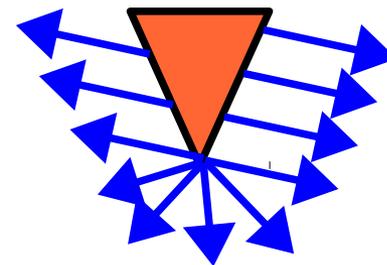
- ◆ 上記の表現(と幾何学的解釈)は新規

# 過去の研究をこの表現であらわすと...

## ◆ 厳密解 (Moreau 1999 など)

$$\lambda - A^{-1}b \in - \lim_{\varepsilon \rightarrow +0} A^{-1} \mathcal{N}_{\mathcal{F}}^{U\varepsilon}(\lambda)$$

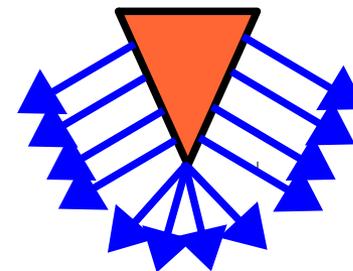
- SICONOS (INRIA) で採用



## ◆ Anitescu & Tasora (2010)

$$\lambda - A^{-1}b \in -A^{-1} \mathcal{N}_{\mathcal{F}}(\lambda)$$

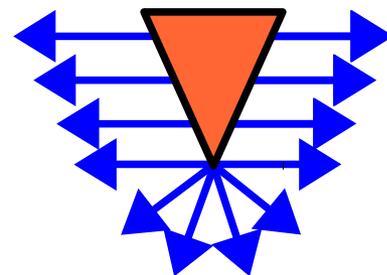
- Chrono (Negrut et al.) で採用
- 二次錐相補性問題 (SOCCP) の標準形だから速い. しかし不正確



## ◆ Nakaoka et al. (2007)

$$\lambda - A^{-1}b \in - \lim_{\varepsilon \rightarrow +0} \mathcal{N}_{\mathcal{F}}^{U\varepsilon}(\lambda)$$

- Choreonoid (AIST) で採用
- 単純な射影だから速い. しかし不正確

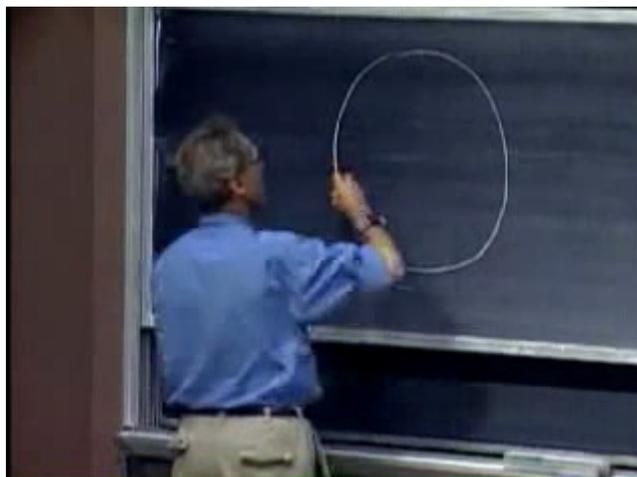
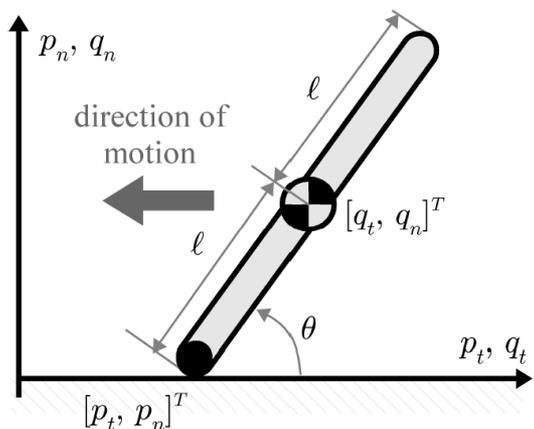


# 解の唯一性？

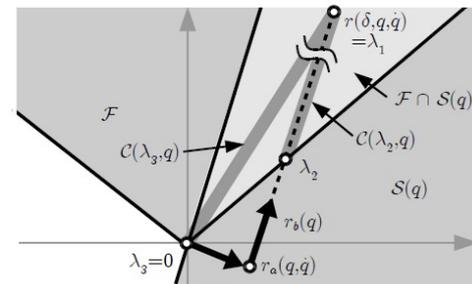
$A, b, U_\varepsilon, \mathcal{F}$  が与えられたとき,  

$$\lambda - A^{-1}b \in - \lim_{\varepsilon \rightarrow +0} A^{-1} \mathcal{N}_{\mathcal{F}}^{U_\varepsilon}(\lambda)$$
 を満たす  $\lambda$  を求めよ。

- ◆ 解が唯一であることは保証されない！！
- ◆ パンルヴェ (Painlevé) のパラドックス



wikipediaより



# このテーマのまとめ

## ◆ 非平滑力学系の新しい表現

$$\begin{cases} M(q)\ddot{q} + \Phi(q, \dot{q}, t) = J(q)^T \lambda \\ \begin{bmatrix} J_v(q)\dot{q} \\ \Psi_p(q) \end{bmatrix} \in - \lim_{\varepsilon \rightarrow +0} \mathcal{N}_{\mathcal{F}^\varepsilon}^U(\lambda) \end{cases}$$

## ◆ その Baumgarte-like 緩和

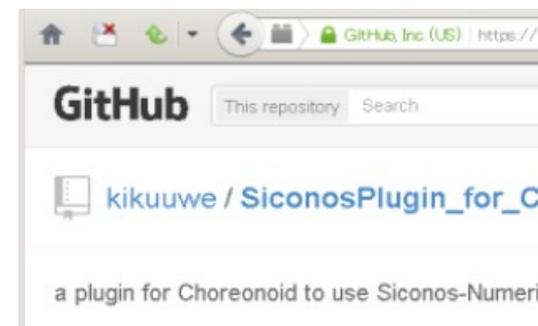
- 離散化せずに、連続時間領域での解析が可能に

$$\begin{cases} M(q)\ddot{q} + \Phi(q, \dot{q}, t) = J(q)^T \lambda \\ J(q)\ddot{q} + \dot{J}(q)\dot{q} + \frac{2\zeta}{\delta} J(q)\dot{q} + \frac{1}{\delta^2} \begin{bmatrix} 0 \\ \Psi_p(q) \end{bmatrix} \in - \lim_{\varepsilon \rightarrow +0} \mathcal{N}_{\mathcal{F}^\varepsilon}^U(\lambda) \end{cases}$$

## ◆ 摩擦接触特有の代数問題の新しい表現

$$\lambda - A^{-1}b \in - \lim_{\varepsilon \rightarrow +0} A^{-1} \mathcal{N}_{\mathcal{F}^\varepsilon}^U(\lambda)$$

- ◆ Choreonoid (AIST) の摩擦演算を厳密解で置換するプラグイン「SiconosPlugin」(INRIAの「SICONOS」を利用)をGitHubで公開中





# まとめ

# まとめ

- ◆ 微分包含式は解析的には特異な性質を持つ.
- ◆ しかし普通の数値計算法では, その特異な性質は失われてしまう.
- ◆ 下記の変換式を用いた計算アルゴリズムにより, 特異な性質を温存した利用・応用を実現.

$$y = \text{sgn}(x - y) \iff y = \text{sat}(x)$$

- ◆ 摩擦シミュレーションとスライディングモード制御のための「**微分代数緩和**」
- ◆ 不等式拘束条件付きの微分方程式の取扱