Quadruped Robotics

第6回MPC勉強会　鶴原康太

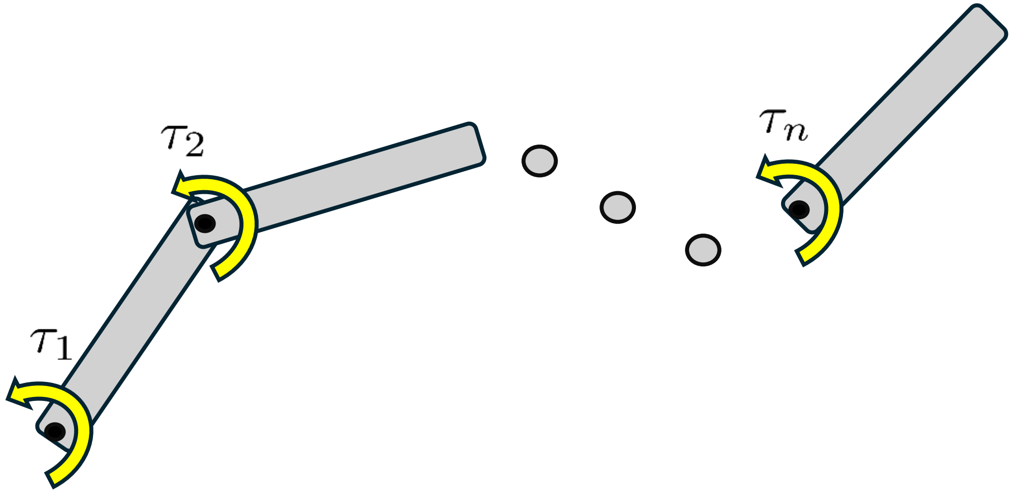
本資料の構成

- ロボットの制御基礎
- 四脚ロボットの制御基礎
- 論文

ロボットの制御基礎

ロボットの制御基礎

ロボットの運動方程式



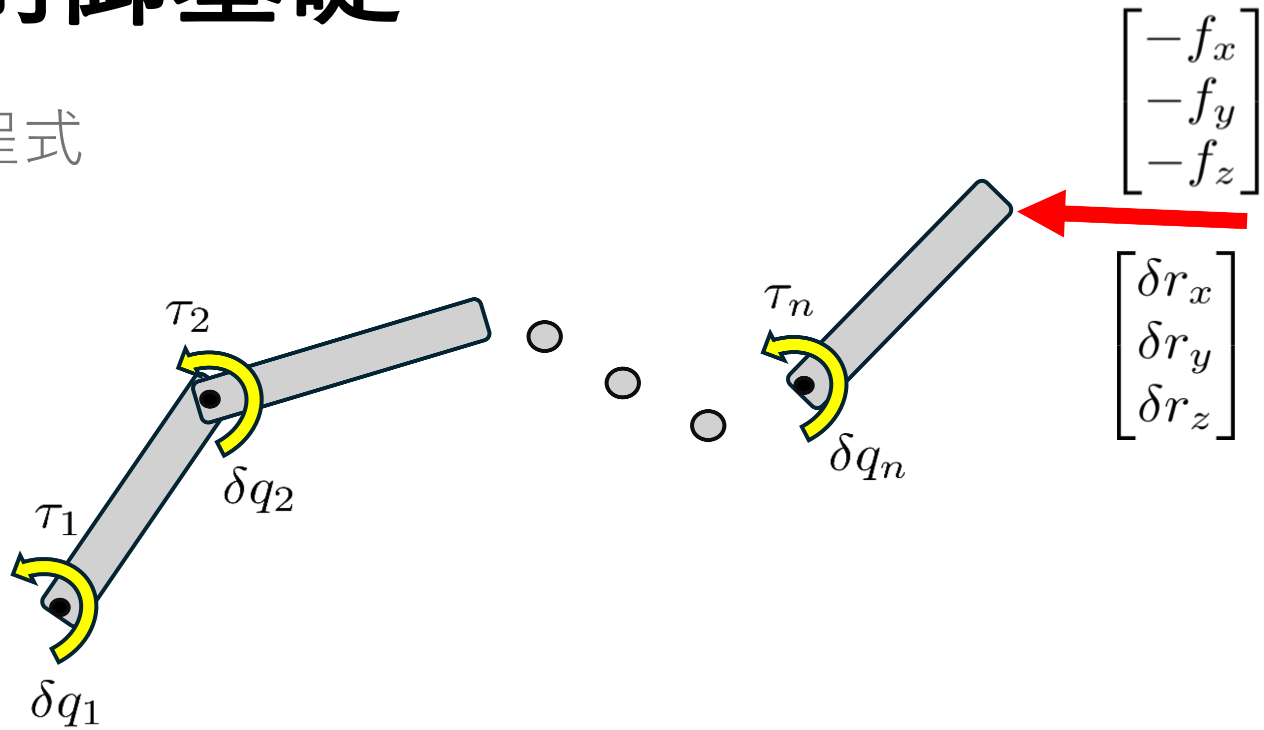
ロボットを**トルク制御**で動かしたい場合,**運動方程式**を導く必要がある

$$\tau = M(q)\ddot{q} + h(q, \dot{q}) + g(q)$$

※関節数が多い場合はNewton-Euler法を用いて漸化的に解く

ロボットの制御基礎

ロボットの運動方程式



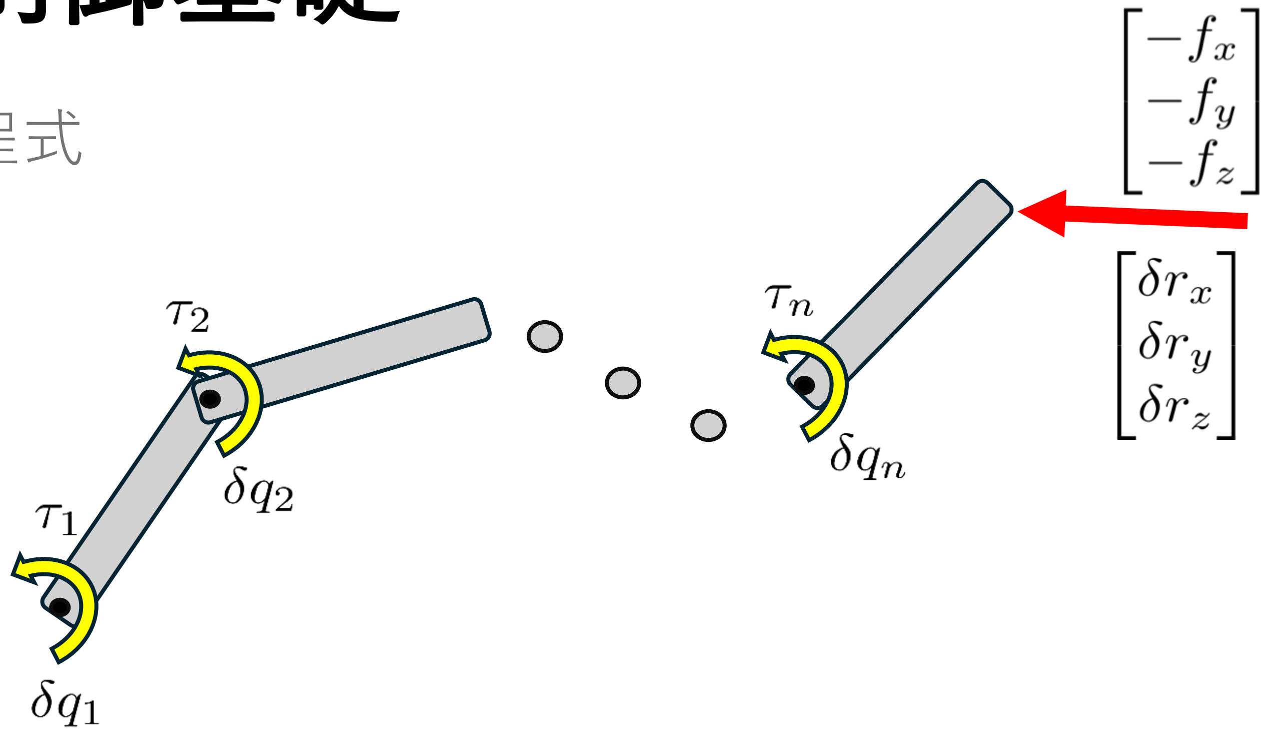
仮想仕事の原理：つり合い時なら仮想的な微小変位に対する仕事はゼロ

$$\delta W = \begin{bmatrix} \delta q_1 & \delta q_2 & \cdots & \delta q_n \end{bmatrix} \begin{bmatrix} \tau_1 \\ \tau_2 \\ \vdots \\ \tau_n \end{bmatrix} + \begin{bmatrix} \delta r_x & \delta r_y & \delta r_z \end{bmatrix} \begin{bmatrix} -f_x \\ -f_y \\ -f_z \end{bmatrix} = 0$$

$$\delta W = \delta \mathbf{q}^{\top} \boldsymbol{\tau} - \delta \mathbf{r}^{\top} \mathbf{f} = 0$$

ロボットの制御基礎

ロボットの運動方程式



$$\delta W = \delta \mathbf{q}^{\top} \boldsymbol{\tau} - \delta \mathbf{r}^{\top} \mathbf{f} = 0$$

ここで微分運動学の関係式$\delta \mathbf{r} = \mathbf{J} \delta \mathbf{q}$を用いる

$$\delta \mathbf{q}^{\top} \boldsymbol{\tau} - \delta \mathbf{q}^{\top} \mathbf{J}^{\top} \mathbf{f} = 0$$

$$\boldsymbol{\tau} = \mathbf{J}^{\top} \mathbf{f}$$

ロボットの制御

歪対称行列

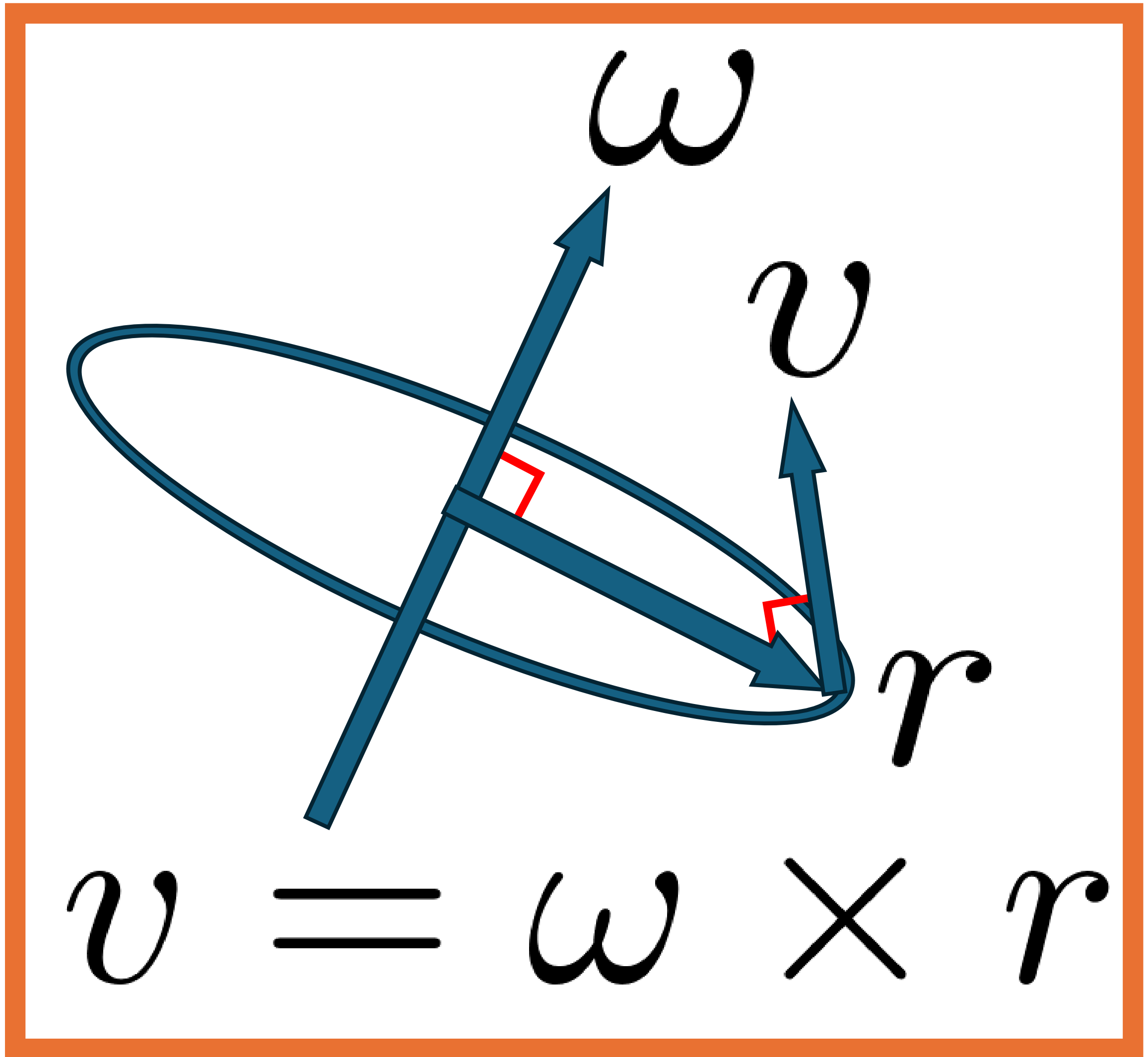
$$\begin{bmatrix} a \\ b \\ c \end{bmatrix} \times \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} bz - cy \\ cx - az \\ ay - bx \end{bmatrix} = \begin{bmatrix} 0 & -c & b \\ c & 0 & -a \\ -b & a & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

$$\boxed{\begin{aligned} \begin{bmatrix} a \\ b \\ c \end{bmatrix}^{\wedge} &= \begin{bmatrix} 0 & -c & b \\ c & 0 & -a \\ -b & a & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & -c & b \\ c & 0 & -a \\ -b & a & 0 \end{bmatrix}^{\vee} &= \begin{bmatrix} a \\ b \\ c \end{bmatrix} \end{aligned}}$$

$$\mathcal{A} \times \mathcal{B} = [\mathcal{A}_{\times}]\,\mathcal{B}$$

ロボットの制御基礎

慣性テンソル



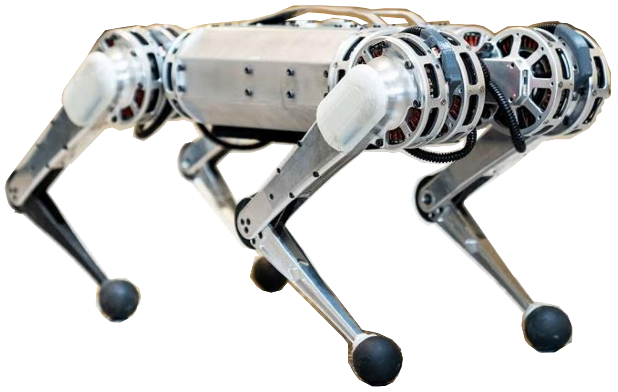
$$
\begin{aligned}
\mathrm{L} &= \sum_{i=1}^{n} r_i \times m_i v_i \\
&= \sum_{i=1}^{n} m_i (r_i \times (\omega \times r_i)) \\
&= -\sum_{i=1}^{n} m_i (r_i \times (r_i \times \omega)) \\
&= -\sum_{i=1}^{n} m_i [r_\times][r_\times]\omega \\
&= \underbrace{\sum_{i=1}^{n} m_i [r_\times]^T [r_\times]}_{I}\omega
\end{aligned}
$$

$$\bar{I} = R I R^T$$

四脚ロボットの制御基礎

四脚ロボットの制御導入

四脚ロボットの難点

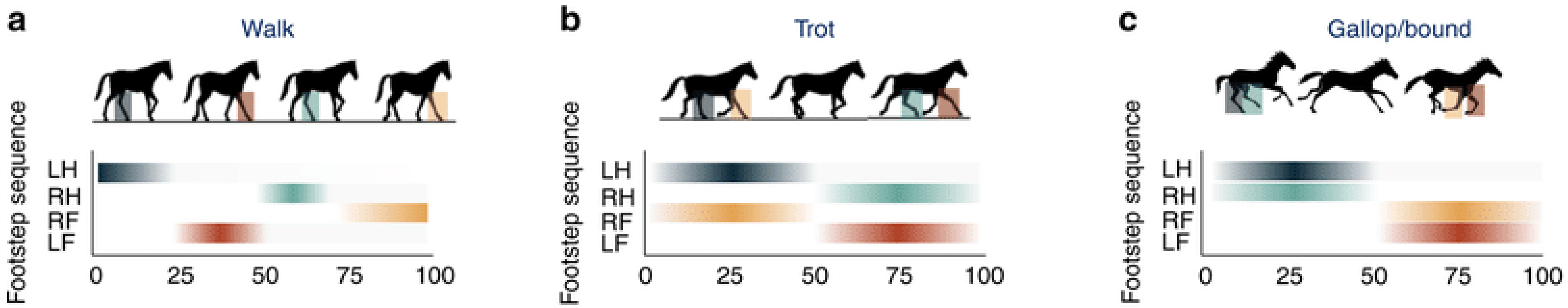


劣駆動(under-actuated)

浮遊基底(floating base)

四脚ロボットの制御基礎

Gaitによる動作パターン

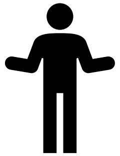


Dutta, Sourav, et al. "Programmable coupled oscillators for synchronized locomotion." *Nature communications* 10.1 (2019): 3299.

遊脚軌道は事前に与える(障害物回避などはここで設定)

各脚について逆運動学を解き目標関節角度を取得する

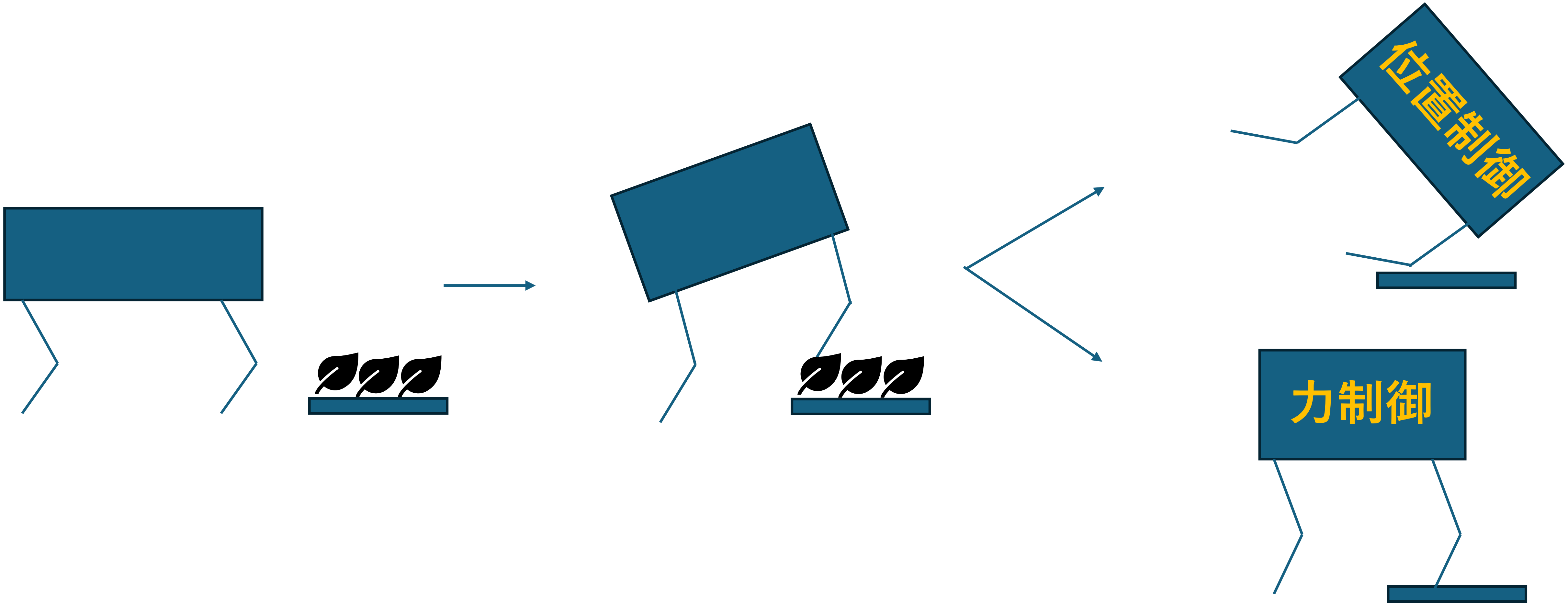
これだけじゃダメ？？



▶実際これだけの制御の四脚ロボットも多く存在する

四脚ロボットの制御基礎

なぜ力制御が必要か



四脚ロボットの制御基礎

- 脚制御
  - 支持脚制御(脚が地面に接してる)
  - 遊脚制御(脚が地面に接してない状態)

四脚ロボットの制御基礎

ロボットの運動方程式

本体目標速度,角速度

↓

関節トルク

→

本体目標速度,角速度

↓

最適化問題(MPC)

↓

目標軌道

↓

関節トルク

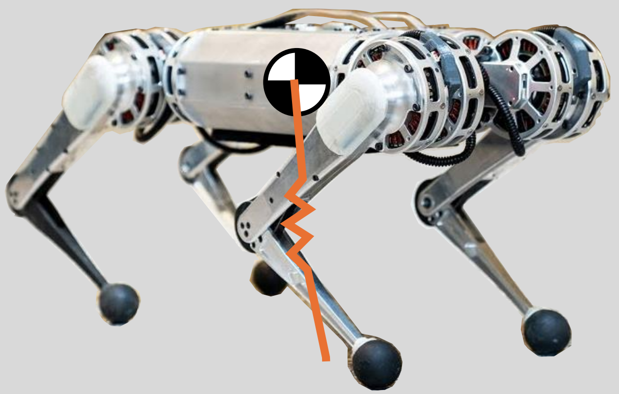
四脚ロボットの制御基礎

モデル分類

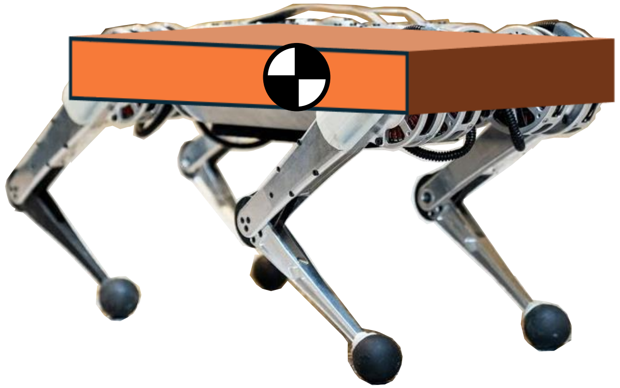
ロボットを完全に数式で表すのは困難

特徴的な性質を数式で表す

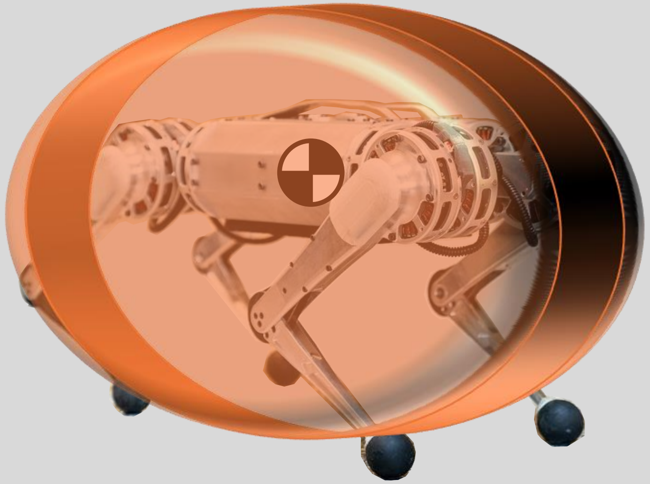
SLIP



Single Rigid Body



Centroidal Dynamics

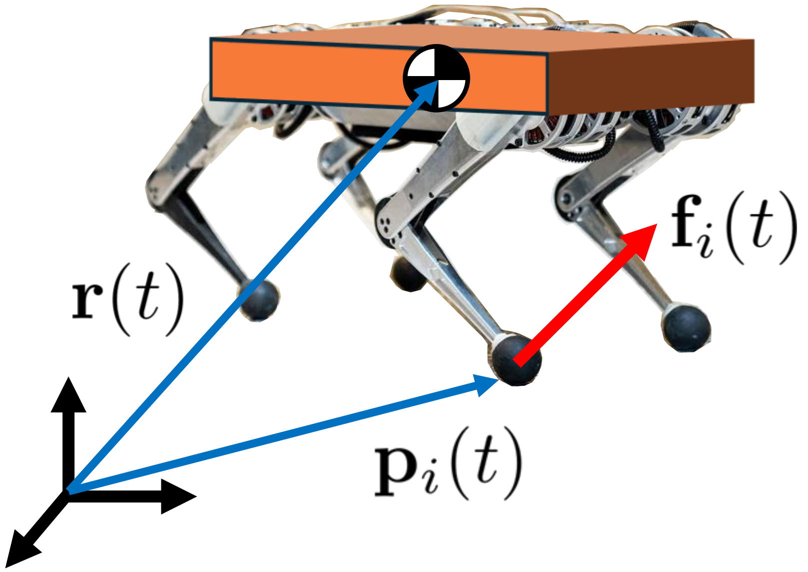


四脚ロボットの制御基礎

SRB詳細

ロボットを**質量m,慣性テンソルI**の**剛体**に近似

Single Rigid Body(SRB)



$$m\ddot{\mathbf{r}}(t) = \sum_{i=1}^{4} \mathbf{f}_i(t) - m\mathbf{g}$$

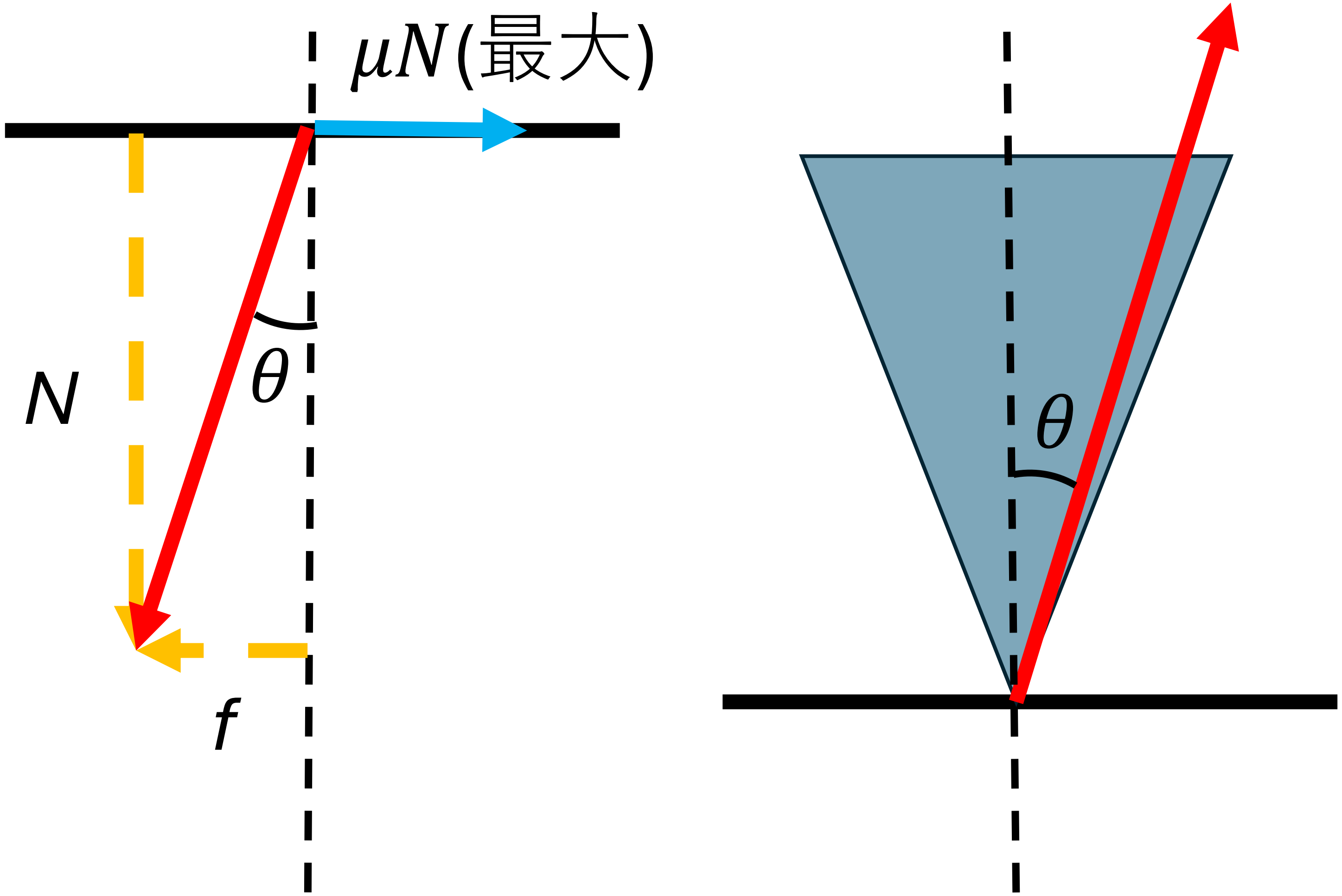
$$I\dot{\boldsymbol{\omega}}(t) + \boldsymbol{\omega}(t) \times I\boldsymbol{\omega}(t) = \sum_{i=1}^{4} (\mathbf{p}_i(t) - \mathbf{r}(t)) \times \mathbf{f}_i(t)$$

注意:脚部の重さが全体の**10%以下**の際に利用する

二足歩行ロボットの場合は使えない
Centroidal Dynamics,SLIP,etc...

四脚ロボットの制御基礎

摩擦円錐



$$f < \mu N \leftrightarrow \frac{f}{N} < \mu$$

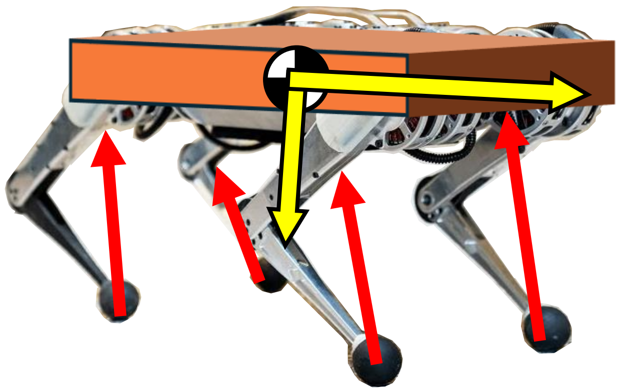
$$\tan\theta = \frac{f}{N} < \mu$$

$$\theta < \tan^{-1}\mu$$

滑りが発生する限界領域を**円錐**で表す

四脚ロボットの課題

歩行条件



重力+慣性力=床反力

どうやって**床反力**を求めるか？？

方法１：
床反力を分配する規則をあらかじめ設定

方法2:
最適化問題を解いて床反力を分配

論文

Dynamic Locomotion in the MIT Cheetah 3 Through Convex Model-Predictive Control

Jared Di Carlo[1], Patrick M. Wensing[2], Benjamin Katz[3], Gerardo Bledt[1,3], and Sangbae Kim[3]

論文の主張：MPCを用いた四脚ロボット制御の定式化

重要な結果：

$$\frac{d}{dt}\begin{bmatrix}\hat{\Theta}\\ \hat{p}\\ \hat{\omega}\\ \hat{\dot{p}}\end{bmatrix} = \begin{bmatrix} O_3 & R_z^T(\psi) & O_3 & O_3 \\ O_3 & O_3 & O_3 & I_3 \\ O_3 & O_3 & O_3 & O_3 \\ O_3 & O_3 & O_3 & O_3 \end{bmatrix}\begin{bmatrix}\hat{\Theta}\\ \hat{p}\\ \hat{\omega}\\ \hat{\dot{p}}\end{bmatrix} + \begin{bmatrix} O_3 & \cdots & O_3 \\ O_3 & \cdots & O_3 \\ \hat{I}^{-1}[r_1]_\times & \cdots & \hat{I}^{-1}[r_n]_\times \\ \frac{1}{m}I_3 & \cdots & \frac{1}{m}I_3 \end{bmatrix}\begin{bmatrix} f_1 \\ \vdots \\ f_n \end{bmatrix} + \begin{bmatrix} 0\\0\\0\\g\end{bmatrix}$$

$$\min_{x,u} \sum_{i=0}^{k-1} \|x_{i+1} - x_{i+1,\mathrm{ref}}\|_{Q_i} + \|u_i\|_{R_i}$$

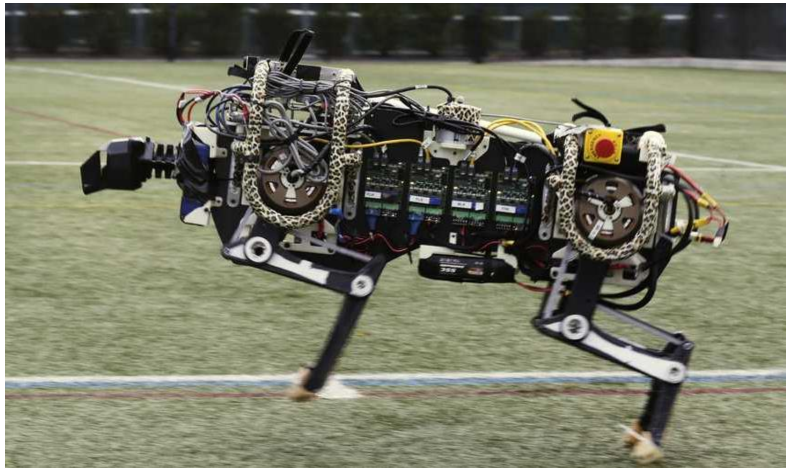
subject to

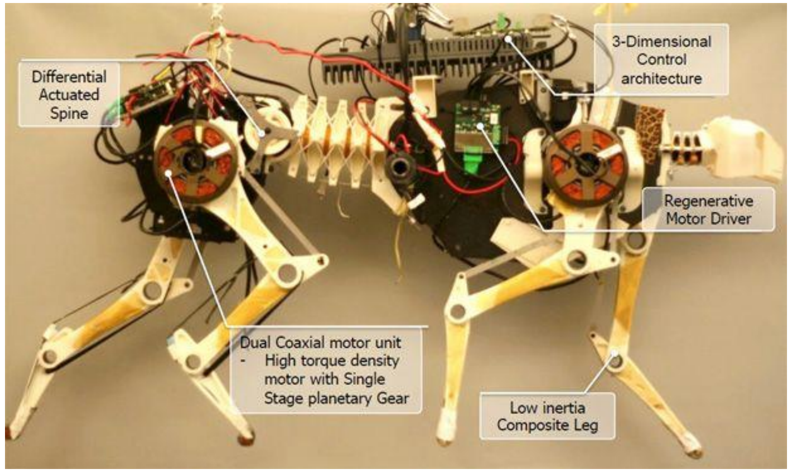
$$x_{i+1} = A_i x_i + B_i u_i,\ i = 0 \dots k-1$$
$$c_i \le C_i u_i \le c_i',\ i = 0 \dots k-1$$
$$D_i u_i = 0,\ i = 0 \dots k-1$$

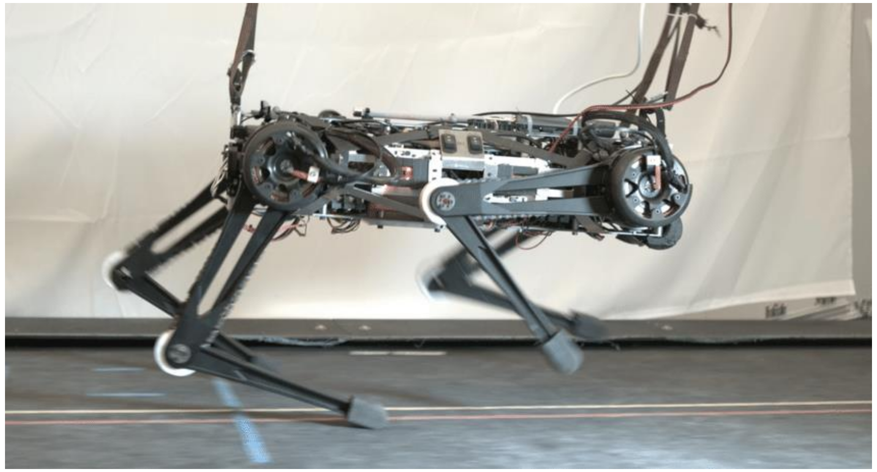
Dynamic Locomotion in the MIT Cheetah 3 Through Convex Model-Predictive Control

Jared Di Carlo[1], Patrick M. Wensing[2], Benjamin Katz[3], Gerardo Bledt[1,3], and Sangbae Kim[3]

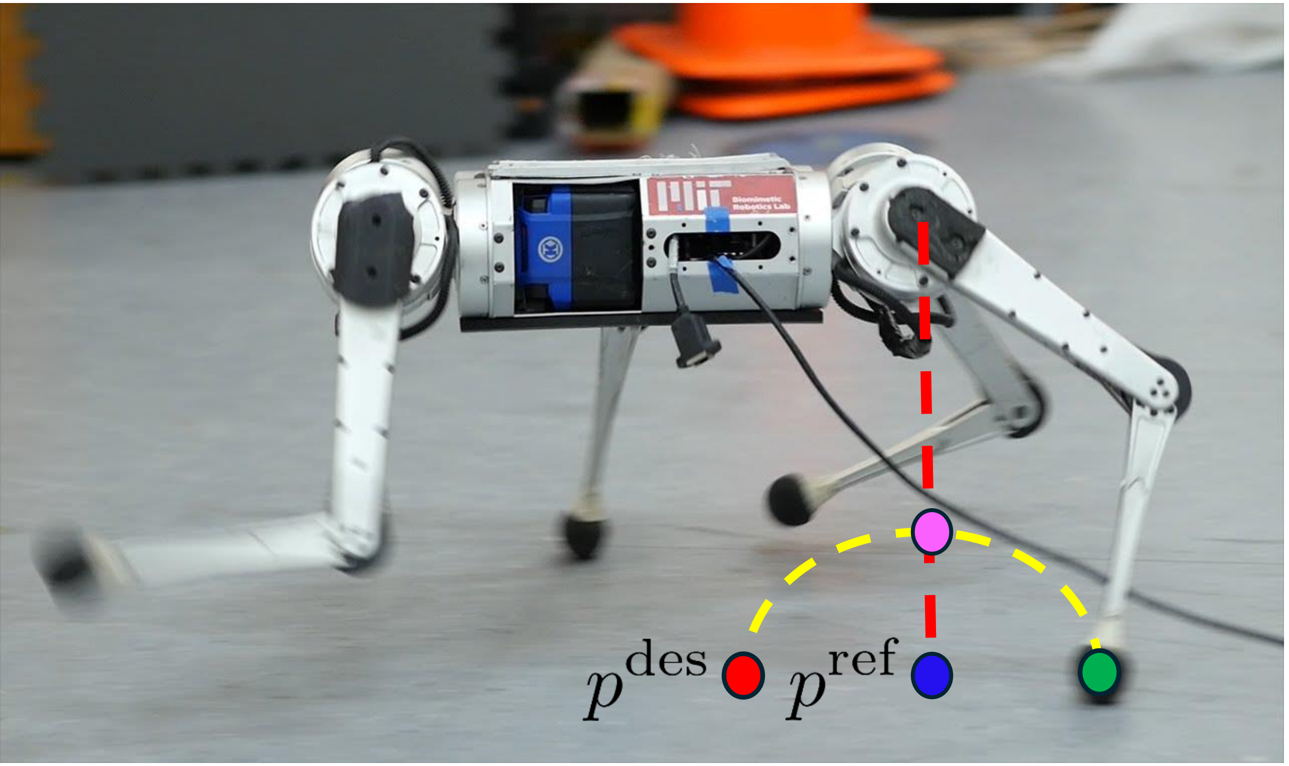
MIT Cheetah




Differential Actuated Spine
3-Dimensional Control architecture
Regenerative Motor Driver
Dual Coaxial motor unit
- High torque density motor with Single Stage planetary Gear
Low inertia Composite Leg



遊脚制御



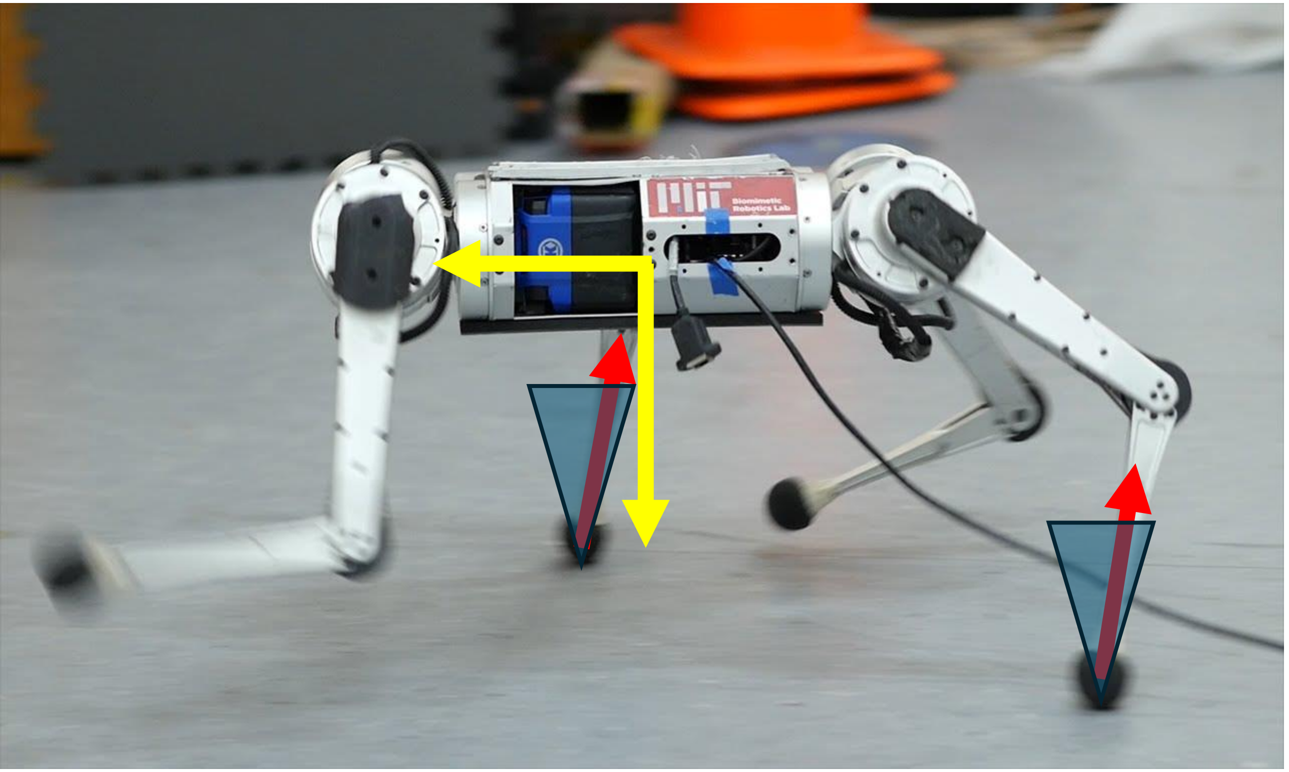
https://www.youtube.com/watch?v=-BqNl3AtPVw

$$p^{\mathrm{des}} = p^{\mathrm{ref}} + \underbrace{v^{\mathrm{CoM}}\frac{\Delta t}{2}}_{\textbf{Raibert Heuristic}}$$

任意のz軸方向高さを決めて**Spline補間**

$$\boxed{\begin{aligned}\tau_i &= J_i^T[K_p({}_{\mathcal{B}}p_{i,ref} - {}_{\mathcal{B}}p_i) + K_d({}_{\mathcal{B}}v_{i,ref} - {}_{\mathcal{B}}v_i)] + \tau_{i,ff} \\ \tau_{i,ff} &= J_i^T\Lambda_i({}_{\mathcal{B}}\ddot{a}_{i,ref} - J_i\dot{q}_i) + C_i\dot{q}_i + G_i\end{aligned}}$$

支持脚制御



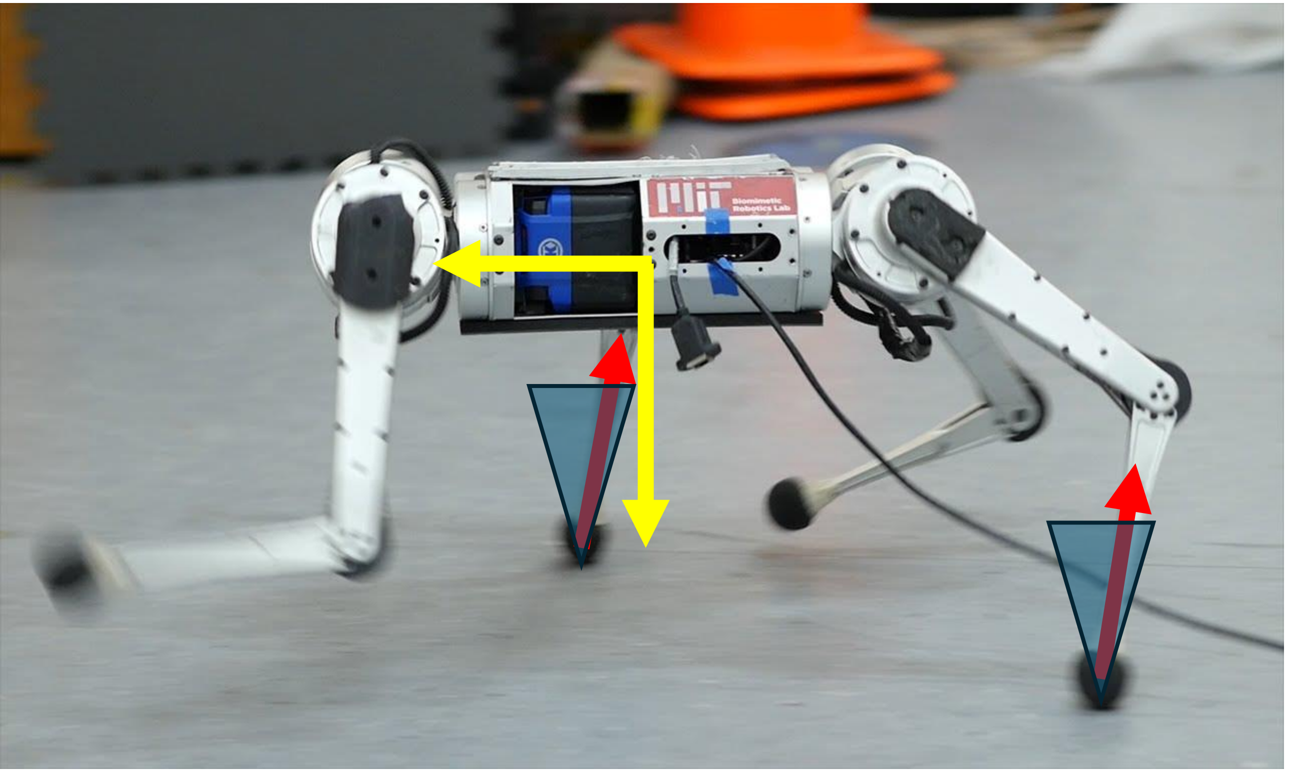
https://www.youtube.com/watch?v=-BqNl3AtPVw

$$\tau_i = J_i^T R_i f_i$$

MPCを利用して求める

まず四脚ロボットが満たす
状態方程式と**評価関数**を求める

支持脚制御　STEP1:状態方程式を求める



https://www.youtube.com/watch?v=-BqNl3AtPVw

$$\begin{cases} \dot{p} = \frac{1}{m}\sum_{i=1}^{4} f_i - g \\ \frac{d}{dt}(I\omega) = \sum_{i=1}^{4} r_i \times f_i \\ \omega = R_z(\psi)R_y(\theta)R_x(\phi)\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \end{cases}$$

非線形要素をできるだけなくす

支持脚制御　STEP1:状態方程式を求める

$$\begin{cases} \dot{p} = \frac{1}{m}\sum_{i=1}^{4} f_i - g \\ \frac{d}{dt}(I\omega) = \sum_{i=1}^{4} r_i \times f_i \\ \omega = R_z(\psi)R_y(\theta)R_x(\phi)\begin{bmatrix}\dot{\phi}\\ \dot{\theta}\\ \dot{\psi}\end{bmatrix} \end{cases}$$

$$\begin{cases} \dot{p} = \frac{1}{m}\sum_{i=1}^{4} f_i - g \\ \dot{\omega} = I^{-1}(\sum_{i=1}^{4} r_i \times f_i) \\ \dot{\Theta} = \begin{bmatrix}\dot{\phi}\\ \dot{\theta}\\ \dot{\psi}\end{bmatrix} = R_z(\psi)^T\omega \end{cases}$$

$$\begin{aligned} \frac{d}{dt}(I\omega) &= I\dot{\omega} + \omega \times (I\omega) \\ &\approx I\dot{\omega} \end{aligned}$$

$$\begin{aligned} \omega &= R_z(\psi)R_y(\theta)R_x(\phi)\begin{bmatrix}\dot{\phi}\\ \dot{\theta}\\ \dot{\psi}\end{bmatrix} \\ &\approx R_z(\psi)\begin{bmatrix}\dot{\phi}\\ \dot{\theta}\\ \dot{\psi}\end{bmatrix} \end{aligned}$$

支持脚制御　STEP1:状態方程式を求める

$$\begin{cases} \dot{p} = \frac{1}{m}\sum_{i=1}^{4} f_i - g \\ \dot{\omega} = I^{-1}(\sum_{i=1}^{4} r_i \times f_i) \\ \dot{\Theta} = \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = R_z(\psi)^T \omega \end{cases}$$

$$\hat{I} = R_z(\psi)_{\mathcal{B}} I R_z(\psi)^T$$

$$\frac{d}{dt}\begin{bmatrix} \hat{\Theta} \\ \hat{p} \\ \hat{\omega} \\ \hat{\dot{p}} \end{bmatrix} = \begin{bmatrix} O_3 & R_z^T(\psi) & O_3 & O_3 \\ O_3 & O_3 & O_3 & I_3 \\ O_3 & O_3 & O_3 & O_3 \\ O_3 & O_3 & O_3 & O_3 \end{bmatrix} \begin{bmatrix} \hat{\Theta} \\ \hat{p} \\ \hat{\omega} \\ \hat{\dot{p}} \end{bmatrix} + \begin{bmatrix} O_3 & \cdots & O_3 \\ O_3 & \cdots & O_3 \\ \hat{I}^{-1}[r_1]_\times & \cdots & \hat{I}^{-1}[r_n]_\times \\ \frac{1}{m}I_3 & \cdots & \frac{1}{m}I_3 \end{bmatrix} \begin{bmatrix} f_1 \\ \vdots \\ f_n \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ g \end{bmatrix}$$

$$\dot{\mathbf{x}}(t) = \mathbf{A_c}(\psi)x(t) + \mathbf{B_c}(\mathbf{r_1}, \ldots, \mathbf{r_n}, \psi)\mathbf{u}(t)$$

支持脚制御　STEP2:評価関数を求める

$$\min_{x,u} \sum_{i=0}^{k-1} \|x_{i+1} - x_{i+1,\mathrm{ref}}\|_{Q_i} + \|u_i\|_{R_i}$$

subject to

$$x_{i+1} = A_i x_i + B_i u_i,\ i = 0 \ldots k-1$$

$$c_i \leq C_i u_i \leq c'_i,\ i = 0 \ldots k-1$$

$$D_i u_i = 0,\ i = 0 \ldots k-1$$

摩擦円錐の拘束

遊脚のuを0

$$\mathrm{f}_{\mathrm{min}} \leq f_z \leq f_{\mathrm{max}}$$

$$-\mu f_z \leq f_x \leq \mu f_z$$

$$-\mu f_z \leq f_y \leq \mu f_z$$

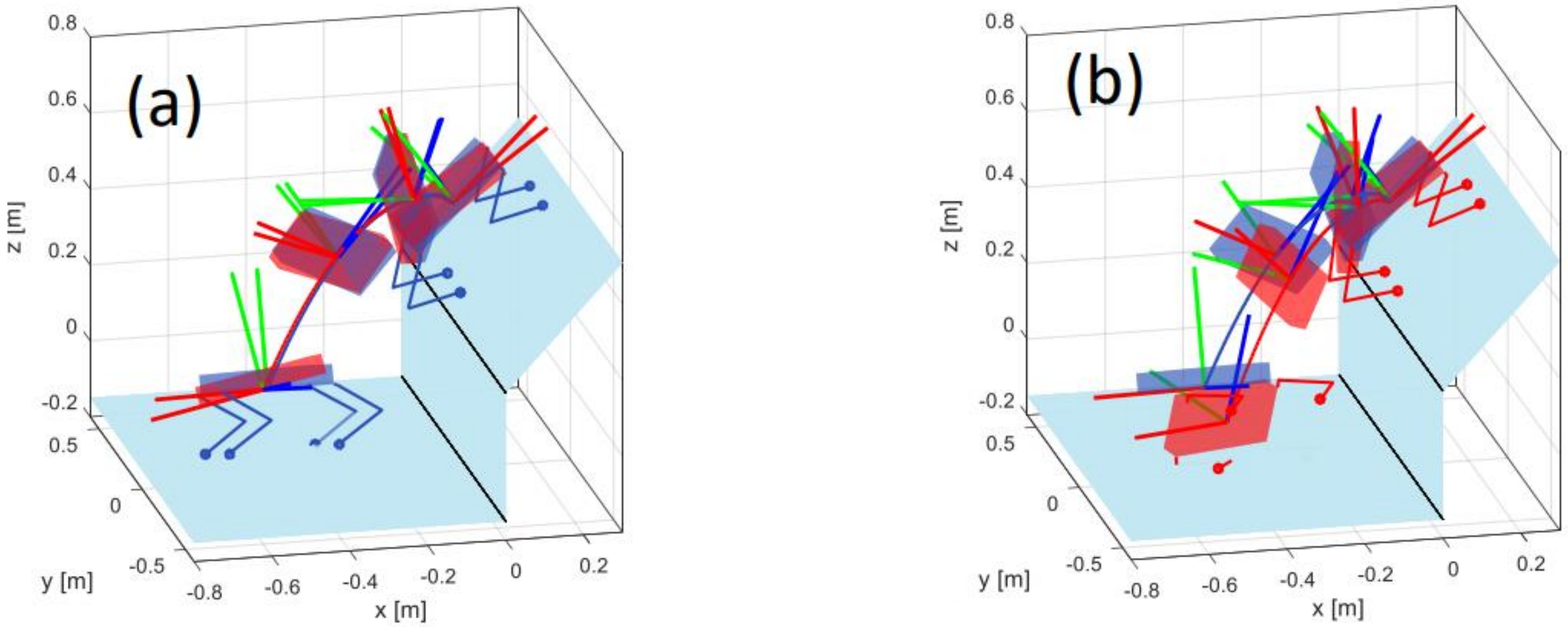
Real-time Model Predictive Control for Versatile Dynamic Motions in Quadrupedal Robots

Yanran Ding, Abhishek Pandala, and Hae-Won Park

論文の主張： リー代数を用いてアクロバットな動きを可能にする

重要な結果：

$$\omega_{k+1} = \omega_k + {}^B I^{-1} \left(R_k^T \tau_k - \hat{\omega}_k^B I_{\omega_k} \right) \Delta t$$
$$R_{k+1} = R_k \exp(\hat{\omega}_k \Delta t)$$



Real-time Model Predictive Control for Versatile Dynamic Motions in Quadrupedal Robots

Yanran Ding, Abhishek Pandala, and Hae-Won Park

Real-time Model Predictive Control for Versatile Dynamic Motions in Quadrupedal Robots

Yanran Ding, Abhishek Pandala, and Hae-Won Park

MIT Cheetah

問題点

オイラー角　　　：ジンバルロック
クォータニオン：Unwind

問題が起きない回転法を探す

リー代数を用いた回転表現

通常の回転行列は**9つの成分**で表される

$$SO(3) = \left\{ R \in \mathbb{R}^{3\times 3} \mid R^T R = I, \det(R) = 1 \right\} \qquad R = \begin{bmatrix} r_1 & r_2 & r_3 \\ r_4 & r_5 & r_6 \\ r_7 & r_8 & r_9 \end{bmatrix}$$

リー代数を用いた回転は**3つの成分**で表される

$$\mathfrak{so}(3) = \left\{ [\omega_\times] \mid \omega \in \mathbb{R}^3 \right\} \qquad [\omega_\times] = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}$$

注意：角速度ではない

リー代数を用いた回転表現

$$SO(3) = \{R \in \mathbb{R}^{3\times 3} \mid R^T R = I, \det(R) = 1\}$$
$$\mathfrak{so}(3) = \{[\omega_\times] \mid \omega \in \mathbb{R}^3\}$$

$$\begin{tikzcd} SO(3) \arrow[r, bend left, "\text{対数写像}"] & \mathfrak{so}(3) \arrow[l, bend left, "\text{指数写像}"] \arrow[r, "{[\cdot]^\vee}"] & \omega \end{tikzcd}$$

回転誤差を含む評価関数

roll, pitch回転を伴う運動でも近似誤差の影響を無くす

→ **アクロバットな動きの制御**

$$\min \quad \sum_{k=1}^{N-1} \ell(x_k, u_k) + \ell_T(x_N)$$

$$\ell(x_k, u_k) = e_{u_k}^T R_u e_{u_k} + e_{p_k}^T Q_p e_{p_k} + e_{\dot{p}_k}^T Q_{\dot{p}} e_{\dot{p}_k} + \boxed{e_{\omega_k}^T Q_\omega e_{\omega_k}} + \boxed{e_{R_k}^T Q_R e_{R_k}}$$

$$e_{R_k} = \log(R_{d,k}^T \cdot R_k)^\vee$$

$$e_{\omega_k} = \omega_k - R_k^T R_{d,k} \omega_{d,k}$$

Real-time Model Predictive Control for Versatile Dynamic Motions in Quadrupedal Robots

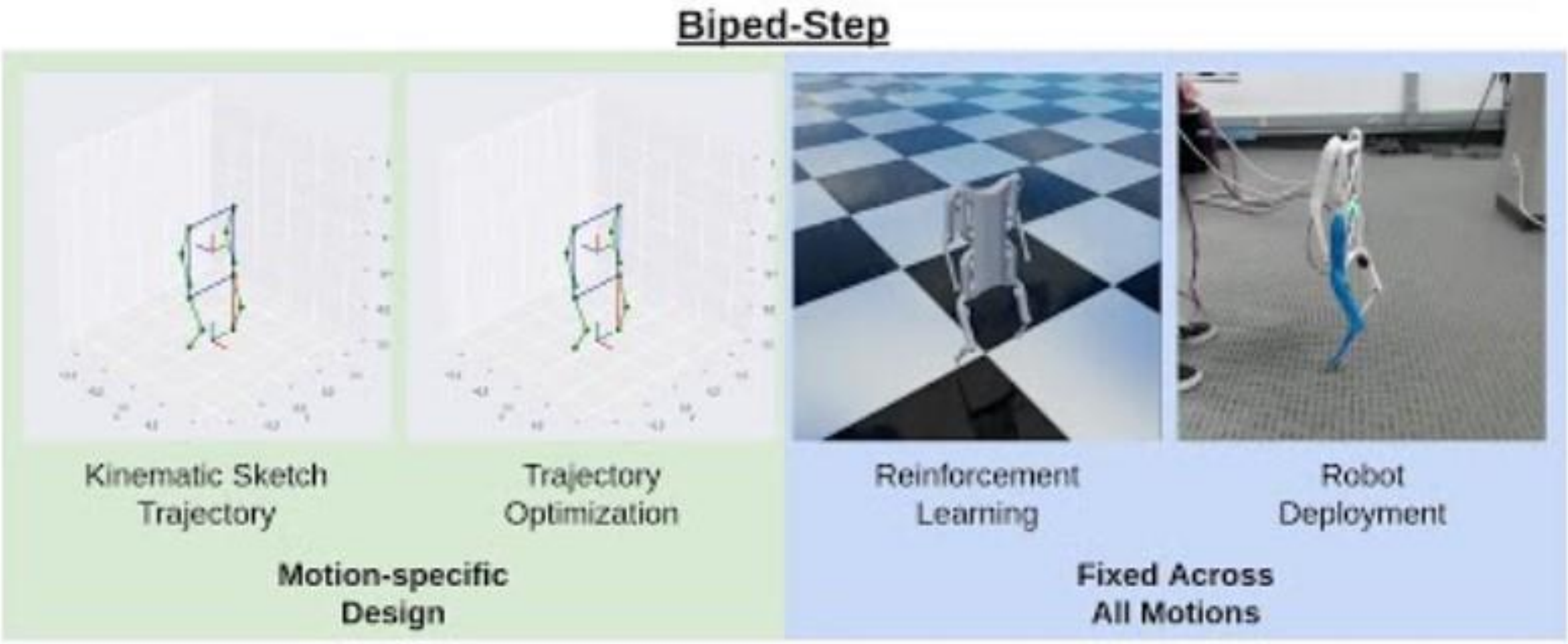
Yanran Ding, Abhishek Pandala, and Hae-Won Park

$$e_{R_k} = \log\underbrace{\underbrace{\underbrace{(R_{d,k}^T \cdot R_k)}_{SO(3)}}_{\mathfrak{so}(3)}{}^{\vee}}_{\mathbb{R}^3}$$

$$\boxed{\begin{aligned} \omega_{k+1} &= \omega_k +^B I^{-1}\left(R_k^T \tau_k - \hat{\omega}_k^B I_{\omega_k}\right)\Delta t \\ R_{k+1} &= R_k \exp(\hat{\omega}_k \Delta t) \end{aligned}}$$

OPT-Mimic: Imitation of Optimized Trajectories for Dynamic Quadruped Behaviors

Yuni Fuchioka[1], Zhaoming Xie[1,2], and Michiel van de Panne[1]



OPT-Mimic: Imitation of Optimized Trajectories for Dynamic Quadruped Behaviors

Yuni Fuchioka[1], Zhaoming Xie[1,2], and Michiel van de Panne[1]

$$p^+ = p + \dot{p}\Delta t$$

$$\dot{p}^+ = \dot{p} + \left(\frac{1}{m} \sum_i f_i + g \right) \Delta t$$

$$\begin{cases} R^+ = R e^{([\omega_\times]\Delta t)} \\ \omega^+ = \omega + {}^B I^{-1} \left(R^T \left(\sum_i (p_i - p) \times f_i \right) - [\omega_\times]^B I_\omega \right) \Delta t \end{cases}$$

Real-time Model Predictive Control for Versatile Dynamic Motions in Quadrupedal Robots

Yanran Ding, Abhishek Pandala, and Hae-Won Park

OPT-Mimic: Imitation of Optimized Trajectories for Dynamic Quadruped Behaviors

Yuni Fuchioka[1], Zhaoming Xie[1,2], and Michiel van de Panne[1]

$$(p_i)_z \geq 0$$

地面にめり込まない

$$(f_i)_z (p_i)_z = 0$$

支持脚か判定

$$(f_i)_z ((p_i)_x^+ - (p_i)_x) = 0$$
$$(f_i)_z ((p_i)_y^+ - (p_i)_y) = 0$$

接触トルクが常に0

$$|(f_i)_x| \leq \mu (f_i)_z$$
$$|(f_i)_y| \leq \mu (f_i)_z$$
$$0 \leq (f_i)_z \leq ((f_i)_z)_{\max}$$

摩擦円錐

$$\left\| \begin{bmatrix} ({}^{B_i}p_i)_x \\ ({}^{B_i}p_i)_z \end{bmatrix} \right\|_1 \leq l_{\text{leg}}$$

脚先が関節可動域内

$$({}^{B_i}p_i)_y = 0$$

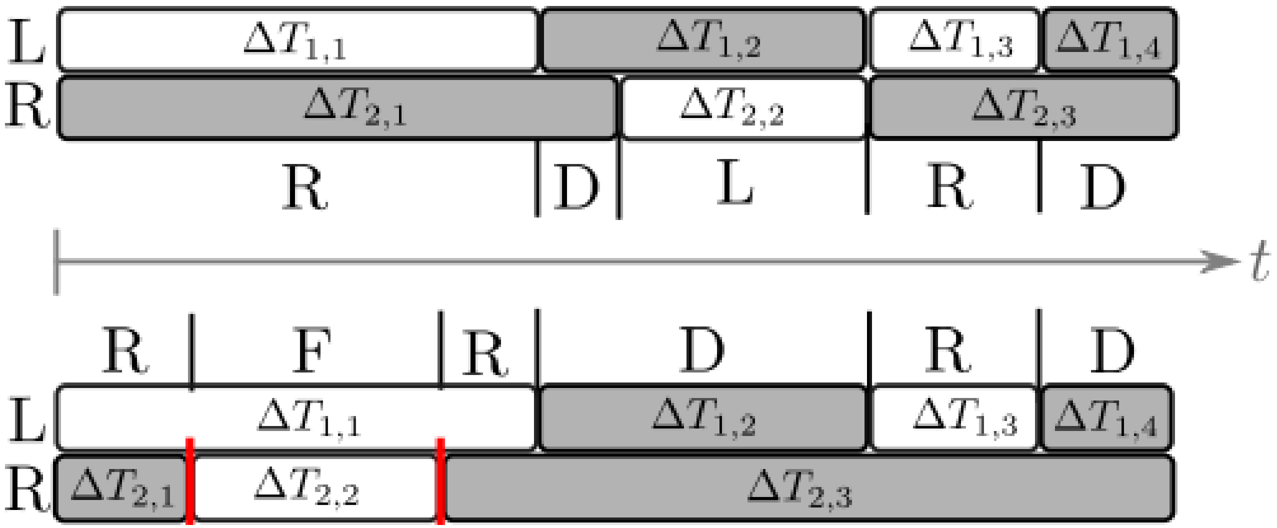
脚が下方に伸びる

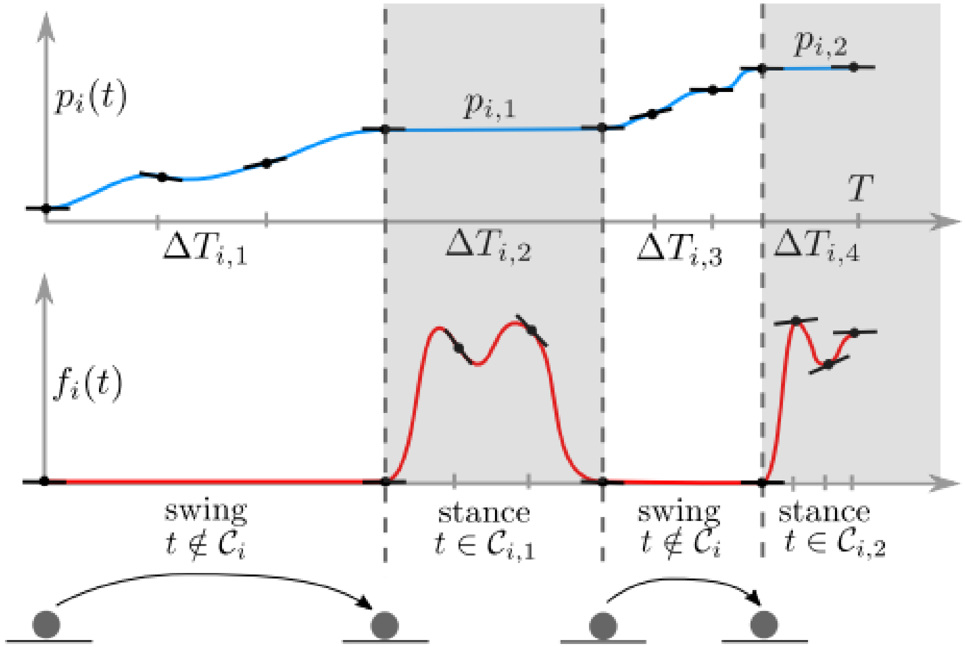
Gait and Trajectory Optimization for Legged Systems through Phase-based End-Effector Parameterization

Alexander W. Winkler[1], C. Dario Bellicoso[2], Marco Hutter[2], Jonas Buchli[1]

論文の主張：より一般的なGait生成を可能にする

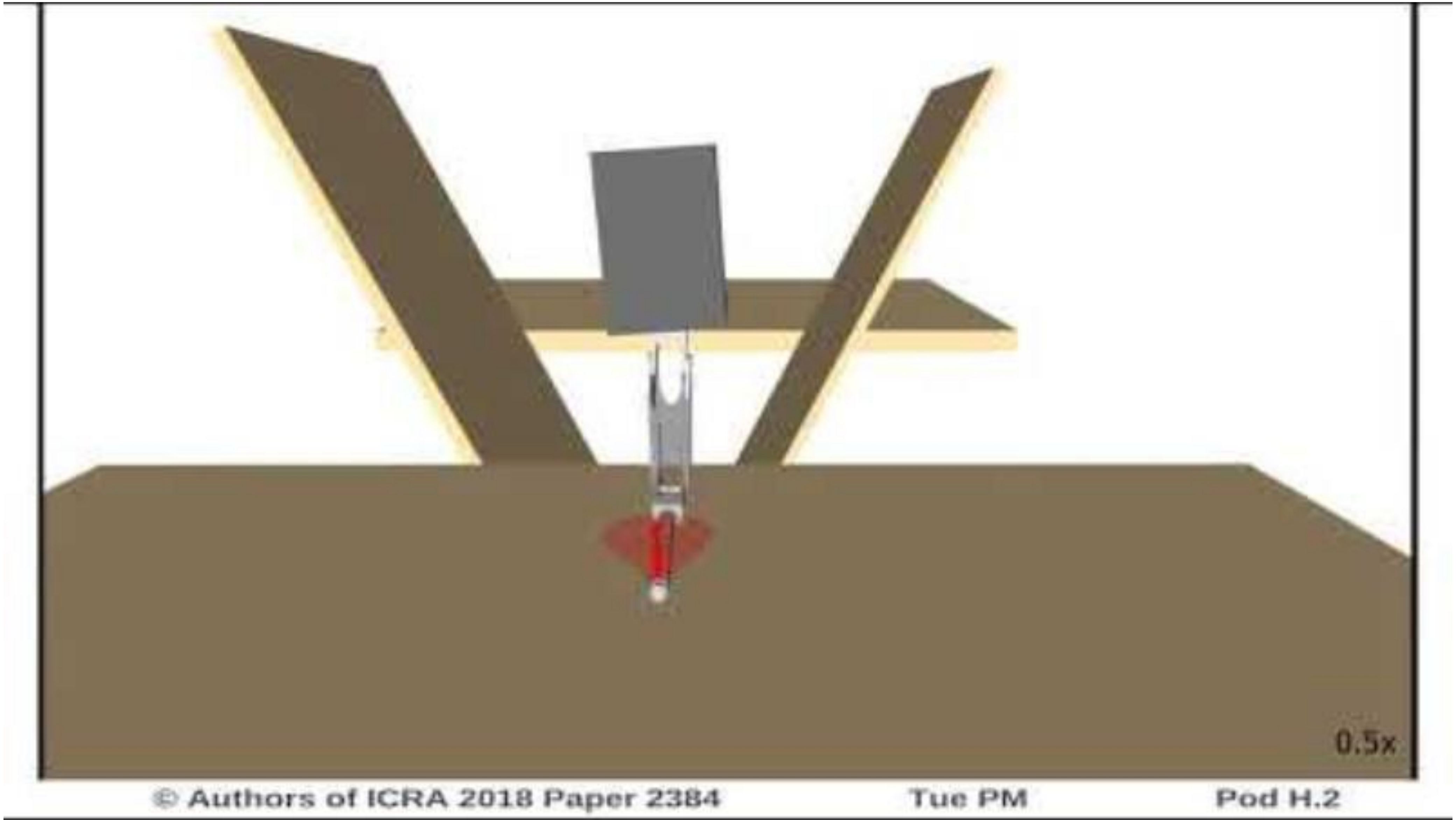
重要な結果：





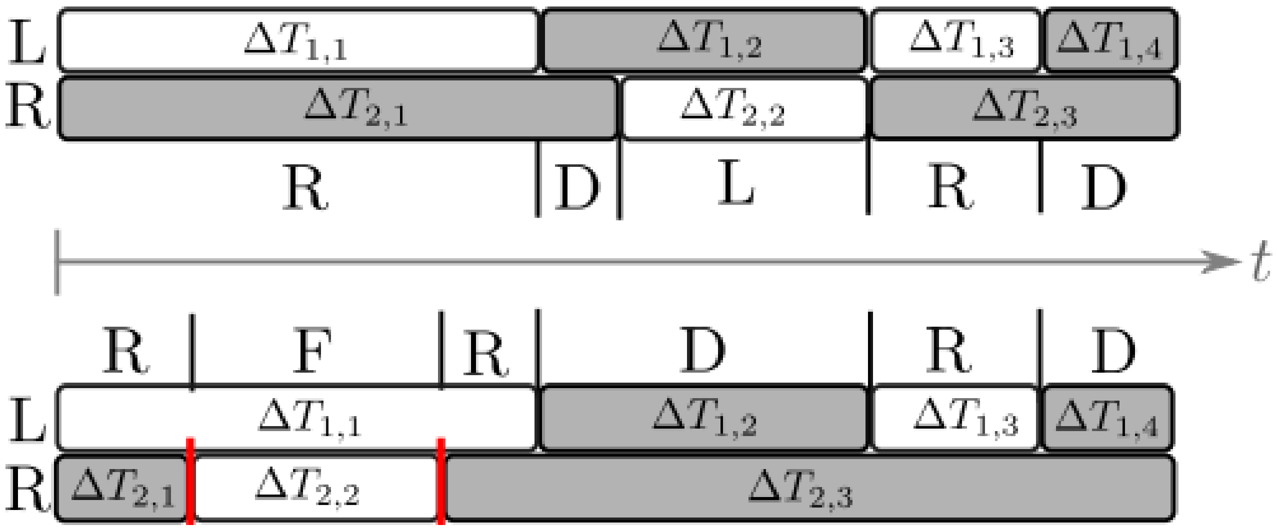
Gait and Trajectory Optimization for Legged Systems through Phase-based End-Effector Parameterization

Alexander W. Winkler[1], C. Dario Bellicoso[2], Marco Hutter[2], Jonas Buchli[1]



Gait and Trajectory Optimization for Legged Systems through Phase-based End-Effector Parameterization

Alexander W. Winkler[1], C. Dario Bellicoso[2], Marco Hutter[2], Jonas Buchli[1]

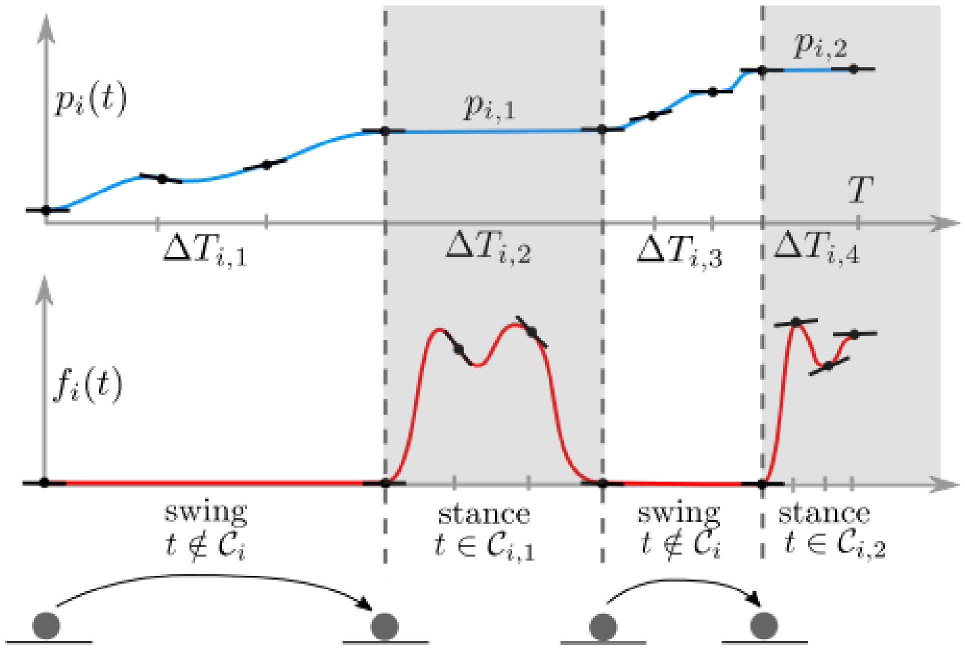


反力最適化(連続最適化)+支持脚位置最適化(整数最適化)

反力最適化(連続最適化)+支持脚時間最適化(整数最適化)

Gait and Trajectory Optimization for Legged Systems through Phase-based End-Effector Parameterization

Alexander W. Winkler[1], C. Dario Bellicoso[2], Marco Hutter[2], Jonas Buchli[1]



計算効率化

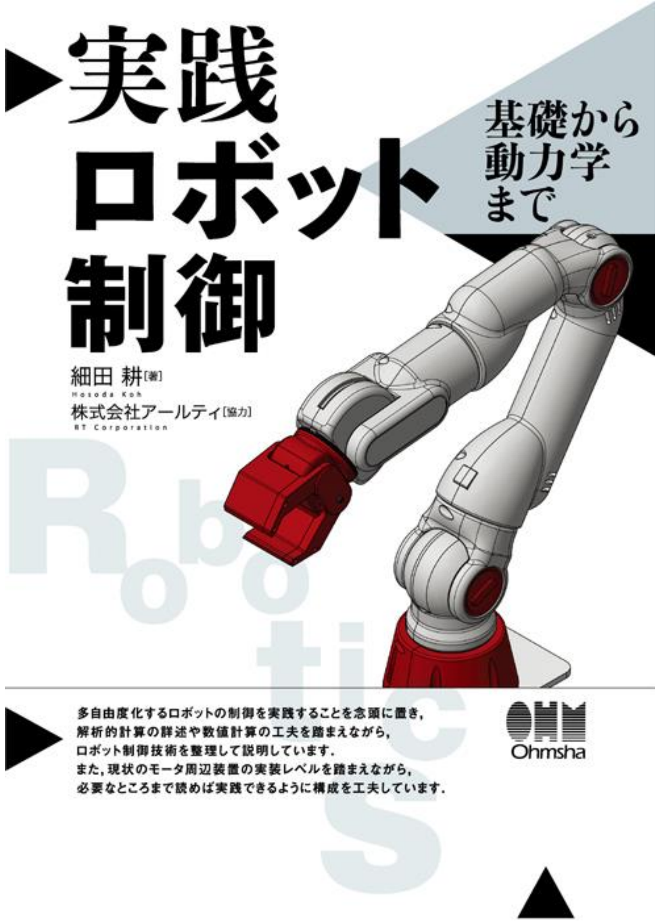
遊脚時の反力fは0

支持脚時の脚先位置pは地面

参考資料

- Bledt, Gerardo, et al. "Mit cheetah 3: Design and control of a robust, dynamic quadruped robot." *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018.
- Di Carlo, Jared, et al. "Dynamic locomotion in the mit cheetah 3 through convex model-predictive control." *2018 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2018.
- Winkler, Alexander W., et al. "Gait and trajectory optimization for legged systems through phase-based end-effector parameterization." *IEEE Robotics and Automation Letters* 3.3 (2018): 1560-1567.
- Ding, Yanran, Abhishek Pandala, and Hae-Won Park. "Real-time model predictive control for versatile dynamic motions in quadrupedal robots." *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019.

参考資料



実践
ロボット
制御
基礎から
動力学
まで
細田 耕[著]
Hosoda Koh
株式会社アールティ[協力]
RT Corporation
Robotics
多自由度化するロボットの制御を実践することを念頭に置き、
解析的計算の詳述や数値計算の工夫を踏まえながら、
ロボット制御技術を整理して説明しています。
また、現状のモータ周辺装置の実装レベルを踏まえながら、
必要なところまで読めば実践できるように構成を工夫しています。
Ohmsha

コンピュータ制御
機械システムシリーズ
10
ロボット制御基礎論
京都大学名誉教授
立命館大学教授
工学博士
吉 川 恒 夫 著
コロナ社

参考資料


不整地移動
ロボティクス
永谷 圭司【編著】
石上 玄也　遠藤 大輔　永岡 健司　遠藤　玄
程島 竜一　亀川 哲志　田中 基康【共著】
コロナ社


ヒューマノイド
ロボット
改訂2版
梶田 秀司 編著
Ohmsha