

# 軌道最適化による動作生成 リファレンスマニュアル

平成 31 年 2 月 12 日

室岡雅樹

murooka@jsk.t.u-tokyo.ac.jp

## 目 次

<b>1</b>	<b>軌道最適化による動作生成の基礎</b>	<b>1</b>
1.1	タスク関数のノルムを最小にするコンフィギュレーションの探索	1
1.2	コンフィギュレーション二次形式の正則化項の追加	2
1.3	コンフィギュレーション更新量の正則項の追加	3
1.4	ソースコードと数式の対応	3
1.5	章の構成	4
<b>2</b>	<b>コンフィギュレーションとタスク関数</b>	<b>4</b>
2.1	瞬時コンフィギュレーションと瞬時タスク関数	4
2.2	軌道コンフィギュレーションと軌道タスク関数	21
<b>3</b>	<b>勾配を用いた制約付き非線形最適化</b>	<b>28</b>
3.1	逐次二次計画法	28
3.2	複数解候補を用いた逐次二次計画法	30
3.2.1	複数解候補を用いた逐次二次計画法の理論	30
3.2.2	複数解候補を用いた逐次二次計画法の実装	34
<b>4</b>	<b>動作生成の拡張</b>	<b>35</b>
4.1	マニピュレーションの動作生成	35
4.2	B スプラインを用いた関節軌道生成	46
4.2.1	B スプラインを用いた関節軌道生成の理論	46
4.2.2	B スプラインを用いた関節軌道生成の実装	55
4.3	B スプラインを用いた動的動作の生成	67
<b>5</b>	<b>補足</b>	<b>90</b>
5.1	既存のロボット基礎クラスの拡張	90
5.2	環境と接触するロボットの関節・リンク構造	93

5.3	irteus の inverse-kinematics 互換関数 . . . . .	94
5.4	関節トルク勾配の計算 . . . . .	98

# 1 軌道最適化による動作生成の基礎

## 1.1 タスク関数のノルムを最小にするコンフィギュレーションの探索

$q \in \mathbb{R}^{n_q}$  を設計対象のコンフィギュレーションとする．例えば一般の逆運動学計算では， $q$  はある瞬間のロボットの関節角度を表すベクトルで，コンフィギュレーションの次元  $n_q$  はロボットの関節自由度数となる．

動作生成問題を，所望のタスクに対応するタスク関数  $e(q) : \mathbb{R}^{n_q} \rightarrow \mathbb{R}^{n_e}$  について，次式を満たす  $q$  を得ることとして定義する．

$$e(q) = 0 \quad (1.1)$$

例えば一般の逆運動学計算では， $e(q)$  はエンドエフェクタの目標位置姿勢と現在位置姿勢の差を表す 6 次元ベクトルである．非線形方程式 (1.1) の解を解析的に得ることは難しく，反復計算による数値解法が採られる．式 (1.1) が解をもたないときでも最善のコンフィギュレーションを得られるように一般化すると，式 (1.1) の求解は次の最適化問題として表される<sup>1</sup>．

$$\min_q F(q) \quad (1.2a)$$

$$\text{where } F(q) \stackrel{\text{def}}{=} \frac{1}{2} \|e(q)\|^2 \quad (1.2b)$$

コンフィギュレーションが最小値  $q_{min}$  と最大値  $q_{max}$  の間に含まれる必要があるとき，逆運動学計算は次の制約付き非線形最適化問題として表される．

$$\min_q F(q) \quad \text{s.t.} \quad q_{min} \leq q \leq q_{max} \quad (1.3)$$

例えば一般の逆運動学計算では， $q_{min}, q_{max}$  は関節角度の許容範囲の最小値，最大値を表す．以降では，式 (1.3) の制約を，より一般の形式である線形等式制約，線形不等式制約として次式のように表す<sup>2</sup>．

$$\min_q F(q) \quad (1.4a)$$

$$\text{s.t.} \quad Aq = \bar{b} \quad (1.4b)$$

$$Cq \geq \bar{d} \quad (1.4c)$$

制約付き非線形最適化問題の解法のひとつである逐次二次計画法では，次の二次計画問題の最適解として得られる  $\Delta q_k^*$  を用いて， $q_{k+1} = q_k + \Delta q_k^*$  として反復更新することで，式 (1.4) の最適解を導出する<sup>3</sup>．

$$\min_{\Delta q_k} F(q_k) + \nabla F(q_k)^T \Delta q_k + \frac{1}{2} \Delta q_k^T \nabla^2 F(q_k) \Delta q_k \quad (1.5a)$$

$$\text{s.t.} \quad A \Delta q_k = \bar{b} - Aq_k \quad (1.5b)$$

$$C \Delta q_k \geq \bar{d} - Cq_k \quad (1.5c)$$

<sup>1</sup> 任意の半正定値行列  $W$  に対して， $\|e(q)\|_W^2 = e(q)^T W e(q) = e(q)^T S^T S e(q) = \|S e(q)\|^2$  を満たす  $S$  が必ず存在するので，式 (1.2b) は任意の重み付きノルムを表現可能である．

<sup>2</sup> 式 (1.3) における関節角度の最小値，最大値に関する制約は次式のように表される．

$$\begin{aligned} q_{min} &\leq q \leq q_{max} \\ \Leftrightarrow \begin{pmatrix} I \\ -I \end{pmatrix} q &\geq \begin{pmatrix} q_{min} \\ -q_{max} \end{pmatrix} \end{aligned}$$

<sup>3</sup> 式 (1.5a) は  $F(q)$  を  $q_k$  の周りでテーラー展開し三次以下の項を省略したものに一致する．逐次二次計画法については，以下の書籍の 18 章で詳しく説明されている．

Numerical optimization, S. Wright and J. Nocedal, Springer Science, vol. 35, 1999, [http://www.xn--vjq503akpco3w.top/literature/Nocedal\\_Wright\\_Numerical\\_optimization\\_v2.pdf](http://www.xn--vjq503akpco3w.top/literature/Nocedal_Wright_Numerical_optimization_v2.pdf).

$\nabla F(\mathbf{q}_k), \nabla^2 F(\mathbf{q}_k)$  はそれぞれ,  $F(\mathbf{q}_k)$  の勾配, ヘッセ行列<sup>4</sup>で, 次式で表される.

$$\nabla F(\mathbf{q}) = \left( \frac{\partial e(\mathbf{q})}{\partial \mathbf{q}} \right)^T e(\mathbf{q}) \quad (1.6a)$$

$$= \mathbf{J}(\mathbf{q})^T e(\mathbf{q}) \quad (1.6b)$$

$$\nabla^2 F(\mathbf{q}) = \sum_{i=1}^m e_i(\mathbf{q}) \nabla^2 e_i(\mathbf{q}) + \left( \frac{\partial e(\mathbf{q})}{\partial \mathbf{q}} \right)^T \frac{\partial e(\mathbf{q})}{\partial \mathbf{q}} \quad (1.6c)$$

$$\approx \left( \frac{\partial e(\mathbf{q})}{\partial \mathbf{q}} \right)^T \frac{\partial e(\mathbf{q})}{\partial \mathbf{q}} \quad (1.6d)$$

$$= \mathbf{J}(\mathbf{q})^T \mathbf{J}(\mathbf{q}) \quad (1.6e)$$

ただし,  $e_i(\mathbf{q})$  ( $i = 1, 2, \dots, m$ ) は  $e(\mathbf{q})$  の  $i$  番目の要素である. 式 (1.6c) から式 (1.6d) への変形では  $e(\mathbf{q})$  の二階微分がゼロであると近似している.  $\mathbf{J}(\mathbf{q}) \stackrel{\text{def}}{=} \frac{\partial e(\mathbf{q})}{\partial \mathbf{q}} \in \mathbb{R}^{n_e \times n_q}$  は  $e(\mathbf{q})$  のヤコビ行列である.

式 (1.6a), 式 (1.6d) から式 (1.5a) の目的関数は次式で表される<sup>5</sup>.

$$\frac{1}{2} \mathbf{e}_k^T \mathbf{e}_k + \mathbf{e}_k^T \mathbf{J}_k \Delta \mathbf{q}_k + \frac{1}{2} \Delta \mathbf{q}_k^T \mathbf{J}_k^T \mathbf{J}_k \Delta \mathbf{q}_k \quad (1.7a)$$

$$= \frac{1}{2} \|\mathbf{e}_k + \mathbf{J}_k \Delta \mathbf{q}_k\|^2 \quad (1.7b)$$

ただし,  $\mathbf{e}_k \stackrel{\text{def}}{=} e(\mathbf{q}_k), \mathbf{J}_k \stackrel{\text{def}}{=} \mathbf{J}(\mathbf{q}_k)$  とした.

結局, 逐次二次計画法で反復的に解かれる二次計画問題 (1.5) は次式で表される.

$$\min_{\Delta \mathbf{q}_k} \frac{1}{2} \Delta \mathbf{q}_k^T \mathbf{J}_k^T \mathbf{J}_k \Delta \mathbf{q}_k + \mathbf{e}_k^T \mathbf{J}_k \Delta \mathbf{q}_k \quad (1.8a)$$

$$\text{s.t. } \mathbf{A} \Delta \mathbf{q}_k = \mathbf{b} \quad (1.8b)$$

$$\mathbf{C} \Delta \mathbf{q}_k \geq \mathbf{d} \quad (1.8c)$$

ここで,

$$\mathbf{b} = \bar{\mathbf{b}} - \mathbf{A} \mathbf{q}_k \quad (1.9)$$

$$\mathbf{d} = \bar{\mathbf{d}} - \mathbf{C} \mathbf{q}_k \quad (1.10)$$

とおいた.

## 1.2 コンフィギュレーション二次形式の正則化項の追加

式 (1.2a) の最適化問題の目的関数を, 次式の  $\hat{F}(\mathbf{q})$  で置き換える.

$$\hat{F}(\mathbf{q}) = F(\mathbf{q}) + F_{reg}(\mathbf{q}) \quad (1.11)$$

$$\text{where } F_{reg}(\mathbf{q}) = \frac{1}{2} \mathbf{q}^T \bar{\mathbf{W}}_{reg} \mathbf{q} \quad (1.12)$$

目的関数  $\hat{F}(\mathbf{q})$  の勾配, ヘッセ行列は次式で表される.

$$\nabla \hat{F}(\mathbf{q}) = \nabla F(\mathbf{q}) + \nabla F_{reg}(\mathbf{q}) \quad (1.13a)$$

$$= \mathbf{J}(\mathbf{q})^T e(\mathbf{q}) + \bar{\mathbf{W}}_{reg} \mathbf{q} \quad (1.13b)$$

$$\nabla^2 \hat{F}(\mathbf{q}) = \nabla^2 F(\mathbf{q}) + \nabla^2 F_{reg}(\mathbf{q}) \quad (1.13c)$$

$$\approx \mathbf{J}(\mathbf{q})^T \mathbf{J}(\mathbf{q}) + \bar{\mathbf{W}}_{reg} \quad (1.13d)$$

<sup>4</sup>式 (1.5a) の  $\nabla^2 F(\mathbf{q}_k)$  の部分は一般にはラグランジュ関数の  $\mathbf{q}_k$  に関するヘッセ行列であるが, 等式・不等式制約が線形の場合は  $F(\mathbf{q}_k)$  のヘッセ行列と等価になる.

<sup>5</sup>式 (1.7b) は, 以下の論文で紹介されている二次計画法によってコンフィギュレーション速度を導出する逆運動学解法における目的関数と一致する.

Feasible pattern generation method for humanoid robots, F. Kanehiro et al., Proceedings of the 2009 IEEE-RAS International Conference on Humanoid Robots, pp. 542-548, 2009.

したがって，式 (1.8) の二次計画問題は次式で表される．

$$\min_{\Delta \mathbf{q}_k} \frac{1}{2} \Delta \mathbf{q}_k^T \left( \mathbf{J}_k^T \mathbf{J}_k + \bar{\mathbf{W}}_{reg} \right) \Delta \mathbf{q}_k + \left( \mathbf{J}_k^T \mathbf{e}_k + \bar{\mathbf{W}}_{reg} \mathbf{q}_k \right)^T \Delta \mathbf{q}_k \quad (1.14a)$$

$$\text{s.t. } \mathbf{A} \Delta \mathbf{q}_k = \mathbf{b} \quad (1.14b)$$

$$\mathbf{C} \Delta \mathbf{q}_k \geq \mathbf{d} \quad (1.14c)$$

### 1.3 コンフィギュレーション更新量の正則項の追加

Gauss-Newton 法と Levenberg-Marquardt 法の比較を参考に，式 (1.14a) の二次形式項の行列に，次式のよう  
に微小な係数をかけた単位行列を加えると，一部の適用例について逐次二次計画法の収束性が改善された<sup>6</sup>．

$$\min_{\Delta \mathbf{q}_k} \frac{1}{2} \Delta \mathbf{q}_k^T \left( \mathbf{J}_k^T \mathbf{J}_k + \bar{\mathbf{W}}_{reg} + \lambda \mathbf{I} \right) \Delta \mathbf{q}_k + \left( \mathbf{J}_k^T \mathbf{e}_k + \bar{\mathbf{W}}_{reg} \mathbf{q}_k \right)^T \Delta \mathbf{q}_k \quad (1.15a)$$

$$\text{s.t. } \mathbf{A} \Delta \mathbf{q}_k = \mathbf{b} \quad (1.15b)$$

$$\mathbf{C} \Delta \mathbf{q}_k \geq \mathbf{d} \quad (1.15c)$$

改良誤差減衰最小二乗法<sup>7</sup>を参考にすると， $\lambda$  は次式のように決定される．

$$\lambda = \lambda_r F(\mathbf{q}_k) + w_r \quad (1.16)$$

$\lambda_r$  と  $w_r$  は正の定数である．

### 1.4 ソースコードと数式の対応

$$\mathbf{W}_{reg} \stackrel{\text{def}}{=} \bar{\mathbf{W}}_{reg} + \lambda \mathbf{I} \quad (1.17a)$$

$$\mathbf{v}_{reg} \stackrel{\text{def}}{=} \bar{\mathbf{W}}_{reg} \mathbf{q}_k \quad (1.17b)$$

とすると，式 (1.15) は次式で表される．

$$\min_{\Delta \mathbf{q}_k} \frac{1}{2} \Delta \mathbf{q}_k^T \left( \mathbf{J}_k^T \mathbf{J}_k + \mathbf{W} \right) \Delta \mathbf{q}_k + \left( \mathbf{J}_k^T \mathbf{e}_k + \mathbf{v}_{reg} \right)^T \Delta \mathbf{q}_k \quad (1.18a)$$

$$\text{s.t. } \mathbf{A} \Delta \mathbf{q}_k = \mathbf{b} \quad (1.18b)$$

$$\mathbf{C} \Delta \mathbf{q}_k \geq \mathbf{d} \quad (1.18c)$$

第 2 節や第 4 章で説明する *\*\*\*-configuration-task* クラスのメソッドは式 (1.18) 中の記号と以下のように対応している．

<i>:config-vector</i>	get $\mathbf{q}$
<i>:set-config</i>	set $\mathbf{q}$
<i>:task-value</i>	get $\mathbf{e}(\mathbf{q})$
<i>:task-jacobian</i>	get $\mathbf{J}(\mathbf{q}) \stackrel{\text{def}}{=} \frac{\partial \mathbf{e}(\mathbf{q})}{\partial \mathbf{q}}$
<i>:config-equality-constraint-matrix</i>	get $\mathbf{A}$
<i>:config-equality-constraint-vector</i>	get $\mathbf{b}$
<i>:config-inequality-constraint-matrix</i>	get $\mathbf{C}$
<i>:config-inequality-constraint-vector</i>	get $\mathbf{d}$
<i>:regular-matrix</i>	get $\mathbf{W}_{reg}$
<i>:regular-vector</i>	get $\mathbf{v}_{reg}$

<sup>6</sup>これは，最適化における信頼領域 (trust region) に関連している．

<sup>7</sup> Levenberg-Marquardt 法による可解性を問わない逆運動学，杉原 知道，日本ロボット学会誌，vol. 29，no. 3，pp. 269-277，2011.

## 1.5 章の構成

第2章では，コンフィギュレーション  $q$  の取得・更新，タスク関数  $e(q)$  の取得，タスク関数のヤコビ行列  $J(q) \stackrel{\text{def}}{=} \frac{\partial e(q)}{\partial q}$  の取得，コンフィギュレーションの等式・不等式制約  $A, b, C, d$  の取得のためのクラスを説明する．第2.1節ではコンフィギュレーション  $q$  が瞬時の情報，第2.2節ではコンフィギュレーション  $q$  が時系列の情報を表す場合をそれぞれ説明する．

第3章では，第2章で説明されるクラスを用いて逐次二次計画法により最適化を行うためのクラスを説明する．

第4章では，用途に応じて拡張されたコンフィギュレーションとタスク関数のクラスを説明する．第4.1節では，マニピュレーションのために，ロボットに加えて物体のコンフィギュレーションを計画する場合を説明する．第4.2節では，ロボットの関節位置の軌道をBスプライン関数でパラメトリックに表現する場合を説明する．いずれにおいても，最適化では第3章で説明された逐次二次計画法のクラスが利用される．

第5章では，その他の補足事項を説明する．第5.1節では，jskeusで定義されているクラスの拡張について説明する．第5.2節では，環境との接触を有するロボットの問題設定を記述するためのクラスについて説明する．第5.4節では，関節トルクを関節角度で微分したヤコビ行列を導出するための関数について説明する．

## 2 コンフィギュレーションとタスク関数

### 2.1 瞬時コンフィギュレーションと瞬時タスク関数

#### instant-configuration-task

[class]

```

:super      propertied-object
:slots      (_robot-env robot-environment instance)
             (_theta-vector  $\theta$  [rad] [m])
             (_wrench-vector  $\hat{w}$  [N] [Nm])
             (_torque-vector  $\tau$  [Nm])
             (_phi-vector  $\phi$  [rad] [m])
             (_num-kin  $N_{kin} := |\mathcal{T}^{kin-trg}| = |\mathcal{T}^{kin-att}|$ )
             (_num-contact  $N_{cnt} := |\mathcal{T}^{cnt-trg}| = |\mathcal{T}^{cnt-att}|$ )
             (_num-variant-joint  $N_{var-joint} := |\mathcal{J}_{var}|$ )
             (_num-invariant-joint  $N_{invar-joint} := |\mathcal{J}_{invar}|$ )
             (_num-drive-joint  $N_{drive-joint} := |\mathcal{J}_{drive}|$ )
             (_num-posture-joint  $N_{posture-joint} := |\mathcal{J}_{posture}|$ )
             (_num-external  $N_{ex} :=$  number of external wrenches)
             (_num-collision  $N_{col} :=$  number of collision check pairs)
             (_dim-theta  $dim(\theta) = N_{var-joint}$ )
             (_dim-wrench  $dim(\hat{w}) = 6N_{cnt}$ )
             (_dim-torque  $dim(\tau) = N_{drive-joint}$ )
             (_dim-phi  $dim(\phi) = N_{invar-joint}$ )
             (_dim-variant-config  $dim(q_{var})$ )
             (_dim-invariant-config  $dim(q_{invar})$ )
             (_dim-config  $dim(q)$ )
             (_dim-task  $dim(e)$ )
             (_kin-scale-mat-list  $K_{kin}$ )
             (_target-posture-scale  $k_{posture}$ )

```

(\_norm-regular-scale-max  $k_{max}$ )  
 (\_norm-regular-scale-offset  $k_{off}$ )  
 (\_torque-regular-scale  $k_{trq}$ )  
 (\_variant-joint-list  $\mathcal{J}_{var}$ )  
 (\_invariant-joint-list  $\mathcal{J}_{invar}$ )  
 (\_drive-joint-list  $\mathcal{J}_{drive}$ )  
 (\_kin-target-coords-list  $\mathcal{T}^{kin-trg}$ )  
 (\_kin-attention-coords-list  $\mathcal{T}^{kin-att}$ )  
 (\_contact-target-coords-list  $\mathcal{T}^{cnt-trg}$ )  
 (\_contact-attention-coords-list  $\mathcal{T}^{cnt-att}$ )  
 (\_variant-joint-angle-margin margin of  $\theta$  [deg] [mm])  
 (\_invariant-joint-angle-margin margin of  $\phi$  [deg] [mm])  
 (\_delta-linear-joint trust region of linear joint configuration [mm])  
 (\_delta-rotational-joint trust region of rotational joint configuration [deg])  
 (\_contact-constraint-list list of contact-constraint instance)  
 (\_posture-joint-list  $\mathcal{J}_{posture}$ )  
 (\_posture-joint-angle-list  $\bar{\theta}^{trg}$ )  
 (\_external-wrench-list  $\{\mathbf{w}_1^{ex}, \mathbf{w}_2^{ex}, \dots, \mathbf{w}_{N_{ex}}^{ex}\}$ )  
 (\_external-coords-list  $\{T_1^{ex}, T_2^{ex}, \dots, T_{N_{ex}}^{ex}\}$ )  
 (\_collision-pair-list list of bodyset-link or body pair)  
 (\_collision-distance-margin-list list of collision distance margin)  
 (\_only-kinematics? whether to consider only kinematics or not)  
 (\_variant-task-jacobi buffer for  $\frac{\partial \mathbf{e}}{\partial \mathbf{q}_{var}}$ )  
 (\_invariant-task-jacobi buffer for  $\frac{\partial \mathbf{e}}{\partial \mathbf{q}_{invar}}$ )  
 (\_task-jacobi buffer for  $\frac{\partial \mathbf{e}}{\partial \mathbf{q}}$ )  
 (\_collision-theta-inequality-constraint-matrix buffer for  $\mathbf{C}_{col, \theta}$ )  
 (\_collision-phi-inequality-constraint-matrix buffer for  $\mathbf{C}_{col, \phi}$ )  
 (\_collision-inequality-constraint-vector buffer for  $\mathbf{C}_{col}$ )

瞬時コンフィギュレーション  $\mathbf{q}^{(l)}$  と瞬時タスク関数  $e^{(l)}(\mathbf{q}^{(l)})$  のクラス .

このクラスの説明で用いる全ての変数は、時間ステップ  $l$  を表す添字をつけて  $x^{(l)}$  と表されるべきだが、このクラス内の説明では省略して  $x$  と表す . また、以降では、説明文やメソッド名で、“瞬時” や “instant” を省略する .

コンフィギュレーション  $\mathbf{q}$  の取得・更新、タスク関数  $e(\mathbf{q})$  の取得、タスク関数のヤコビ行列  $\frac{\partial \mathbf{e}(\mathbf{q})}{\partial \mathbf{q}}$  の取得、コンフィギュレーションの等式・不等式制約  $A, b, C, d$  の取得のためのメソッドが定義されている .

コンフィギュレーション・タスク関数を定めるために、初期化時に以下を与える

- ロボット・環境

robot-environment ロボット・環境を表す robot-environment クラスのインスタンス

variant-joint-list  $\mathcal{J}_{var}$  時変関節

invariant-joint-list  $\mathcal{J}_{invar}$  時不変関節 (与えなければ時不変関節は考慮されない)

drive-joint-list  $\mathcal{J}_{drive}$  駆動関節 (与えなければ関節駆動トルクは考慮されない)

- 幾何拘束

kin-target-coords-list  $\mathcal{T}^{kin-trg}$  幾何到達目標位置姿勢リスト

kin-attention-coords-list  $\mathcal{T}^{kin-att}$  幾何到達着目位置姿勢リスト  
 kin-scale-mat-list  $K_{kin}$  幾何拘束の座標系, 重みを表す変換行列のリスト

- 接触拘束

contact-target-coords-list  $\mathcal{T}^{cnt-trg}$  接触目標位置姿勢リスト  
 contact-attention-coords-list  $\mathcal{T}^{cnt-att}$  接触着目位置姿勢リスト  
 contact-constraint-list 接触レンチ制約リスト

- コンフィギュレーション拘束 (必要な場合のみ)

posture-joint-list  $\mathcal{J}_{posture}$  着目関節リスト  
 posture-joint-angle-list  $\bar{\theta}^{trg}$  着目関節の目標関節角  
 target-posture-scale  $k_{posture}$  コンフィギュレーション拘束の重み

- 干渉回避拘束 (必要な場合のみ)

collision-pair-list 干渉回避する bodyset-link もしくは body のペアのリスト  
 collision-distance-margin 干渉回避の距離マージン (全てのペアで同じ値の場合)  
 collision-distance-margin-list 干渉回避の距離マージンのリスト (ペアごとに異なる値の場合)

- 外レンチ (必要な場合のみ)

external-wrench-list 外レンチのリスト (ワールド座標系で表す)  
 external-coords-list 外レンチの作用点座標のリスト (位置のみを使用)

- 目的関数の重み

norm-regular-scale-max  $k_{max}$  コンフィギュレーション更新量正則化の重み最大値  
 norm-regular-scale-offset  $k_{off}$  コンフィギュレーション更新量正則化の重みオフセット  
 torque-regular-scale  $k_{trq}$  トルク正則化の重み

コンフィギュレーション  $q$  は以下から構成される .

$$q := \begin{pmatrix} \theta^T & \hat{w}^T & \tau^T & \phi^T \end{pmatrix}^T \quad (2.1)$$

$\theta \in \mathbb{R}^{N_{var-joint}}$  時変関節角度 [rad] [m]

$\hat{w} \in \mathbb{R}^{6N_{cnt}}$  接触レンチ [N] [Nm]

$\tau \in \mathbb{R}^{N_{drive-joint}}$  関節駆動トルク [Nm] [N]

$\phi \in \mathbb{R}^{N_{invar-joint}}$  時不変関節角度 [rad] [m]

$\hat{w}$  は次式のように, 全接触部位でのワールド座標系での力・モーメントを並べたベクトルである .

$$\hat{w} = \begin{pmatrix} w_1^T & w_2^T & \cdots & w_{N_{cnt}}^T \end{pmatrix}^T \quad (2.2)$$

$$= \begin{pmatrix} f_1^T & n_1^T & f_2^T & n_2^T & \cdots & f_{N_{cnt}}^T & n_{N_{cnt}}^T \end{pmatrix}^T \quad (2.3)$$

タスク関数  $e(q)$  は以下から構成される .

$$e(q) := \begin{pmatrix} e^{kinT}(q) & e^{eom-transT}(q) & e^{eom-rotT}(q) & e^{trqT}(q) & e^{postureT}(q) \end{pmatrix}^T \quad (2.4)$$

$e^{kin}(q) \in \mathbb{R}^{6N_{kin}}$  幾何到達拘束 [rad] [m]

$e^{eom-trans}(q) \in \mathbb{R}^3$  力の釣り合い [N]

$e^{eom-rot}(q) \in \mathbb{R}^3$  モーメントの釣り合い [Nm]

$e^{trq}(q) \in \mathbb{R}^{N_{drive-joint}}$  関節トルクの釣り合い [rad] [m]



$e^{posture}(\mathbf{q}) \in \mathbb{R}^{N_{posture-joint}}$  関節角目標 [rad] [m]

<b>:init</b>	<i>ℰkey (name)</i> <i>(robot-env)</i> <i>(variant-joint-list (send robot-env :variant-joint-list))</i> <i>(invariant-joint-list (send robot-env :invariant-joint-list))</i> <i>(drive-joint-list (send robot-env :drive-joint-list))</i> <i>(only-kinematics?)</i> <i>(kin-target-coords-list)</i> <i>(kin-attention-coords-list)</i> <i>(contact-target-coords-list)</i> <i>(contact-attention-coords-list)</i> <i>(variant-joint-angle-margin 3.0)</i> <i>(invariant-joint-angle-margin 3.0)</i> <i>(delta-linear-joint)</i> <i>(delta-rotational-joint)</i> <i>(contact-constraint-list (send-all contact-attention-coords-list :get :contact-constraint))</i> <i>(posture-joint-list)</i> <i>(posture-joint-angle-list)</i> <i>(external-wrench-list)</i> <i>(external-coords-list)</i> <i>(collision-pair-list)</i> <i>(collision-distance-margin 0.01)</i> <i>(collision-distance-margin-list)</i> <i>(kin-scale 1.0)</i> <i>(kin-scale-list)</i> <i>(kin-scale-mat-list)</i> <i>(target-posture-scale 0.001)</i> <i>(norm-regular-scale-max (if only-kinematics? 0.001 1.000000e-05))</i> <i>(norm-regular-scale-offset 1.000000e-07)</i> <i>(torque-regular-scale 1.000000e-04)</i> <i>ℰallow-other-keys</i>	[method]
	Initialize instance	
<b>:robot-env</b>	return robot-environment instance	[method]
<b>:variant-joint-list</b>	return $\mathcal{J}_{var}$	[method]
<b>:invariant-joint-list</b>	return $\mathcal{J}_{invar}$	[method]
<b>:drive-joint-list</b>	return $\mathcal{J}_{drive}$	[method]
<b>:only-kinematics?</b>	return whether to consider only kinematics or not	[method]

<b>:theta</b>	[method]
return $\boldsymbol{\theta}$	
<b>:wrench</b>	[method]
return $\hat{\boldsymbol{w}}$	
<b>:torque</b>	[method]
return $\boldsymbol{\tau}$	
<b>:phi</b>	[method]
return $\phi$	
<b>:num-kin</b>	[method]
return $N_{kin} :=  \mathcal{T}^{kin-trg}  =  \mathcal{T}^{kin-att} $	
<b>:num-contact</b>	[method]
return $N_{cnt} :=  \mathcal{T}^{cnt-trg}  =  \mathcal{T}^{cnt-att} $	
<b>:num-variant-joint</b>	[method]
return $N_{var-joint} :=  \mathcal{J}_{var} $	
<b>:num-invariant-joint</b>	[method]
return $N_{invar-joint} :=  \mathcal{J}_{invar} $	
<b>:num-drive-joint</b>	[method]
return $N_{drive-joint} :=  \mathcal{J}_{drive} $	
<b>:num-posture-joint</b>	[method]
return $N_{target-joint} :=  \mathcal{J}_{target} $	
<b>:num-external</b>	[method]
return $N_{ex} :=$ number of external wrench	
<b>:num-collision</b>	[method]
return $N_{col} :=$ number of collision check pairs	
<b>:dim-variant-config</b>	[method]
	$\dim(\mathbf{q}_{var}) \quad := \quad \dim(\boldsymbol{\theta}) + \dim(\hat{\boldsymbol{w}}) + \dim(\boldsymbol{\tau}) \quad (2.5)$
	$= \quad N_{var-joint} + 6N_{cnt} + N_{drive-joint} \quad (2.6)$
return $\dim(\mathbf{q}_{var})$	
<b>:dim-invariant-config</b>	[method]
return $\dim(\mathbf{q}_{invar}) := \dim(\phi) = N_{invar-joint}$	
<b>:dim-config</b>	[method]
return $\dim(\mathbf{q}) := \dim(\mathbf{q}_{var}) + \dim(\mathbf{q}_{invar})$	
<b>:dim-task</b>	[method]

$$\dim(\mathbf{e}) \quad := \quad \dim(\mathbf{e}^{kin}) + \dim(\mathbf{e}^{com-trans}) + \dim(\mathbf{e}^{com-rot}) + \dim(\mathbf{e}^{trq}) + \dim(\mathbf{e}^{posture}) \quad (2.7)$$

$$= \quad 6N_{kin} + 3 + 3 + N_{drive-joint} + N_{posture-joint} \quad (2.8)$$

return  $\dim(\mathbf{e})$

**:variant-config-vector**

[method]

return  $\mathbf{q}_{var} := \begin{pmatrix} \boldsymbol{\theta} \\ \hat{\mathbf{w}} \\ \boldsymbol{\tau} \end{pmatrix}$

**:invariant-config-vector**

[method]

return  $\mathbf{q}_{invar} := \boldsymbol{\phi}$

**:config-vector**

[method]

return  $\mathbf{q} := \begin{pmatrix} \mathbf{q}_{var} \\ \mathbf{q}_{invar} \end{pmatrix} = \begin{pmatrix} \boldsymbol{\theta} \\ \hat{\mathbf{w}} \\ \boldsymbol{\tau} \\ \boldsymbol{\phi} \end{pmatrix}$

**:set-theta** *theta-new* *ℰkey* (*relative?* *nil*)

[method]

(*apply-to-robot?* *t*)

Set  $\boldsymbol{\theta}$ .

**:set-wrench** *wrench-new* *ℰkey* (*relative?* *nil*)

[method]

Set  $\hat{\mathbf{w}}$ .

**:set-torque** *torque-new* *ℰkey* (*relative?* *nil*)

[method]

Set  $\boldsymbol{\tau}$ .

**:set-phi** *phi-new* *ℰkey* (*relative?* *nil*)

[method]

(*apply-to-robot?* *t*)

Set  $\boldsymbol{\phi}$ .

**:set-variant-config** *variant-config-new* *ℰkey* (*relative?* *nil*)

[method]

(*apply-to-robot?* *t*)

Set  $\mathbf{q}_{var}$ .

**:set-invariant-config** *invariant-config-new* *ℰkey* (*relative?* *nil*)

[method]

(*apply-to-robot?* *t*)

Set  $\mathbf{q}_{invar}$ .

**:set-config** *config-new* *ℰkey* (*relative?* *nil*)

[method]

(*apply-to-robot?* *t*)

Set  $\mathbf{q}$ .

**:kin-target-coords-list**

[method]

$$T_m^{kin-trg} = \{\mathbf{p}_m^{kin-trg}, \mathbf{R}_m^{kin-trg}\} \quad (m = 1, 2, \dots, N_{kin}) \quad (2.9)$$

return  $\mathcal{T}^{kin-trg} := \{T_1^{kin-trg}, T_2^{kin-trg}, \dots, T_{N_{kin}}^{kin-trg}\}$

**:kin-attention-coords-list**

[method]

$$T_m^{kin-att} = \{\mathbf{p}_m^{kin-att}, \mathbf{R}_m^{kin-att}\} \quad (m = 1, 2, \dots, N_{kin}) \quad (2.10)$$

return  $\mathcal{T}^{kin-att} := \{T_1^{kin-att}, T_2^{kin-att}, \dots, T_{N_{kin}}^{kin-att}\}$

**:contact-target-coords-list**

[method]

$$T_m^{cnt-trg} = \{\mathbf{p}_m^{cnt-trg}, \mathbf{R}_m^{cnt-trg}\} \quad (m = 1, 2, \dots, N_{cnt}) \quad (2.11)$$

return  $\mathcal{T}^{cnt-trg} := \{T_1^{cnt-trg}, T_2^{cnt-trg}, \dots, T_{N_{cnt}}^{cnt-trg}\}$

**:contact-attention-coords-list**

[method]

$$T_m^{cnt-att} = \{\mathbf{p}_m^{cnt-att}, \mathbf{R}_m^{cnt-att}\} \quad (m = 1, 2, \dots, N_{cnt}) \quad (2.12)$$

return  $\mathcal{T}^{cnt-att} := \{T_1^{cnt-att}, T_2^{cnt-att}, \dots, T_{N_{cnt}}^{cnt-att}\}$

**:contact-constraint-list**

[method]

return list of contact-constraint instance

**:wrench-list**

[method]

return  $\{\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_{N_{cnt}}\}$

**:force-list**

[method]

return  $\{\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_{N_{cnt}}\}$

**:moment-list**

[method]

return  $\{\mathbf{n}_1, \mathbf{n}_2, \dots, \mathbf{n}_{N_{cnt}}\}$

**:external-wrench-list**

[method]

return  $\{\mathbf{w}_1^{ex}, \mathbf{w}_2^{ex}, \dots, \mathbf{w}_{N_{ex}}^{ex}\}$

**:external-force-list**

[method]

return  $\{\mathbf{f}_1^{ex}, \mathbf{f}_2^{ex}, \dots, \mathbf{f}_{N_{ex}}^{ex}\}$

**:external-moment-list**

[method]

return  $\{\mathbf{n}_1^{ex}, \mathbf{n}_2^{ex}, \dots, \mathbf{n}_{N_{ex}}^{ex}\}$

**:mg-vec**

[method]

return  $m\mathbf{g}$

**:cog**  $\mathcal{E}key$  (update?  $t$ )

[method]

return  $\mathbf{p}_G(\mathbf{q})$

**:kinematics-task-value**  $\mathcal{E}key$  (update?  $t$ )

[method]

$$\mathbf{e}^{kin}(\mathbf{q}) = \mathbf{e}^{kin}(\boldsymbol{\theta}, \phi) \quad (2.13)$$

$$= \begin{pmatrix} \mathbf{e}_1^{kin}(\boldsymbol{\theta}, \phi) \\ \mathbf{e}_2^{kin}(\boldsymbol{\theta}, \phi) \\ \vdots \\ \mathbf{e}_{N_{kin}}^{kin}(\boldsymbol{\theta}, \phi) \end{pmatrix} \quad (2.14)$$

$$\mathbf{e}_m^{kin}(\boldsymbol{\theta}, \phi) = K_{kin} \begin{pmatrix} \mathbf{p}_m^{kin-trg}(\boldsymbol{\theta}, \phi) - \mathbf{p}_m^{kin-att}(\boldsymbol{\theta}, \phi) \\ \mathbf{a} \left( \mathbf{R}_m^{kin-trg}(\boldsymbol{\theta}, \phi) \mathbf{R}_m^{kin-att}(\boldsymbol{\theta}, \phi)^T \right) \end{pmatrix} \in \mathbb{R}^6 \quad (m = 1, 2, \dots, N_{kin}) \quad (2.15)$$

$a(R)$  は姿勢行列  $R$  の等価角軸ベクトルを表す .

return  $e^{kin}(q) \in \mathbb{R}^{6N_{kin}}$

**:eom-trans-task-value**  $\mathcal{E}key$  (update?  $t$ )

[method]

$$e^{eom-trans}(q) = e^{eom-trans}(\hat{w}) \quad (2.16)$$

$$= \sum_{m=1}^{N_{cnt}} f_m - mg + \sum_{m=1}^{N_{ex}} f_m^{ex} \quad (2.17)$$

return  $e^{eom-trans}(q) \in \mathbb{R}^3$

**:eom-rot-task-value**  $\mathcal{E}key$  (update?  $t$ )

[method]

$$e^{eom-rot}(q) = e^{eom-rot}(\theta, \hat{w}, \phi) \quad (2.18)$$

$$\begin{aligned} &= \sum_{m=1}^{N_{cnt}} \{ (p_m^{cnt-trg}(\theta, \phi) - p_G(\theta, \phi)) \times f_m + n_m \} \\ &+ \sum_{m=1}^{N_{ex}} \{ (p_m^{ex}(\theta, \phi) - p_G(\theta, \phi)) \times f_m^{ex} + n_m^{ex} \} \end{aligned} \quad (2.19)$$

return  $e^{eom-rot}(q) \in \mathbb{R}^3$

**:torque-task-value**  $\mathcal{E}key$  (update?  $t$ )

[method]

$$e^{trq}(q) = e^{trq}(\theta, \hat{w}, \tau, \phi) \quad (2.20)$$

$$= \tau + \sum_{m=1}^{N_{cnt}} \tau_m^{cnt}(\theta, \phi) - \tau^{grav}(\theta, \phi) + \sum_{m=1}^{N_{ex}} \tau_m^{ex}(\theta, \phi) \quad (2.21)$$

$$= \tau + \sum_{m=1}^{N_{cnt}} J_{drive-joint,m}^{cnt-att}(\theta, \phi)^T w_m - \tau^{grav}(\theta, \phi) + \sum_{m=1}^{N_{ex}} J_{drive-joint,m}^{ex}(\theta, \phi)^T w_m^{ex} \quad (2.22)$$

$\tau_m^{cnt}(\theta, \phi)$  は  $m$  番目の接触部位にかかる接触レンチ  $w_m$  による関節トルク ,  $\tau_m^{grav}(\theta, \phi)$  は自重による関節トルクを表す .

return  $e^{trq}(q) \in \mathbb{R}^{N_{drive-joint}}$

**:posture-task-value**  $\mathcal{E}key$  (update?  $t$ )

[method]

$$e^{posture}(q) = e^{posture}(\theta) \quad (2.23)$$

$$= k_{posture}(\bar{\theta}^{trg} - \bar{\theta}) \quad (2.24)$$

$\bar{\theta}^{trg}, \bar{\theta}$  は着目関節リスト  $\mathcal{J}_{posture}$  の目標関節角と現在の関節角 .

return  $e^{posture}(q) \in \mathbb{R}^{N_{posture-joint}}$

**:task-value**  $\mathcal{E}key$  (update?  $t$ )

[method]

$$\text{return } e(q) := \begin{pmatrix} e^{kin}(q) \\ e^{eom-trans}(q) \\ e^{eom-rot}(q) \\ e^{trq}(q) \\ e^{posture}(q) \end{pmatrix} = \begin{pmatrix} e^{kin}(\theta, \phi) \\ e^{eom-trans}(\hat{w}) \\ e^{eom-rot}(\theta, \hat{w}, \phi) \\ e^{trq}(\theta, \hat{w}, \tau, \phi) \\ e^{posture}(\theta) \end{pmatrix}$$

:kinematics-task-jacobian-with-theta

[method]

$$\frac{\partial \mathbf{e}^{kin}}{\partial \boldsymbol{\theta}} = \begin{pmatrix} \frac{\partial \mathbf{e}_1^{kin}}{\partial \boldsymbol{\theta}} \\ \frac{\partial \mathbf{e}_2^{kin}}{\partial \boldsymbol{\theta}} \\ \vdots \\ \frac{\partial \mathbf{e}_{N_{kin}}^{kin}}{\partial \boldsymbol{\theta}} \end{pmatrix} \quad (2.25)$$

$$\frac{\partial \mathbf{e}_m^{kin}}{\partial \boldsymbol{\theta}} = K_{kin} \left\{ \mathbf{J}_{\theta,m}^{kin-trg}(\boldsymbol{\theta}, \phi) - \mathbf{J}_{\theta,m}^{kin-att}(\boldsymbol{\theta}, \phi) \right\} \quad (m = 1, 2, \dots, N_{kin}) \quad (2.26)$$

$$\text{return } \frac{\partial \mathbf{e}^{kin}}{\partial \boldsymbol{\theta}} \in \mathbb{R}^{6N_{kin} \times N_{var-joint}}$$

:kinematics-task-jacobian-with-phi

[method]

$$\frac{\partial \mathbf{e}^{kin}}{\partial \boldsymbol{\phi}} = \begin{pmatrix} \frac{\partial \mathbf{e}_1^{kin}}{\partial \boldsymbol{\phi}} \\ \frac{\partial \mathbf{e}_2^{kin}}{\partial \boldsymbol{\phi}} \\ \vdots \\ \frac{\partial \mathbf{e}_{N_{kin}}^{kin}}{\partial \boldsymbol{\phi}} \end{pmatrix} \quad (2.27)$$

$$\frac{\partial \mathbf{e}_m^{kin}}{\partial \boldsymbol{\phi}} = K_{kin} \left\{ \mathbf{J}_{\phi,m}^{kin-trg}(\boldsymbol{\theta}, \phi) - \mathbf{J}_{\phi,m}^{kin-att}(\boldsymbol{\theta}, \phi) \right\} \quad (m = 1, 2, \dots, N_{kin}) \quad (2.28)$$

$$\text{return } \frac{\partial \mathbf{e}^{kin}}{\partial \boldsymbol{\phi}} \in \mathbb{R}^{6N_{kin} \times N_{invar-joint}}$$

:eom-trans-task-jacobian-with-wrench

[method]

$$\frac{\partial \mathbf{e}^{eom-trans}}{\partial \hat{\mathbf{w}}} = \begin{pmatrix} \frac{\partial \mathbf{e}^{eom-trans}}{\partial \mathbf{f}_1} & \frac{\partial \mathbf{e}^{eom-trans}}{\partial \mathbf{n}_1} & \dots & \frac{\partial \mathbf{e}^{eom-trans}}{\partial \mathbf{f}_{N_{cnt}}} & \frac{\partial \mathbf{e}^{eom-trans}}{\partial \mathbf{n}_{N_{cnt}}} \end{pmatrix} \quad (2.29)$$

$$= \begin{pmatrix} \mathbf{I}_3 & \mathbf{O}_3 & \dots & \mathbf{I}_3 & \mathbf{O}_3 \end{pmatrix} \quad (2.30)$$

$$\text{return } \frac{\partial \mathbf{e}^{eom-trans}}{\partial \hat{\mathbf{w}}} \in \mathbb{R}^{3 \times 6N_{cnt}}$$

:eom-rot-task-jacobian-with-theta

[method]

$$\begin{aligned} \frac{\partial \mathbf{e}^{eom-rot}}{\partial \boldsymbol{\theta}} &= \sum_{m=1}^{N_{cnt}} \left\{ -[\mathbf{f}_m \times] \left( \mathbf{J}_{\theta,m}^{cnt-trg}(\boldsymbol{\theta}, \phi) - \mathbf{J}_{G\theta}(\boldsymbol{\theta}, \phi) \right) \right\} \\ &\quad + \sum_{m=1}^{N_{ex}} \left\{ -[\mathbf{f}_m^{ex} \times] \left( \mathbf{J}_{\theta,m}^{ex}(\boldsymbol{\theta}, \phi) - \mathbf{J}_{G\theta}(\boldsymbol{\theta}, \phi) \right) \right\} \end{aligned} \quad (2.31)$$

$$\begin{aligned} &= \left[ \left( \sum_{m=1}^{N_{cnt}} \mathbf{f}_m + \sum_{m=1}^{N_{ex}} \mathbf{f}_m^{ex} \right) \times \right] \mathbf{J}_{G\theta}(\boldsymbol{\theta}, \phi) \\ &\quad - \sum_{m=1}^{N_{cnt}} [\mathbf{f}_m \times] \mathbf{J}_{\theta,m}^{cnt-trg}(\boldsymbol{\theta}, \phi) - \sum_{m=1}^{N_{ex}} [\mathbf{f}_m^{ex} \times] \mathbf{J}_{\theta,m}^{ex}(\boldsymbol{\theta}, \phi) \end{aligned} \quad (2.32)$$

$\sum_{m=1}^{N_{cnt}} \mathbf{f}_m + \sum_{m=1}^{N_{ex}} \mathbf{f}_m^{ex} = m\mathbf{g}$  つまり, eom-trans-task が成立すると仮定すると次式が成り立つ.

$$\frac{\partial \mathbf{e}^{eom-rot}}{\partial \boldsymbol{\theta}} = [m\mathbf{g} \times] \mathbf{J}_{G\theta}(\boldsymbol{\theta}, \phi) - \sum_{m=1}^{N_{cnt}} [\mathbf{f}_m \times] \mathbf{J}_{\theta,m}^{cnt-trg}(\boldsymbol{\theta}, \phi) - \sum_{m=1}^{N_{ex}} [\mathbf{f}_m^{ex} \times] \mathbf{J}_{\theta,m}^{ex}(\boldsymbol{\theta}, \phi) \quad (2.33)$$

return  $\frac{\partial \mathbf{e}^{eom-rot}}{\partial \boldsymbol{\theta}} \in \mathbb{R}^{3 \times N_{var-joint}}$

**:eom-rot-task-jacobian-with-wrench**

[method]

$$\frac{\partial \mathbf{e}^{eom-rot}}{\partial \hat{\mathbf{w}}} = \begin{pmatrix} \frac{\partial \mathbf{e}^{eom-rot}}{\partial \mathbf{f}_1} & \frac{\partial \mathbf{e}^{eom-rot}}{\partial \mathbf{n}_1} & \cdots & \frac{\partial \mathbf{e}^{eom-rot}}{\partial \mathbf{f}_{N_{cnt}}} & \frac{\partial \mathbf{e}^{eom-rot}}{\partial \mathbf{n}_{N_{cnt}}} \end{pmatrix} \quad (2.34)$$

$$\frac{\partial \mathbf{e}^{eom-rot}}{\partial \mathbf{f}_m} = [(\mathbf{p}_m^{cnt-trg}(\boldsymbol{\theta}, \phi) - \mathbf{p}_G(\boldsymbol{\theta}, \phi)) \times] \quad (m = 1, 2, \dots, N_{cnt}) \quad (2.35)$$

$$\frac{\partial \mathbf{e}^{eom-rot}}{\partial \mathbf{n}_m} = \mathbf{I}_3 \quad (m = 1, 2, \dots, N_{cnt}) \quad (2.36)$$

return  $\frac{\partial \mathbf{e}^{eom-rot}}{\partial \hat{\mathbf{w}}} \in \mathbb{R}^{3 \times 6N_{cnt}}$

**:eom-rot-task-jacobian-with-phi**

[method]

$$\begin{aligned} \frac{\partial \mathbf{e}^{eom-rot}}{\partial \phi} &= \sum_{m=1}^{N_{cnt}} \left\{ -[\mathbf{f}_m \times] \left( \mathbf{J}_{\phi, m}^{cnt-trg}(\boldsymbol{\theta}, \phi) - \mathbf{J}_{G\phi}(\boldsymbol{\theta}, \phi) \right) \right\} \\ &\quad + \sum_{m=1}^{N_{ex}} \left\{ -[\mathbf{f}_m^{ex} \times] \left( \mathbf{J}_{\phi, m}^{ex}(\boldsymbol{\theta}, \phi) - \mathbf{J}_{G\phi}(\boldsymbol{\theta}, \phi) \right) \right\} \end{aligned} \quad (2.37)$$

$$\begin{aligned} &= \left[ \left( \sum_{m=1}^{N_{cnt}} \mathbf{f}_m + \sum_{m=1}^{N_{ex}} \mathbf{f}_m^{ex} \right) \times \right] \mathbf{J}_{G\phi}(\boldsymbol{\theta}, \phi) \\ &\quad - \sum_{m=1}^{N_{cnt}} [\mathbf{f}_m \times] \mathbf{J}_{\phi, m}^{cnt-trg}(\boldsymbol{\theta}, \phi) - \sum_{m=1}^{N_{ex}} [\mathbf{f}_m^{ex} \times] \mathbf{J}_{\phi, m}^{ex}(\boldsymbol{\theta}, \phi) \end{aligned} \quad (2.38)$$

$\sum_{m=1}^{N_{cnt}} \mathbf{f}_m + \sum_{m=1}^{N_{ex}} \mathbf{f}_m^{ex} = m\mathbf{g}$  つまり, eom-trans-task が成立すると仮定すると次式が成り立つ.

$$\frac{\partial \mathbf{e}^{eom-rot}}{\partial \phi} = [m\mathbf{g} \times] \mathbf{J}_{G\phi}(\boldsymbol{\theta}, \phi) - \sum_{m=1}^{N_{cnt}} [\mathbf{f}_m \times] \mathbf{J}_{\phi, m}^{cnt-trg}(\boldsymbol{\theta}, \phi) - \sum_{m=1}^{N_{ex}} [\mathbf{f}_m^{ex} \times] \mathbf{J}_{\phi, m}^{ex}(\boldsymbol{\theta}, \phi) \quad (2.39)$$

return  $\frac{\partial \mathbf{e}^{eom-rot}}{\partial \phi} \in \mathbb{R}^{3 \times N_{invar-joint}}$

**:torque-task-jacobian-with-theta**

[method]

$$\frac{\partial \mathbf{e}^{trq}}{\partial \boldsymbol{\theta}} = \sum_{m=1}^{N_{cnt}} \frac{\partial \boldsymbol{\tau}_m^{cnt}}{\partial \boldsymbol{\theta}} - \frac{\partial \boldsymbol{\tau}^{grav}}{\partial \boldsymbol{\theta}} + \sum_{m=1}^{N_{ex}} \frac{\partial \boldsymbol{\tau}_m^{ex}}{\partial \boldsymbol{\theta}} \quad (2.40)$$

return  $\frac{\partial \mathbf{e}^{trq}}{\partial \boldsymbol{\theta}} \in \mathbb{R}^{N_{drive-joint} \times N_{var-joint}}$

**:torque-task-jacobian-with-wrench**

[method]

$$\frac{\partial \mathbf{e}^{trq}}{\partial \hat{\mathbf{w}}} = \begin{pmatrix} \frac{\partial \mathbf{e}^{trq}}{\partial \mathbf{w}_1} & \frac{\partial \mathbf{e}^{trq}}{\partial \mathbf{w}_2} & \cdots & \frac{\partial \mathbf{e}^{trq}}{\partial \mathbf{w}_{N_{cnt}}} \end{pmatrix} \quad (2.41)$$

$$\frac{\partial \mathbf{e}^{trq}}{\partial \mathbf{w}_m} = \mathbf{J}_{drive-joint, m}^{cnt-att}(\boldsymbol{\theta}, \phi)^T \quad (m = 1, 2, \dots, N_{cnt}) \quad (2.42)$$

return  $\frac{\partial \mathbf{e}^{trq}}{\partial \hat{\mathbf{w}}} \in \mathbb{R}^{N_{drive-joint} \times 6N_{cnt}}$

**:torque-task-jacobian-with-phi**

[method]

$$\frac{\partial \mathbf{e}^{trq}}{\partial \phi} = \sum_{m=1}^{N_{cnt}} \frac{\partial \tau_m^{cnt}}{\partial \phi} - \frac{\partial \tau_m^{grav}}{\partial \phi} + \sum_{m=1}^{N_{ex}} \frac{\partial \tau_m^{ex}}{\partial \phi} \quad (2.43)$$

$$\text{return } \frac{\partial \mathbf{e}^{trq}}{\partial \phi} \in \mathbb{R}^{N_{drive-joint} \times N_{invar-joint}}$$

**:torque-task-jacobian-with-torque**

[method]

$$\frac{\partial \mathbf{e}^{trq}}{\partial \tau} = \mathbf{I}_{N_{drive-joint}} \quad (2.44)$$

$$\text{return } \frac{\partial \mathbf{e}^{trq}}{\partial \tau} \in \mathbb{R}^{N_{drive-joint} \times N_{drive-joint}}$$

**:posture-task-jacobian-with-theta** *key (update? nil)*

[method]

$$\left( \frac{\partial \mathbf{e}^{posture}}{\partial \theta} \right)_{i,j} = \begin{cases} -k_{posture} & (\mathcal{J}_{posture,i} = \mathcal{J}_{var,j}) \\ 0 & \text{otherwise} \end{cases} \quad (2.45)$$

$$\text{return } \frac{\partial \mathbf{e}^{posture}}{\partial \theta} \in \mathbb{R}^{N_{posture-joint} \times N_{var-joint}}$$

**:variant-task-jacobian**

[method]

$$\frac{\partial \mathbf{e}}{\partial \mathbf{q}_{var}} = \begin{matrix} 6N_{kin} \\ 3 \\ 3 \\ N_{drive-joint} \\ N_{posture-joint} \end{matrix} \begin{pmatrix} N_{var-joint} & 6N_{cnt} & N_{drive-joint} \\ \frac{\partial \mathbf{e}^{kin}}{\partial \theta} & \frac{\partial \mathbf{e}^{com-trans}}{\partial \hat{\mathbf{w}}} & \\ \frac{\partial \mathbf{e}^{com-rot}}{\partial \theta} & \frac{\partial \mathbf{e}^{com-rot}}{\partial \hat{\mathbf{w}}} & \\ \frac{\partial \mathbf{e}^{trq}}{\partial \theta} & \frac{\partial \mathbf{e}^{trq}}{\partial \hat{\mathbf{w}}} & \frac{\partial \mathbf{e}^{trq}}{\partial \tau} \\ \frac{\partial \mathbf{e}^{posture}}{\partial \theta} & & \end{pmatrix} \quad (2.46)$$

$$\text{return } \frac{\partial \mathbf{e}}{\partial \mathbf{q}_{var}} \in \mathbb{R}^{(6N_{kin}+3+3+N_{drive-joint}+N_{posture-joint}) \times (N_{var-joint}+6N_{cnt}+N_{drive-joint})}$$

**:invariant-task-jacobian**

[method]

$$\frac{\partial \mathbf{e}}{\partial \mathbf{q}_{invar}} = \begin{matrix} 6N_{kin} \\ 3 \\ 3 \\ N_{drive-joint} \\ N_{posture-joint} \end{matrix} \begin{pmatrix} N_{invar-joint} \\ \frac{\partial \mathbf{e}^{kin}}{\partial \phi} \\ \frac{\partial \mathbf{e}^{com-rot}}{\partial \phi} \\ \frac{\partial \mathbf{e}^{trq}}{\partial \phi} \\ \end{pmatrix} \quad (2.47)$$

$$\text{return } \frac{\partial \mathbf{e}}{\partial \mathbf{q}_{invar}} \in \mathbb{R}^{(6N_{kin}+3+3+N_{drive-joint}+N_{posture-joint}) \times N_{invar-joint}}$$

**:task-jacobian**

[method]



$$\frac{\partial \mathbf{e}}{\partial \mathbf{q}} = \begin{pmatrix} \frac{\partial \mathbf{e}}{\partial \mathbf{q}_{var}} & \frac{\partial \mathbf{e}}{\partial \mathbf{q}_{invar}} \end{pmatrix} \quad (2.48)$$

$$= \begin{matrix} & N_{var-joint} & 6N_{cnt} & N_{drive-joint} & N_{invar-joint} \\ 6N_{kin} & \left( \frac{\partial \mathbf{e}^{kin}}{\partial \boldsymbol{\theta}} & & & \frac{\partial \mathbf{e}^{kin}}{\partial \boldsymbol{\phi}} \right) \\ 3 & & \frac{\partial \mathbf{e}^{com-trans}}{\partial \hat{\mathbf{w}}} & & \\ 3 & \frac{\partial \mathbf{e}^{com-rot}}{\partial \boldsymbol{\theta}} & \frac{\partial \mathbf{e}^{com-rot}}{\partial \hat{\mathbf{w}}} & & \frac{\partial \mathbf{e}^{com-rot}}{\partial \boldsymbol{\phi}} \\ N_{drive-joint} & \frac{\partial \mathbf{e}^{trq}}{\partial \boldsymbol{\theta}} & \frac{\partial \mathbf{e}^{trq}}{\partial \hat{\mathbf{w}}} & \frac{\partial \mathbf{e}^{trq}}{\partial \boldsymbol{\tau}} & \frac{\partial \mathbf{e}^{trq}}{\partial \boldsymbol{\phi}} \\ N_{posture-joint} & \frac{\partial \mathbf{e}^{posture}}{\partial \boldsymbol{\theta}} & & & \end{matrix} \quad (2.49)$$

return  $\frac{\partial \mathbf{e}}{\partial \mathbf{q}} \in \mathbb{R}^{(6N_{kin}+3+3+N_{drive-joint}+N_{posture-joint}) \times (N_{var-joint}+6N_{cnt}+N_{drive-joint}+N_{invar-joint})}$

**:theta-max-vector**  $\mathcal{E}key$  (update? nil) [method]

return  $\boldsymbol{\theta}_{max} \in \mathbb{R}^{N_{var-joint}}$

**:theta-min-vector**  $\mathcal{E}key$  (update? nil) [method]

return  $\boldsymbol{\theta}_{min} \in \mathbb{R}^{N_{var-joint}}$

**:delta-theta-limit-vector**  $\mathcal{E}key$  (update? nil) [method]

get trust region of  $\boldsymbol{\theta}$

return  $\Delta \boldsymbol{\theta}_{limit}$

**:theta-inequality-constraint-matrix**  $\mathcal{E}key$  (update? nil) [method]

$$\begin{cases} \boldsymbol{\theta}_{min} \leq \boldsymbol{\theta} + \Delta \boldsymbol{\theta} \leq \boldsymbol{\theta}_{max} \\ -\Delta \boldsymbol{\theta}_{limit} \leq \Delta \boldsymbol{\theta} \leq \Delta \boldsymbol{\theta}_{limit} \end{cases} \quad (\text{if } \Delta \boldsymbol{\theta}_{limit} \text{ is set}) \quad (2.50)$$

$$\Leftrightarrow \begin{pmatrix} \mathbf{I} \\ -\mathbf{I} \\ \mathbf{I} \\ -\mathbf{I} \end{pmatrix} \Delta \boldsymbol{\theta} \geq \begin{pmatrix} \boldsymbol{\theta}_{min} - \boldsymbol{\theta} \\ -(\boldsymbol{\theta}_{max} - \boldsymbol{\theta}) \\ -\Delta \boldsymbol{\theta}_{limit} \\ -\Delta \boldsymbol{\theta}_{limit} \end{pmatrix} \quad (2.51)$$

$$\Leftrightarrow \mathbf{C}_{\boldsymbol{\theta}} \Delta \boldsymbol{\theta} \geq \mathbf{d}_{\boldsymbol{\theta}} \quad (2.52)$$

$$\text{return } \mathbf{C}_{\boldsymbol{\theta}} := \begin{pmatrix} \mathbf{I} \\ -\mathbf{I} \\ \mathbf{I} \\ -\mathbf{I} \end{pmatrix} \in \mathbb{R}^{4N_{var-joint} \times N_{var-joint}}$$

**:theta-inequality-constraint-vector**  $\mathcal{E}key$  (update? t) [method]

$$\text{return } \mathbf{d}_{\boldsymbol{\theta}} := \begin{pmatrix} \boldsymbol{\theta}_{min} - \boldsymbol{\theta} \\ -(\boldsymbol{\theta}_{max} - \boldsymbol{\theta}) \\ -\Delta \boldsymbol{\theta}_{limit} \\ -\Delta \boldsymbol{\theta}_{limit} \end{pmatrix} \in \mathbb{R}^{4N_{var-joint}}$$

**:wrench-inequality-constraint-matrix**  $\mathcal{E}key$  (update? t) [method]

接触レンチ  $\mathbf{w} \in \mathbb{R}^6$  が満たすべき制約（非負制約，摩擦制約，圧力中心制約）が次式のように表されるとする．

$$\mathbf{C}_w \mathbf{w} \geq \mathbf{d}_w \quad (2.53)$$

$N_{cnt}$  箇所の接触部位の接触レンチを並べたベクトル  $\hat{w}$  の不等式制約は次式で表される .

$$C_{w,m}(\mathbf{w}_m + \Delta \mathbf{w}_m) \geq \mathbf{d}_{w,m} \quad (m = 1, 2, \dots, N_{cnt}) \quad (2.54)$$

$$\Leftrightarrow C_{w,m} \Delta \mathbf{w}_m \geq \mathbf{d}_{w,m} - C_{w,m} \mathbf{w}_m \quad (m = 1, 2, \dots, N_{cnt}) \quad (2.55)$$

$$\Leftrightarrow \begin{pmatrix} C_{w,1} & & & \\ & C_{w,2} & & \\ & & \ddots & \\ & & & C_{w,N_{cnt}} \end{pmatrix} \begin{pmatrix} \Delta \mathbf{w}_1 \\ \Delta \mathbf{w}_2 \\ \vdots \\ \Delta \mathbf{w}_{N_{cnt}} \end{pmatrix} \geq \begin{pmatrix} \mathbf{d}_{w,1} - C_{w,1} \mathbf{w}_1 \\ \mathbf{d}_{w,2} - C_{w,2} \mathbf{w}_2 \\ \vdots \\ \mathbf{d}_{w,N_{cnt}} - C_{w,N_{cnt}} \mathbf{w}_{N_{cnt}} \end{pmatrix} \quad (2.56)$$

$$\Leftrightarrow C_{\hat{w}} \Delta \hat{\mathbf{w}} \geq \mathbf{d}_{\hat{w}} \quad (2.57)$$

$$\text{return } C_{\hat{w}} := \begin{pmatrix} C_{w,1} & & & \\ & C_{w,2} & & \\ & & \ddots & \\ & & & C_{w,N_{cnt}} \end{pmatrix} \in \mathbb{R}^{N_{wrench-ineq} \times \dim(\hat{\mathbf{w}})}$$

**:wrench-inequality-constraint-vector**  $\mathcal{E}key \text{ (update? } t)$  [method]

$$\text{return } \mathbf{d}_{\hat{w}} := \begin{pmatrix} \mathbf{d}_{w,1} - C_{w,1} \mathbf{w}_1 \\ \mathbf{d}_{w,2} - C_{w,2} \mathbf{w}_2 \\ \vdots \\ \mathbf{d}_{w,N_{cnt}} - C_{w,N_{cnt}} \mathbf{w}_{N_{cnt}} \end{pmatrix} \in \mathbb{R}^{N_{wrench-ineq}}$$

**:torque-max-vector**  $\mathcal{E}key \text{ (update? nil)}$  [method]

$$\text{return } \boldsymbol{\tau}_{max} \in \mathbb{R}^{N_{drive-joint}}$$

**:torque-min-vector**  $\mathcal{E}key \text{ (update? nil)}$  [method]

$$\text{return } \boldsymbol{\tau}_{min} \in \mathbb{R}^{N_{drive-joint}}$$

**:torque-inequality-constraint-matrix**  $\mathcal{E}key \text{ (update? nil)}$  [method]

$$\boldsymbol{\tau}_{min} \leq \boldsymbol{\tau} + \Delta \boldsymbol{\tau} \leq \boldsymbol{\tau}_{max} \quad (2.58)$$

$$\Leftrightarrow \begin{pmatrix} \mathbf{I} \\ -\mathbf{I} \end{pmatrix} \Delta \boldsymbol{\tau} \geq \begin{pmatrix} \boldsymbol{\tau}_{min} - \boldsymbol{\tau} \\ -(\boldsymbol{\tau}_{max} - \boldsymbol{\tau}) \end{pmatrix} \quad (2.59)$$

$$\Leftrightarrow C_{\boldsymbol{\tau}} \Delta \boldsymbol{\tau} \geq \mathbf{d}_{\boldsymbol{\tau}} \quad (2.60)$$

$$\text{return } C_{\boldsymbol{\tau}} := \begin{pmatrix} \mathbf{I} \\ -\mathbf{I} \end{pmatrix} \in \mathbb{R}^{2N_{drive-joint} \times N_{drive-joint}}$$

**:torque-inequality-constraint-vector**  $\mathcal{E}key \text{ (update? } t)$  [method]

$$\text{return } \mathbf{d}_{\boldsymbol{\tau}} := \begin{pmatrix} \boldsymbol{\tau}_{min} - \boldsymbol{\tau} \\ -(\boldsymbol{\tau}_{max} - \boldsymbol{\tau}) \end{pmatrix} \in \mathbb{R}^{2N_{drive-joint}}$$

**:phi-max-vector**  $\mathcal{E}key \text{ (update? nil)}$  [method]

$$\text{return } \boldsymbol{\phi}_{max} \in \mathbb{R}^{N_{invar-joint}}$$

**:phi-min-vector**  $\mathcal{E}key \text{ (update? nil)}$  [method]

$$\text{return } \boldsymbol{\phi}_{min} \in \mathbb{R}^{N_{invar-joint}}$$

**:delta-phi-limit-vector**  $\mathcal{E}key \text{ (update? nil)}$  [method]

get trust region of  $\boldsymbol{\phi}$

$$\text{return } \Delta \boldsymbol{\phi}_{limit}$$

**:phi-inequality-constraint-matrix**  $\mathcal{E}key$  (*update? nil*)

[method]

$$\begin{cases} \phi_{min} \leq \phi + \Delta\phi \leq \phi_{max} \\ -\Delta\phi_{limit} \leq \Delta\phi \leq \Delta\phi_{limit} \quad (\text{if } \Delta\phi_{limit} \text{ is set}) \end{cases} \quad (2.61)$$

$$\Leftrightarrow \begin{pmatrix} I \\ -I \\ I \\ -I \end{pmatrix} \Delta\phi \geq \begin{pmatrix} \phi_{min} - \phi \\ -(\phi_{max} - \phi) \\ -\Delta\phi_{limit} \\ -\Delta\phi_{limit} \end{pmatrix} \quad (2.62)$$

$$\Leftrightarrow C_\phi \Delta\phi \geq d_\phi \quad (2.63)$$

$$\text{return } C_\phi := \begin{pmatrix} I \\ -I \\ I \\ -I \end{pmatrix} \in \mathbb{R}^{4N_{invar-joint} \times N_{invar-joint}}$$

**:phi-inequality-constraint-vector**  $\mathcal{E}key$  (*update? t*)

[method]

$$\text{return } d_\phi := \begin{pmatrix} \phi_{min} - \phi \\ -(\phi_{max} - \phi) \\ -\Delta\phi_{limit} \\ -\Delta\phi_{limit} \end{pmatrix} \in \mathbb{R}^{4N_{invar-joint}}$$

**:variant-config-inequality-constraint-matrix**  $\mathcal{E}key$  (*update? nil*)

[method]

$$\begin{cases} C_\theta \Delta\theta \geq d_\theta \\ C_{\hat{w}} \Delta\hat{w} \geq d_{\hat{w}} \\ C_\tau \Delta\tau \geq d_\tau \end{cases} \quad (2.64)$$

$$\Leftrightarrow \begin{pmatrix} C_\theta & & \\ & C_{\hat{w}} & \\ & & C_\tau \end{pmatrix} \begin{pmatrix} \Delta\theta \\ \Delta\hat{w} \\ \Delta\tau \end{pmatrix} \geq \begin{pmatrix} d_\theta \\ d_{\hat{w}} \\ d_\tau \end{pmatrix} \quad (2.65)$$

$$\Leftrightarrow C_{var} \Delta q_{var} \geq d_{var} \quad (2.66)$$

$$\text{return } C_{var} := \begin{pmatrix} C_\theta & & \\ & C_{\hat{w}} & \\ & & C_\tau \end{pmatrix} \in \mathbb{R}^{N_{var-ineq} \times \dim(\mathbf{q}_{var})}$$

**:variant-config-inequality-constraint-vector**  $\mathcal{E}key$  (*update? t*)

[method]

$$\text{return } d_{var} := \begin{pmatrix} d_\theta \\ d_{\hat{w}} \\ d_\tau \end{pmatrix} \in \mathbb{R}^{N_{var-ineq}}$$

**:invariant-config-inequality-constraint-matrix**  $\mathcal{E}key$  (*update? nil*)

[method]

$$C_\phi \Delta\phi \geq d_\phi \quad (2.67)$$

$$\Leftrightarrow C_{invar} \Delta q_{invar} \geq d_{invar} \quad (2.68)$$

$$\text{return } C_{invar} := C_\phi \in \mathbb{R}^{N_{invar-ineq} \times \dim(\mathbf{q}_{invar})}$$

**:invariant-config-inequality-constraint-vector**  $\mathcal{E}key$  (*update? t*)

[method]

$$\begin{cases} C_{var} \Delta \mathbf{q}_{var} \geq \mathbf{d}_{var} \\ C_{invar} \Delta \mathbf{q}_{invar} \geq \mathbf{d}_{invar} \\ C_{col} \begin{pmatrix} \Delta \mathbf{q}_{var} \\ \Delta \mathbf{q}_{invar} \end{pmatrix} \geq \mathbf{d}_{col} \end{cases} \quad (2.69)$$

$$\Leftrightarrow \begin{pmatrix} \mathbf{C}_{var} \\ \hline \mathbf{C}_{invar} \\ \mathbf{C}_{col} \end{pmatrix} \begin{pmatrix} \Delta \mathbf{q}_{var} \\ \Delta \mathbf{q}_{invar} \end{pmatrix} \geq \begin{pmatrix} \mathbf{d}_{var} \\ \mathbf{d}_{invar} \\ \mathbf{d}_{col} \end{pmatrix} \quad (2.70)$$

$$\Leftrightarrow C\Delta q \geq d \quad (2.71)$$

$$\text{return } C := \begin{pmatrix} C_{var} & \\ & C_{invar} \\ \hline & C_{col} \end{pmatrix} \in \mathbb{R}^{N_{ineq} \times \dim(\mathbf{q})}$$

$$\text{return } \mathbf{d} := \begin{pmatrix} \mathbf{d}_{var} \\ \mathbf{d}_{invar} \\ \mathbf{d}_{col} \end{pmatrix} \in \mathbb{R}^{N_{ineq}}$$

return  $\mathbf{A}_{var} \in \mathbb{R}^{0 \times \dim(\mathbf{q}_{var})}$  (no equality constraint)

return  $\mathbf{b}_{var} \in \mathbb{R}^0$  (no equality constraint)

return  $\mathbf{A}_{invar} \in \mathbb{R}^{0 \times \dim(\mathbf{q}_{invar})}$  (no equality constraint)

return  $\mathbf{b}_{invar} \in \mathbb{R}^0$  (no equality constraint)

return  $\mathbf{A} \in \mathbb{R}^{0 \times \dim(\mathbf{q})}$  (no equality constraint)

return  $\mathbf{b} \in \mathbb{R}^0$  (no equality constraint)

二次形式の正則化項として次式を考える.

$$F_{tau}(\mathbf{q}) = \left\| \frac{\boldsymbol{\tau}}{\tau_{max}} \right\|^2 \quad (\text{ベクトルの要素ごとの割り算を表す}) \quad (2.72)$$

$$= \tau^T \bar{W}_{tra} \tau \quad (2.73)$$

ここで,

$$\bar{\mathbf{W}}_{trq} := \begin{pmatrix} \frac{1}{\tau_{max,1}^2} & & & \\ & \frac{1}{\tau_{max,2}^2} & & \\ & & \ddots & \\ & & & \frac{1}{\tau_{max,N_{drive-joint}}^2} \end{pmatrix} \in \mathbb{R}^{dim(\boldsymbol{\tau}) \times dim(\boldsymbol{\tau})} \quad (2.74)$$

*only-variant? is true:*

$$\mathbf{W}_{trq} := \begin{matrix} & dim(\boldsymbol{\theta}) & dim(\hat{\mathbf{w}}) & dim(\boldsymbol{\tau}) \\ \begin{matrix} dim(\boldsymbol{\theta}) \\ dim(\hat{\mathbf{w}}) \\ dim(\boldsymbol{\tau}) \end{matrix} & \begin{pmatrix} & & \\ & & \\ & & \bar{\mathbf{W}}_{trq} \end{pmatrix} \end{matrix} \in \mathbb{R}^{dim(\mathbf{q}_{var}) \times dim(\mathbf{q}_{var})} \quad (2.75)$$

*otherwise:*

$$\mathbf{W}_{trq} := \begin{matrix} & dim(\boldsymbol{\theta}) & dim(\hat{\mathbf{w}}) & dim(\boldsymbol{\tau}) & dim(\phi) \\ \begin{matrix} dim(\boldsymbol{\theta}) \\ dim(\hat{\mathbf{w}}) \\ dim(\boldsymbol{\tau}) \\ dim(\phi) \end{matrix} & \begin{pmatrix} & & & \\ & & & \\ & & \bar{\mathbf{W}}_{trq} & \\ & & & \end{pmatrix} \end{matrix} \in \mathbb{R}^{dim(\mathbf{q}) \times dim(\mathbf{q})} \quad (2.76)$$

return  $\mathbf{W}_{trq}$

**:torque-regular-vector**  $\mathcal{E}key$  (*update? t*) [method]  
(*only-variant? nil*)

$$\bar{\mathbf{v}}_{trq} := \bar{\mathbf{W}}_{trq} \boldsymbol{\tau} \quad (2.77)$$

$$= \begin{pmatrix} \frac{\tau_1}{\tau_{max,1}^2} \\ \frac{\tau_2}{\tau_{max,2}^2} \\ \vdots \\ \frac{\tau_{dim(\boldsymbol{\tau})}}{\tau_{max,dim(\boldsymbol{\tau})}^2} \end{pmatrix} \in \mathbb{R}^{dim(\boldsymbol{\tau})} \quad (2.78)$$

*only-variant? is true:*

$$\mathbf{v}_{trq} := \begin{matrix} & 1 \\ \begin{matrix} dim(\boldsymbol{\theta}) \\ dim(\hat{\mathbf{w}}) \\ dim(\boldsymbol{\tau}) \end{matrix} & \begin{pmatrix} \\ \\ \bar{\mathbf{v}}_{trq} \end{pmatrix} \end{matrix} \in \mathbb{R}^{dim(\mathbf{q}_{var})} \quad (2.79)$$

*otherwise:*

$$\mathbf{v}_{trq} := \begin{matrix} & 1 \\ \begin{matrix} dim(\boldsymbol{\theta}) \\ dim(\hat{\mathbf{w}}) \\ dim(\boldsymbol{\tau}) \\ dim(\phi) \end{matrix} & \begin{pmatrix} \\ \\ \bar{\mathbf{v}}_{trq} \\ \end{pmatrix} \end{matrix} \in \mathbb{R}^{dim(\mathbf{q})} \quad (2.80)$$

return  $\mathbf{v}_{trq}$

:torque-ratio

[method]

$$\text{return } \frac{\boldsymbol{\tau}}{\tau_{max}} := \begin{pmatrix} \frac{\tau_1}{\tau_{max,1}} \\ \frac{\tau_2}{\tau_{max,2}} \\ \vdots \\ \frac{\tau_{N_{drive-joint}}}{\tau_{max,N_{drive-joint}}} \end{pmatrix}$$

:regular-matrix

[method]

$$\mathbf{W}_{reg} := \min(k_{max}, \|\mathbf{e}\|^2 + k_{off})\mathbf{I} + k_{trq}\mathbf{W}_{trq} \quad (2.81)$$

$$\text{return } \mathbf{W}_{reg} \in \mathbb{R}^{dim(\mathbf{q}) \times dim(\mathbf{q})}$$

:regular-vector

[method]

$$\mathbf{v}_{reg} := k_{trq}\mathbf{v}_{trq} \quad (2.82)$$

$$\text{return } \mathbf{v}_{reg} \in \mathbb{R}^{dim(\mathbf{q})}$$

:update-collision-inequality-constraint

[method]

リンク 1 とリンク 2 の最近点を  $\mathbf{p}_1, \mathbf{p}_2$  とする．リンク 1 とリンク 2 が干渉しない条件を，最近点  $\mathbf{p}_1, \mathbf{p}_2$  の距離が  $d_{margin}$  以上である条件に置き換えて考える．これは次式で表される．

$$\mathbf{d}_{12}^T(\mathbf{p}_1 - \mathbf{p}_2) \geq d_{margin} \quad (2.83)$$

$$\text{where } \mathbf{d}_{12} = \mathbf{p}_1 - \mathbf{p}_2 \quad (2.84)$$

コンフィギュレーションが  $\Delta \mathbf{q}$  だけ更新されてもこれが成立するための条件は次式で表される．

$$\mathbf{d}_{12}^T \{(\mathbf{p}_1 + \Delta \mathbf{p}_1) - (\mathbf{p}_2 + \Delta \mathbf{p}_2)\} \geq d_{margin} \quad (2.85)$$

$$\text{where } \Delta \mathbf{p}_1 = \mathbf{J}_{\theta,1}\Delta \boldsymbol{\theta} + \mathbf{J}_{\phi,1}\Delta \boldsymbol{\phi} \quad (2.86)$$

$$\Delta \mathbf{p}_2 = \mathbf{J}_{\theta,2}\Delta \boldsymbol{\theta} + \mathbf{J}_{\phi,2}\Delta \boldsymbol{\phi} \quad (2.87)$$

$$\mathbf{J}_{\theta,i} = \frac{\partial \mathbf{p}_i}{\partial \boldsymbol{\theta}}, \quad \mathbf{J}_{\phi,i} = \frac{\partial \mathbf{p}_i}{\partial \boldsymbol{\phi}} \quad (i = 1, 2) \quad (2.88)$$

これは以下のように変形される．

$$\mathbf{d}_{12}^T \{(\mathbf{p}_1 + \mathbf{J}_{\theta,1}\Delta \boldsymbol{\theta} + \mathbf{J}_{\phi,1}\Delta \boldsymbol{\phi}) - (\mathbf{p}_2 + \mathbf{J}_{\theta,2}\Delta \boldsymbol{\theta} + \mathbf{J}_{\phi,2}\Delta \boldsymbol{\phi})\} \geq d_{margin} \quad (2.89)$$

$$\Leftrightarrow \mathbf{d}_{12}^T(\mathbf{J}_{\theta,1} - \mathbf{J}_{\theta,2})\Delta \boldsymbol{\theta} + \mathbf{d}_{12}^T(\mathbf{J}_{\phi,1} - \mathbf{J}_{\phi,2})\Delta \boldsymbol{\phi} \geq -(\mathbf{d}_{12}^T(\mathbf{p}_1 - \mathbf{p}_2) - d_{margin}) \quad (2.90)$$

$$\Leftrightarrow \mathbf{c}_{col,var}^T \Delta \boldsymbol{\theta} + \mathbf{c}_{col,invar}^T \Delta \boldsymbol{\phi} \geq d_{col} \quad (2.91)$$

$$\text{where } \mathbf{c}_{col,var}^T = \mathbf{d}_{12}^T(\mathbf{J}_{\theta,1} - \mathbf{J}_{\theta,2}) \quad (2.92)$$

$$\mathbf{c}_{col,invar}^T = \mathbf{d}_{12}^T(\mathbf{J}_{\phi,1} - \mathbf{J}_{\phi,2}) \quad (2.93)$$

$$d_{col} = -(\mathbf{d}_{12}^T(\mathbf{p}_1 - \mathbf{p}_2) - d_{margin}) \quad (2.94)$$

$i$  番目の干渉回避リンクペアに関する行列，ベクトルをそれぞれ  $\mathbf{c}_{col,var,i}^T, \mathbf{c}_{col,invar,i}^T, d_{col,i}$  とする． $i =$

$1, 2, \dots, N_{col}$  の全てのリンクペアにおいて干渉が生じないための条件は次式で表される .

$$\begin{pmatrix} \mathbf{C}_{col,\theta} & \mathbf{C}_{col,\phi} \end{pmatrix} \begin{pmatrix} \Delta\theta \\ \Delta\phi \end{pmatrix} \geq \mathbf{d}_{col} \quad (2.95)$$

$$\mathbf{C}_{col,\theta} := \begin{pmatrix} \mathbf{c}_{col,var,1}^T \\ \vdots \\ \mathbf{c}_{col,var,N_{col}}^T \end{pmatrix} \in \mathbb{R}^{N_{col} \times \dim(\theta)} \quad (2.96)$$

$$\mathbf{C}_{col,\phi} := \begin{pmatrix} \mathbf{c}_{col,invar,1}^T \\ \vdots \\ \mathbf{c}_{col,invar,N_{col}}^T \end{pmatrix} \in \mathbb{R}^{N_{col} \times \dim(\phi)}, \quad (2.97)$$

$$\mathbf{d}_{col} := \begin{pmatrix} d_{col,1} \\ \vdots \\ d_{col,N_{col}} \end{pmatrix} \in \mathbb{R}^{N_{col}} \quad (2.98)$$

update inequality matrix  $\mathbf{C}_{col,\theta}, \mathbf{C}_{col,\phi}$  and inequality vector  $\mathbf{d}_{col}$  for collision avoidance

**:collision-theta-inequality-constraint-matrix** [method]

return  $\mathbf{C}_{col,\theta} \in \mathbb{R}^{N_{col} \times \dim(\theta)}$

**:collision-phi-inequality-constraint-matrix** [method]

return  $\mathbf{C}_{col,\phi} \in \mathbb{R}^{N_{col} \times \dim(\phi)}$

**:collision-inequality-constraint-matrix** *key (update? nil)* [method]

$$\mathbf{C}_{col} := N_{col} \begin{pmatrix} \dim(\theta) & \dim(\hat{\mathbf{w}}) & \dim(\tau) & \dim(\phi) \\ \mathbf{C}_{col,\theta} & \mathbf{O} & \mathbf{O} & \mathbf{C}_{col,\phi} \end{pmatrix} \quad (2.99)$$

return  $\mathbf{C}_{col} \in \mathbb{R}^{N_{col} \times \dim(\mathbf{q})}$

**:collision-inequality-constraint-vector** *key (update? nil)* [method]

return  $\mathbf{d}_{col} \in \mathbb{R}^{N_{col}}$

**:update-viewer** [method]

Update viewer.

**:print-status** [method]

Print status.

## 2.2 軌道コンフィギュレーションと軌道タスク関数

**trajectory-configuration-task** [class]

**:super** **propertied-object**

**:slots** (*\_instant-config-task-list* list of instant-config-task instance)

(*\_num-instant-config-task* L)

(*\_dim-variant-config*  $\dim(\mathbf{q}_{var})$ )

(*\_dim-invariant-config*  $\dim(\mathbf{q}_{invar})$ )

```

(dim-config  $\dim(\mathbf{q})$ )
(dim-task  $\dim(\mathbf{e})$ )
(norm-regular-scale-max  $k_{max}$ )
(norm-regular-scale-offset  $k_{off}$ )
(adjacent-regular-scale  $k_{adj}$ )
(torque-regular-scale  $k_{trq}$ )
(task-jacobi buffer for  $\frac{\partial \mathbf{e}}{\partial \mathbf{q}}$ )

```

軌道コンフィギュレーション  $\mathbf{q}$  と軌道タスク関数  $\mathbf{e}(\mathbf{q})$  のクラス .

以降では , 説明文やメソッド名で , “軌道” や “trajectory” を省略する .

コンフィギュレーション  $\mathbf{q}$  の取得・更新 , タスク関数  $\mathbf{e}(\mathbf{q})$  の取得 , タスク関数のヤコビ行列  $\frac{\partial \mathbf{e}(\mathbf{q})}{\partial \mathbf{q}}$  の取得 , コンフィギュレーションの等式・不等式制約  $A, b, C, d$  の取得のためのメソッドが定義されている .

コンフィギュレーション・タスク関数を定めるために , 初期化時に以下を与える

- 瞬時のコンフィギュレーション・タスクのリスト

`instant-config-task-list` `instant-configuration-task` のリスト

- 目的関数の重み

`norm-regular-scale-max`  $k_{max}$  コンフィギュレーション更新量正則化の重み最大値

`norm-regular-scale-offset`  $k_{off}$  コンフィギュレーション更新量正則化の重みオフセット

`adjacent-regular-scale`  $k_{adj}$  隣接コンフィギュレーション正則化の重み

`torque-regular-scale`  $k_{trq}$  トルク正則化の重み

コンフィギュレーション  $\mathbf{q}$  は以下から構成される .

$$\mathbf{q} := \begin{pmatrix} \mathbf{q}_{var}^{(1)T} & \mathbf{q}_{var}^{(2)T} & \cdots & \mathbf{q}_{var}^{(L)T} & \mathbf{q}_{invar}^T \end{pmatrix}^T \quad (2.100)$$

ここで ,

$$\mathbf{q}_{invar} := \mathbf{q}_{invar}^{(1)} = \mathbf{q}_{invar}^{(2)} = \cdots = \mathbf{q}_{invar}^{(L)} \quad (2.101)$$

$\mathbf{q}_{var}^{(l)}, \mathbf{q}_{invar}^{(l)}$  ( $l = 1, 2, \dots, L$ ) は  $l$  番目の瞬時の時変 , 時不変コンフィギュレーションを表す .

タスク関数  $\mathbf{e}(\mathbf{q})$  は以下から構成される .

$$\mathbf{e}(\mathbf{q}) := \begin{pmatrix} \mathbf{e}^{(1)T}(\mathbf{q}_{var}^{(1)}, \mathbf{q}_{invar}^{(1)}) & \mathbf{e}^{(2)T}(\mathbf{q}_{var}^{(2)}, \mathbf{q}_{invar}^{(2)}) & \cdots & \mathbf{e}^{(L)T}(\mathbf{q}_{var}^{(L)}, \mathbf{q}_{invar}^{(L)}) \end{pmatrix}^T \quad (2.102)$$

$\mathbf{e}^{(l)}(\mathbf{q}_{var}^{(l)}, \mathbf{q}_{invar}^{(l)})$  ( $l = 1, 2, \dots, L$ ) は  $l$  番目の瞬時のタスク関数を表す .

**:init** *key* (*name*) [method]

(*instant-config-task-list*)

(*norm-regular-scale-max* 1.000000e-04)

(*norm-regular-scale-offset* 1.000000e-07)

(*adjacent-regular-scale* 0.005)

(*torque-regular-scale* 0.001)

Initialize instance

**:instant-config-task-list** [method]

return instant-config-task-list



**:dim-variant-config** [method]  
 return  $\dim(\mathbf{q}_{var}) := \sum_{l=1}^L \dim(\mathbf{q}_{var}^{(l)})$

**:dim-invariant-config** [method]  
 return  $\dim(\mathbf{q}_{invar}) := \dim(\mathbf{q}_{invar}^{(l)})$  ( $l = 1, 2, \dots, L$  で同じ)

**:dim-config** [method]  
 return  $\dim(\mathbf{q}) := \dim(\mathbf{q}_{var}) + \dim(\mathbf{q}_{invar})$

**:dim-task** [method]  
 return  $\dim(\mathbf{e}) := \sum_{l=1}^L \dim(\mathbf{e}^{(l)})$

**:variant-config-vector** [method]  
 return  $\mathbf{q}_{var} := \begin{pmatrix} \mathbf{q}_{var}^{(1)} \\ \mathbf{q}_{var}^{(2)} \\ \vdots \\ \mathbf{q}_{var}^{(L)} \end{pmatrix}$

**:invariant-config-vector** [method]  
 return  $\mathbf{q}_{invar} := \mathbf{q}_{invar}^{(l)}$  ( $l = 1, 2, \dots, L$  で同じ)

**:config-vector** [method]  
 return  $\mathbf{q} := \begin{pmatrix} \mathbf{q}_{var} \\ \mathbf{q}_{invar} \end{pmatrix} = \begin{pmatrix} \mathbf{q}_{var}^{(1)} \\ \mathbf{q}_{var}^{(2)} \\ \vdots \\ \mathbf{q}_{var}^{(L)} \\ \mathbf{q}_{invar} \end{pmatrix}$

**:set-variant-config** *variant-config-new* *ℰkey* (*relative?* *nil*) [method]  
 (*apply-to-robot?* *t*)

Set  $\mathbf{q}_{var}$ .

**:set-invariant-config** *invariant-config-new* *ℰkey* (*relative?* *nil*) [method]  
 (*apply-to-robot?* *t*)

Set  $\mathbf{q}_{invar}$ .

**:set-config** *config-new* *ℰkey* (*relative?* *nil*) [method]  
 (*apply-to-robot?* *t*)

Set  $\mathbf{q}$ .

**:task-value** *ℰkey* (*update?* *t*) [method]  
 return  $\mathbf{e}(\mathbf{q}) := \begin{pmatrix} \mathbf{e}^{(1)}(\mathbf{q}_{var}^{(1)}, \mathbf{q}_{invar}) \\ \mathbf{e}^{(2)}(\mathbf{q}_{var}^{(2)}, \mathbf{q}_{invar}) \\ \vdots \\ \mathbf{e}^{(L)}(\mathbf{q}_{var}^{(L)}, \mathbf{q}_{invar}) \end{pmatrix}$

**:variant-task-jacobian** [method]

$$\frac{\partial \mathbf{e}}{\partial \mathbf{q}_{var}} = \begin{pmatrix} \frac{\partial \mathbf{e}^{(1)}}{\partial \mathbf{q}_{var}^{(1)}} & & \mathbf{O} \\ & \frac{\partial \mathbf{e}^{(2)}}{\partial \mathbf{q}_{var}^{(2)}} & \\ & & \ddots \\ \mathbf{O} & & & \frac{\partial \mathbf{e}^{(L)}}{\partial \mathbf{q}_{var}^{(L)}} \end{pmatrix} \quad (2.103)$$

return  $\frac{\partial \mathbf{e}}{\partial \mathbf{q}_{var}} \in \mathbb{R}^{dim(\mathbf{e}) \times dim(\mathbf{q}_{var})}$

**:invariant-task-jacobian**

[method]

$$\frac{\partial \mathbf{e}}{\partial \mathbf{q}_{invar}} = \begin{pmatrix} \frac{\partial \mathbf{e}^{(1)}}{\partial \mathbf{q}_{invar}^{(1)}} \\ \frac{\partial \mathbf{e}^{(2)}}{\partial \mathbf{q}_{invar}^{(2)}} \\ \vdots \\ \frac{\partial \mathbf{e}^{(L)}}{\partial \mathbf{q}_{invar}^{(L)}} \end{pmatrix} \quad (2.104)$$

return  $\frac{\partial \mathbf{e}}{\partial \mathbf{q}_{invar}} \in \mathbb{R}^{dim(\mathbf{e}) \times dim(\mathbf{q}_{invar})}$

**:task-jacobian**

[method]

$$\frac{\partial \mathbf{e}}{\partial \mathbf{q}} = \begin{pmatrix} \frac{\partial \mathbf{e}}{\partial \mathbf{q}_{var}} & \frac{\partial \mathbf{e}}{\partial \mathbf{q}_{invar}} \end{pmatrix} \quad (2.105)$$

$$= \begin{pmatrix} \frac{\partial \mathbf{e}^{(1)}}{\partial \mathbf{q}_{var}^{(1)}} & & \mathbf{O} & \frac{\partial \mathbf{e}^{(1)}}{\partial \mathbf{q}_{invar}^{(1)}} \\ & \frac{\partial \mathbf{e}^{(2)}}{\partial \mathbf{q}_{var}^{(2)}} & & \frac{\partial \mathbf{e}^{(2)}}{\partial \mathbf{q}_{invar}^{(2)}} \\ & & \ddots & \\ \mathbf{O} & & & \frac{\partial \mathbf{e}^{(L)}}{\partial \mathbf{q}_{var}^{(L)}} & \frac{\partial \mathbf{e}^{(L)}}{\partial \mathbf{q}_{invar}^{(L)}} \end{pmatrix} \quad (2.106)$$

return  $\frac{\partial \mathbf{e}}{\partial \mathbf{q}} \in \mathbb{R}^{dim(\mathbf{e}) \times dim(\mathbf{q})}$

**:variant-config-inequality-constraint-matrix** *ℰkey (update? nil)*

[method]

$$\mathbf{C}_{var} := \begin{pmatrix} \mathbf{C}_{var}^{(1)} & & \mathbf{O} \\ & \mathbf{C}_{var}^{(2)} & \\ & & \ddots \\ \mathbf{O} & & & \mathbf{C}_{var}^{(L)} \end{pmatrix} \quad (2.107)$$

return  $\mathbf{C}_{var} \in \mathbb{R}^{N_{var-ineq} \times dim(\mathbf{q}_{var})}$

**:variant-config-inequality-constraint-vector** *ℰkey (update? t)*

[method]

$$\mathbf{d}_{var} := \begin{pmatrix} \mathbf{d}_{var}^{(1)} \\ \mathbf{d}_{var}^{(2)} \\ \vdots \\ \mathbf{d}_{var}^{(L)} \end{pmatrix} \quad (2.108)$$

return  $\mathbf{d}_{var} \in \mathbb{R}^{N_{var-ineq}}$

**:invariant-config-inequality-constraint-matrix**  $\mathcal{E}key$  (*update?* *nil*) [method]

$$\mathbf{C}_{invar} := \mathbf{C}_{invar}^{(l)} \quad (l = 1, 2, \dots, L \text{ で同じ}) \quad (2.109)$$

return  $\mathbf{C}_{invar} \in \mathbb{R}^{N_{invar-ineq} \times \dim(\mathbf{q}_{invar})}$

**:invariant-config-inequality-constraint-vector**  $\mathcal{E}key$  (*update?* *t*) [method]

$$\mathbf{d}_{invar} := \mathbf{d}_{invar}^{(l)} \quad (l = 1, 2, \dots, L \text{ で同じ}) \quad (2.110)$$

return  $\mathbf{d}_{invar} \in \mathbb{R}^{N_{invar-ineq}}$

**:config-inequality-constraint-matrix**  $\mathcal{E}key$  (*update?* *nil*) [method]  
(*update-collision?* *nil*)

$$\mathbf{C} := \begin{pmatrix} \mathbf{C}_{var} & & \\ & \mathbf{C}_{invar} & \\ \hline & & \mathbf{C}_{col} \end{pmatrix} \in \mathbb{R}^{N_{ineq} \times \dim(\mathbf{q})} \quad (2.111)$$

return  $\mathbf{C} \in \mathbb{R}^{N_{ineq} \times \dim(\mathbf{q})}$

**:config-inequality-constraint-vector**  $\mathcal{E}key$  (*update?* *t*) [method]  
(*update-collision?* *nil*)

$$\mathbf{d} := \begin{pmatrix} \mathbf{d}_{var} \\ \mathbf{d}_{invar} \\ \mathbf{d}_{col} \end{pmatrix} \quad (2.112)$$

return  $\mathbf{d} \in \mathbb{R}^{N_{ineq}}$

**:variant-config-equality-constraint-matrix**  $\mathcal{E}key$  (*update?* *nil*) [method]

$$\mathbf{A}_{var} := \begin{pmatrix} \mathbf{A}_{var}^{(1)} & & & \mathbf{O} \\ & \mathbf{A}_{var}^{(2)} & & \\ & & \ddots & \\ \mathbf{O} & & & \mathbf{A}_{var}^{(L)} \end{pmatrix} \quad (2.113)$$

return  $\mathbf{A}_{var} \in \mathbb{R}^{N_{var-eq} \times \dim(\mathbf{q}_{var})}$

**:variant-config-equality-constraint-vector**  $\mathcal{E}key$  (*update?* *t*) [method]

$$\mathbf{b}_{var} := \begin{pmatrix} \mathbf{b}_{var}^{(1)} \\ \mathbf{b}_{var}^{(2)} \\ \vdots \\ \mathbf{b}_{var}^{(L)} \end{pmatrix} \quad (2.114)$$

return  $\mathbf{b}_{var} \in \mathbb{R}^{N_{var-eq}}$

**:invariant-config-equality-constraint-matrix**  $\mathcal{E}key$  (*update?* *nil*) [method]

$$\mathbf{A}_{invar} := \mathbf{A}_{invar}^{(l)} \quad (l = 1, 2, \dots, L \text{ で同じ}) \quad (2.115)$$

return  $\mathbf{A}_{invar} \in \mathbb{R}^{N_{invar-eq} \times \dim(\mathbf{q}_{invar})}$

**:invariant-config-equality-constraint-vector**  $\mathcal{E}key$  (*update?* *t*) [method]

$$\mathbf{b}_{invar} := \mathbf{b}_{invar}^{(l)} \quad (l = 1, 2, \dots, L \text{ で同じ}) \quad (2.116)$$

return  $\mathbf{b}_{invar} \in \mathbb{R}^{N_{invar-eq}}$

**:config-equality-constraint-matrix**  $\mathcal{E}key$  (*update?* *nil*) [method]

$$\mathbf{A} := \begin{pmatrix} \mathbf{A}_{var} \\ \mathbf{A}_{invar} \end{pmatrix} \in \mathbb{R}^{N_{eq} \times \dim(\mathbf{q})} \quad (2.117)$$

return  $\mathbf{A} \in \mathbb{R}^{N_{eq} \times \dim(\mathbf{q})}$

**:config-equality-constraint-vector**  $\mathcal{E}key$  (*update?* *t*) [method]

$$\mathbf{b} := \begin{pmatrix} \mathbf{b}_{var} \\ \mathbf{b}_{invar} \end{pmatrix} \quad (2.118)$$

return  $\mathbf{b} \in \mathbb{R}^{N_{eq}}$

**:update-collision-inequality-constraint** [method]

update inequality matrix  $\mathbf{C}_{col,\theta}^{(l)}, \mathbf{C}_{col,\phi}^{(l)}$  and inequality vector  $\mathbf{d}_{col}^{(l)}$  for collision avoidance ( $l = 1, 2, \dots, L$ )

**:collision-inequality-constraint-matrix**  $\mathcal{E}key$  (*update?* *nil*) [method]

$$\hat{\mathbf{C}}_{col,\theta}^{(l)} := N_{col}^{(l)} \begin{pmatrix} \dim(\boldsymbol{\theta}^{(l)}) & \dim(\hat{\mathbf{w}}^{(l)}) & \dim(\boldsymbol{\tau}^{(l)}) \\ \mathbf{C}_{col,\theta}^{(l)} & \mathbf{O} & \mathbf{O} \end{pmatrix} \quad (2.119)$$

$$\mathbf{C}_{col} := \begin{pmatrix} \hat{\mathbf{C}}_{col,\theta}^{(1)} & & & \mathbf{C}_{col,\phi}^{(1)} \\ & \hat{\mathbf{C}}_{col,\theta}^{(2)} & & \mathbf{C}_{col,\phi}^{(2)} \\ & & \ddots & \vdots \\ & & & \hat{\mathbf{C}}_{col,\theta}^{(L)} & \mathbf{C}_{col,\phi}^{(L)} \end{pmatrix} \quad (2.120)$$

return  $\mathbf{C}_{col} \in \mathbb{R}^{N_{col} \times \dim(\mathbf{q})}$

**:collision-inequality-constraint-vector**  $\mathcal{E}key$  (*update?* *nil*) [method]

$$\mathbf{d}_{col} := \begin{pmatrix} \mathbf{d}_{col}^{(1)} \\ \mathbf{d}_{col}^{(2)} \\ \vdots \\ \mathbf{d}_{col}^{(L)} \end{pmatrix} \quad (2.121)$$

return  $\mathbf{d}_{col} \in \mathbb{R}^{N_{col}}$

**:adjacent-regular-matrix** *ℰkey (update? nil)* [method]  
 二次形式の正則化項として次式を考える .

$$F_{adj}(\mathbf{q}) = \sum_{l=1}^{L-1} \|\boldsymbol{\theta}_{l+1} - \boldsymbol{\theta}_l\|^2 \quad (2.122)$$

$$= \mathbf{q}^T \mathbf{W}_{adj} \mathbf{q} \quad (2.123)$$

ここで ,

$$\bar{\mathbf{I}}_{adj} := \begin{matrix} & \dim(\boldsymbol{\theta}^{(l)}) & \dim(\hat{\mathbf{w}}^{(l)}) & \dim(\boldsymbol{\tau}^{(l)}) \\ \begin{matrix} \dim(\boldsymbol{\theta}^{(l)}) \\ \dim(\hat{\mathbf{w}}^{(l)}) \\ \dim(\boldsymbol{\tau}^{(l)}) \end{matrix} & \begin{pmatrix} \mathbf{I} & & \\ & & \\ & & \end{pmatrix} \end{matrix} \in \mathbb{R}^{\dim(\mathbf{q}_{var}^{(l)}) \times \dim(\mathbf{q}_{var}^{(l)})} \quad (2.124)$$

$$\bar{\mathbf{W}}_{adj} := \begin{pmatrix} \bar{\mathbf{I}}_{adj} & -\bar{\mathbf{I}}_{adj} & & & & \\ -\bar{\mathbf{I}}_{adj} & 2\bar{\mathbf{I}}_{adj} & -\bar{\mathbf{I}}_{adj} & & & \mathbf{O} \\ & & \ddots & & & \\ & & & -\bar{\mathbf{I}}_{adj} & 2\bar{\mathbf{I}}_{adj} & -\bar{\mathbf{I}}_{adj} \\ \mathbf{O} & & & -\bar{\mathbf{I}}_{adj} & \bar{\mathbf{I}}_{adj} & \end{pmatrix} \in \mathbb{R}^{\dim(\mathbf{q}_{var}) \times \dim(\mathbf{q}_{var})} \quad (2.125)$$

$$\mathbf{W}_{adj} := \begin{pmatrix} \bar{\mathbf{W}}_{adj} \\ \mathbf{O} \end{pmatrix} \quad (2.126)$$

return  $\mathbf{W}_{adj} \in \mathbb{R}^{\dim(\mathbf{q}) \times \dim(\mathbf{q})}$

**:adjacent-regular-vector** *ℰkey (update? t)* [method]

$$\mathbf{v}_{adj} := \mathbf{W}_{adj} \mathbf{q} \quad (2.127)$$

return  $\mathbf{v}_{adj} \in \mathbb{R}^{\dim(\mathbf{q})}$

**:torque-regular-matrix** *ℰkey (update? nil)* [method]

$$\bar{\mathbf{W}}_{trq} := \begin{pmatrix} \mathbf{W}_{trq}^{(1)} & & & \mathbf{O} \\ & \mathbf{W}_{trq}^{(2)} & & \\ & & \ddots & \\ \mathbf{O} & & & \mathbf{W}_{trq}^{(L)} \end{pmatrix} \in \mathbb{R}^{\dim(\mathbf{q}_{var}) \times \dim(\mathbf{q}_{var})} \quad (2.128)$$

$$\mathbf{W}_{trq} := \begin{pmatrix} \bar{\mathbf{W}}_{trq} \\ \mathbf{O} \end{pmatrix} \quad (2.129)$$

return  $\mathbf{W}_{trq} \in \mathbb{R}^{\dim(\mathbf{q}) \times \dim(\mathbf{q})}$

**:torque-regular-vector** *ℰkey (update? t)* [method]

$$\bar{\mathbf{v}}_{trq} := \begin{pmatrix} \mathbf{v}_{trq}^{(1)} \\ \mathbf{v}_{trq}^{(2)} \\ \vdots \\ \mathbf{v}_{trq}^{(L)} \end{pmatrix} \in \mathbb{R}^{\dim(\mathbf{q}_{var})} \quad (2.130)$$

$$\mathbf{v}_{trq} := \begin{pmatrix} \bar{\mathbf{v}}_{trq} \\ \mathbf{0} \end{pmatrix} \quad (2.131)$$

return  $\mathbf{v}_{trq} \in \mathbb{R}^{dim(\mathbf{q})}$

**:regular-matrix** [method]

$$\mathbf{W}_{reg} := \min(k_{max}, \|\mathbf{e}\|^2 + k_{off})\mathbf{I} + k_{adj}\mathbf{W}_{adj} + k_{trq}\mathbf{W}_{trq} \quad (2.132)$$

return  $\mathbf{W}_{reg} \in \mathbb{R}^{dim(\mathbf{q}) \times dim(\mathbf{q})}$

**:regular-vector** [method]

$$\mathbf{v}_{reg} := k_{adj}\mathbf{v}_{adj} + k_{trq}\mathbf{v}_{trq} \quad (2.133)$$

return  $\mathbf{v}_{reg} \in \mathbb{R}^{dim(\mathbf{q})}$

**:update-viewer** [method]

Update viewer.

**:print-status** [method]

Print status.

**:play-animation** *key* (*robot-env*) [method]

(*loop?* *t*)

(*visualize-callback-func*)

Play motion.

**:generate-robot-state-list** *key* (*robot-env*) [method]

(*joint-name-list* (*send-all* (*send robot-env :robot :joint-list*) *:name*))

(*root-link-name* (*send* (*car* (*send robot-env :robot :links*)) *:name*))

(*step-time* 0.004)

(*divide-num* 100)

(*limb-list* (*list :rleg :lleg :rarm :larm*))

Generate and return robot state list.

### 3 勾配を用いた制約付き非線形最適化

#### 3.1 逐次二次計画法

**sqp-optimization** [class]

:super

**propertied-object**

:slots

(\_config-task instance of configuration-task)

(\_qp-retval buffer for QP return value)

(\_qp-status buffer for QP status)

(\_qp-int-status QP status)

(\_task-value buffer for task value  $\mathbf{e}(\mathbf{q})$ )

(\_task-jacobian buffer for task jacobian  $\frac{\partial \mathbf{e}}{\partial \mathbf{q}}$ )

```

(_dim-config-buf-matrix matrix buffer)
(_convergence-check-func function to check convergence)
(_failure-callback-func callback function of failure)
(_pre-process-func pre-process function)
(_post-process-func post-process function)
(_i buffer for iteration count)
(_status status of sqp optimization)
(_no-visualize? whether to suppress visualization)
(_no-print? whether to suppress print)

```

逐次二次計画法のクラス .

instant-configuration-task クラスや trajectory-configuration-task クラスの instance (以降, configuration-task と呼ぶ) が与えられた時に, configuration-task のタスク関数ノルム二乗  $\|e(q)\|^2$  を最小にするコンフィギュレーション  $q$  を反復計算により求める .

```

:init ℰkey (config-task) [method]
      (convergence-check-func)
      (failure-callback-func)
      (pre-process-func)
      (post-process-func)
      (no-visualize?)
      (no-print?)
      ℰallow-other-keys

```

Initialize instance

```

:config-task [method]
  Return configuration-task instance

```

```

:optimize ℰkey (loop-num 100) [method]
      (loop-num-min)
      (update-viewer-interval 1)
      (print-status-interval 10)

```

Optimize

In each iteration, do following:

1. check convergence
2. call pre-process function
3. print status
4. solve QP and update configuration
5. call post-process function

Solve following QP:

$$\min_{\Delta \mathbf{q}^{(k)}} \frac{1}{2} \Delta \mathbf{q}^{(k)T} \mathbf{W} \Delta \mathbf{q}^{(k)} + \mathbf{v}^T \Delta \mathbf{q}^{(k)} \quad (3.1)$$

$$\text{s.t. } \mathbf{A} \Delta \mathbf{q}^{(k)} = \mathbf{b} \quad (3.2)$$

$$\mathbf{C} \Delta \mathbf{q}^{(k)} \geq \mathbf{d} \quad (3.3)$$

$$\text{where } \mathbf{W} = \left( \frac{\partial \mathbf{e}(\mathbf{q}^{(k)})}{\partial \mathbf{q}^{(k)}} \right)^T \left( \frac{\partial \mathbf{e}(\mathbf{q}^{(k)})}{\partial \mathbf{q}^{(k)}} \right) + \mathbf{W}_{reg} \quad (3.4)$$

$$\mathbf{v} = \left( \frac{\partial \mathbf{e}(\mathbf{q}^{(k)})}{\partial \mathbf{q}^{(k)}} \right)^T \mathbf{e}(\mathbf{q}^{(k)}) + \mathbf{v}_{reg} \quad (3.5)$$

and update configuration:

$$\mathbf{q}^{(k+1)} = \mathbf{q}^{(k)} + \Delta \mathbf{q}^{(k)*} \quad (3.6)$$

**:iteration** [method]

Return iteration index.

**:status** [method]

Return status of sqp optimization.

## 3.2 複数解候補を用いた逐次二次計画法

### 3.2.1 複数解候補を用いた逐次二次計画法の理論

式 (1.4a) の最適化問題に逐次二次計画法などの制約付き非線形最適化手法を適用すると、初期値から勾配方向に進行して至る局所最適解が得られると考えられる。したがって解は初期値に強く依存する。

式 (1.4a) の代わりに、以下の最適化問題を考える。

$$\min_{\hat{\mathbf{q}}} \sum_{i \in \mathcal{I}} \left\{ F(\mathbf{q}^{(i)}) + k_{msc} F_{msc}(\hat{\mathbf{q}}; i) \right\} \quad (3.7)$$

$$\text{s.t. } \mathbf{A} \mathbf{q}^{(i)} = \bar{\mathbf{b}} \quad i \in \mathcal{I} \quad (3.8)$$

$$\mathbf{C} \mathbf{q}^{(i)} \geq \bar{\mathbf{d}} \quad i \in \mathcal{I} \quad (3.9)$$

$$\text{where } \hat{\mathbf{q}} \stackrel{\text{def}}{=} \left( \mathbf{q}^{(1)T} \quad \mathbf{q}^{(2)T} \quad \dots \quad \mathbf{q}^{(N_{msc})T} \right)^T \quad (3.10)$$

$$\mathcal{I} \stackrel{\text{def}}{=} \{1, 2, \dots, N_{msc}\} \quad (3.11)$$

$$F_{msc}(\hat{\mathbf{q}}; i) \stackrel{\text{def}}{=} -\frac{1}{2} \sum_{\substack{j \in \mathcal{I} \\ j \neq i}} \log \|d(\mathbf{q}^{(i)}, \mathbf{q}^{(j)})\|^2 \quad (3.12)$$

$$d(\mathbf{q}^{(i)}, \mathbf{q}^{(j)}) \stackrel{\text{def}}{=} \mathbf{q}^{(i)} - \mathbf{q}^{(j)} \quad (3.13)$$

$N_{msc}$  は解候補の個数で、事前に与えるものとする。 $msc$  は複数解候補 (multiple solution candidates) を表す。これは、複数の解候補を同時に探索し、それぞれの解候補  $\mathbf{q}^{(i)}$  が本来の目的関数  $F(\mathbf{q}^{(i)})$  を小さくして、なおかつ、解候補どうしの距離が大きくなるように最適化することを表している。これにより、初期値に依存した唯一の局所解だけでなく、そこから離れた複数の局所解を得ることが可能となり、通常の最適化に比べてより良い解が得られることが期待される。以降では、解候補どうしの距離のコストを表す項  $F_{msc}(\hat{\mathbf{q}}; i)$  を解候補分散項と呼ぶ<sup>8</sup>。

<sup>8</sup> 解分散項の  $\log$  を無くすことは適切ではない。なぜなら、 $d = \|d(\mathbf{q}^{(i)}, \mathbf{q}^{(j)})\|$  として、解分散項の勾配は、

$$\frac{\partial}{\partial d} \left( -\frac{1}{2} \log d^2 \right) = -\frac{1}{d} \rightarrow -\infty \quad (d \rightarrow +0) \quad \frac{\partial}{\partial d} \left( -\frac{1}{2} \log d^2 \right) = -\frac{1}{d} \rightarrow 0 \quad (d \rightarrow \infty) \quad (3.14)$$



解候補分散項のヤコビ行列，ヘッセ行列の各成分は次式で得られる<sup>9</sup>．

$$\nabla_i F_{msc}(\hat{\mathbf{q}}; i) = \frac{\partial F_{msc}(\hat{\mathbf{q}}; i)}{\partial \mathbf{q}^{(i)}} \quad (3.16a)$$

$$= -\frac{1}{2} \sum_{\substack{j \in \mathcal{I} \\ j \neq i}} \frac{\partial}{\partial \mathbf{q}^{(i)}} \log \|\mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(j)})\|^2 \quad (3.16b)$$

$$= -\sum_{\substack{j \in \mathcal{I} \\ j \neq i}} \frac{1}{\|\mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(j)})\|^2} \left( \frac{\partial \mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(j)})}{\partial \mathbf{q}^{(i)}} \right)^T \mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(j)}) \quad (3.16c)$$

$$= -\sum_{\substack{j \in \mathcal{I} \\ j \neq i}} \frac{\mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(j)})}{\|\mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(j)})\|^2} \quad (3.16d)$$

$$(3.16e)$$

$$\nabla_k F_{msc}(\hat{\mathbf{q}}; i) = \frac{\partial F_{msc}(\hat{\mathbf{q}}; i)}{\partial \mathbf{q}^{(k)}} \quad k \in \mathcal{I} \wedge k \neq i \quad (3.17a)$$

$$= -\frac{1}{2} \sum_{\substack{j \in \mathcal{I} \\ j \neq i}} \frac{\partial}{\partial \mathbf{q}^{(k)}} \log \|\mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(j)})\|^2 \quad (3.17b)$$

$$= -\frac{1}{2} \frac{\partial}{\partial \mathbf{q}^{(k)}} \log \|\mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(k)})\|^2 \quad (3.17c)$$

$$= -\frac{1}{\|\mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(k)})\|^2} \left( \frac{\partial \mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(k)})}{\partial \mathbf{q}^{(k)}} \right)^T \mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(k)}) \quad (3.17d)$$

$$= \frac{\mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(k)})}{\|\mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(k)})\|^2} \quad (3.17e)$$

$$(3.17f)$$

$$\nabla_{ii}^2 F_{msc}(\hat{\mathbf{q}}; i) = \frac{\partial^2 F_{msc}(\hat{\mathbf{q}}; i)}{\partial \mathbf{q}^{(i)2}} \quad (3.18a)$$

$$= -\sum_{\substack{j \in \mathcal{I} \\ j \neq i}} \frac{\partial}{\partial \mathbf{q}^{(i)}} \left( \left\{ \|\mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(j)})\|^2 \right\}^{-1} \mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(j)}) \right) \quad (3.18b)$$

$$= -\sum_{\substack{j \in \mathcal{I} \\ j \neq i}} \left( -2 \left\{ \|\mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(j)})\|^2 \right\}^{-2} \mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(j)}) \mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(j)})^T + \left\{ \|\mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(j)})\|^2 \right\}^{-1} \mathbf{I} \right) \quad (3.18c)$$

$$= -\sum_{\substack{j \in \mathcal{I} \\ j \neq i}} \left( -\frac{2}{\|\mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(j)})\|^4} \mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(j)}) \mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(j)})^T + \frac{1}{\|\mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(j)})\|^2} \mathbf{I} \right) \quad (3.18d)$$

$$= -\sum_{\substack{j \in \mathcal{I} \\ j \neq i}} \mathbf{H}(\mathbf{q}^{(i)}, \mathbf{q}^{(j)}) \quad (3.18e)$$

となり，最適化により，コンフィギュレーションが近いときほど離れるように更新し，遠くなるとその影響が小さくなる効果が期待される．それに対し， $\log$  が無い場合の勾配は，

$$\frac{\partial}{\partial d} \left( -\frac{1}{2} d^2 \right) = -d \rightarrow 0 \quad (d \rightarrow +0) \quad \frac{\partial}{\partial d} \left( -\frac{1}{2} d^2 \right) = -d \rightarrow -\infty \quad (d \rightarrow \infty) \quad (3.15)$$

となり，コンフィギュレーションが遠くなるほど離れるように更新し，近いときはその影響が小さくなる．これは，コンフィギュレーションが一致する勾配ゼロの点と，無限に離れ発散する最適値をもち，これらは最適化において望まない挙動をもたらす．

<sup>9</sup>ヘッセ行列の導出は以下を参考にした．<https://math.stackexchange.com/questions/175263/gradient-and-hessian-of-general-2-norm>

ただし ,

$$\mathbf{H}(\mathbf{q}^{(i)}, \mathbf{q}^{(j)}) \stackrel{\text{def}}{=} -\frac{2}{\|\mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(j)})\|^4} \mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(j)}) \mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(j)})^T + \frac{1}{\|\mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(j)})\|^2} \mathbf{I} \quad (3.19)$$

$$\nabla_{ik}^2 F_{msc}(\hat{\mathbf{q}}; i) = \frac{\partial^2 F_{msc}(\hat{\mathbf{q}}; i)}{\partial \mathbf{q}^{(i)} \partial \mathbf{q}^{(k)}} \quad k \in \mathcal{I} \wedge k \neq i \quad (3.20a)$$

$$= -\sum_{\substack{j \in \mathcal{I} \\ j \neq i}} \frac{\partial}{\partial \mathbf{q}^{(k)}} \left( \left\{ \|\mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(j)})\|^2 \right\}^{-1} \mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(j)}) \right) \quad (3.20b)$$

$$= -\frac{\partial}{\partial \mathbf{q}^{(k)}} \left( \left\{ \|\mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(k)})\|^2 \right\}^{-1} \mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(k)}) \right) \quad (3.20c)$$

$$= -\left( 2 \left\{ \|\mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(k)})\|^2 \right\}^{-2} \mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(k)}) \mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(k)})^T - \left\{ \|\mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(k)})\|^2 \right\}^{-1} \mathbf{I} \right) \quad (3.20d)$$

$$= -\frac{2}{\|\mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(k)})\|^4} \mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(k)}) \mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(k)})^T + \frac{1}{\|\mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(k)})\|^2} \mathbf{I} \quad (3.20e)$$

$$= \mathbf{H}(\mathbf{q}^{(i)}, \mathbf{q}^{(k)}) \quad (3.20f)$$

$$\nabla_{kk}^2 F_{msc}(\hat{\mathbf{q}}; i) = \frac{\partial^2 F_{msc}(\hat{\mathbf{q}}; i)}{\partial \mathbf{q}^{(k)2}} \quad k \in \mathcal{I} \wedge k \neq i \quad (3.21a)$$

$$= \frac{\partial}{\partial \mathbf{q}^{(k)}} \left( \left\{ \|\mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(k)})\|^2 \right\}^{-1} \mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(k)}) \right) \quad (3.21b)$$

$$= -\left( -\frac{2}{\|\mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(k)})\|^4} \mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(k)}) \mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(k)})^T + \frac{1}{\|\mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(k)})\|^2} \mathbf{I} \right) \quad (3.21c)$$

$$= -\mathbf{H}(\mathbf{q}^{(i)}, \mathbf{q}^{(k)}) \quad (3.21d)$$

$$\nabla_{kl}^2 F_{msc}(\hat{\mathbf{q}}; i) = \frac{\partial^2 F_{msc}(\hat{\mathbf{q}}; i)}{\partial \mathbf{q}^{(k)} \partial \mathbf{q}^{(l)}} \quad k \in \mathcal{I} \wedge l \in \mathcal{I} \wedge k \neq i \wedge l \neq i \wedge k \neq l \quad (3.22a)$$

$$= \frac{\partial}{\partial \mathbf{q}^{(l)}} \left( \left\{ \|\mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(k)})\|^2 \right\}^{-1} \mathbf{d}(\mathbf{q}^{(i)}, \mathbf{q}^{(k)}) \right) \quad (3.22b)$$

$$= \mathbf{O} \quad (3.22c)$$

したがって、解候補分散項のヤコビ行列、ヘッセ行列は次式で表される。

$$\nabla F_{msc}(\hat{\mathbf{q}}; i) = \frac{\partial F_{msc}(\hat{\mathbf{q}}; i)}{\partial \hat{\mathbf{q}}} \quad (3.23a)$$

$$= \begin{pmatrix} \frac{d(\mathbf{q}^{(i)}, \mathbf{q}^{(1)})}{\|d(\mathbf{q}^{(i)}, \mathbf{q}^{(1)})\|^2} \\ \vdots \\ \frac{d(\mathbf{q}^{(i)}, \mathbf{q}^{(i-1)})}{\|d(\mathbf{q}^{(i)}, \mathbf{q}^{(i-1)})\|^2} \\ - \sum_{\substack{j \in \mathcal{I} \\ j \neq i}} \frac{d(\mathbf{q}^{(i)}, \mathbf{q}^{(j)})}{\|d(\mathbf{q}^{(i)}, \mathbf{q}^{(j)})\|^2} \\ \frac{d(\mathbf{q}^{(i)}, \mathbf{q}^{(i+1)})}{\|d(\mathbf{q}^{(i)}, \mathbf{q}^{(i+1)})\|^2} \\ \vdots \\ \frac{d(\mathbf{q}^{(i)}, \mathbf{q}^{(N_{msc})})}{\|d(\mathbf{q}^{(i)}, \mathbf{q}^{(N_{msc})})\|^2} \end{pmatrix} \quad (3.23b)$$

$$\mathbf{v}_{msc} \stackrel{\text{def}}{=} \sum_{i \in \mathcal{I}} \nabla F_{msc}(\hat{\mathbf{q}}; i) \quad (3.23c)$$

$$= 2 \begin{pmatrix} - \sum_{\substack{j \in \mathcal{I} \\ j \neq 1}} \frac{d(\mathbf{q}^{(1)}, \mathbf{q}^{(j)})}{\|d(\mathbf{q}^{(1)}, \mathbf{q}^{(j)})\|^2} \\ \vdots \\ - \sum_{\substack{j \in \mathcal{I} \\ j \neq N_{msc}}} \frac{d(\mathbf{q}^{(N_{msc})}, \mathbf{q}^{(j)})}{\|d(\mathbf{q}^{(N_{msc})}, \mathbf{q}^{(j)})\|^2} \end{pmatrix} \quad (3.23d)$$

$$\nabla^2 F_{msc}(\hat{\mathbf{q}}; i) = \frac{\partial^2 F_{msc}(\hat{\mathbf{q}}; i)}{\partial \hat{\mathbf{q}}^2} \quad (3.24a)$$

$$= \begin{matrix} & 1 & \cdots & i-1 & i & i+1 & \cdots & N_{msc} \\ \begin{matrix} 1 \\ \vdots \\ i-1 \\ i \\ i+1 \\ \vdots \\ N_{msc} \end{matrix} & \begin{pmatrix} -\mathbf{H}_{i,1} & & & \mathbf{H}_{i,1} & & & \\ & \ddots & & \vdots & & & \\ & & -\mathbf{H}_{i,i-1} & \mathbf{H}_{i,i-1} & & & \\ \mathbf{H}_{i,1} & \cdots & \mathbf{H}_{i,i-1} & - \sum_{\substack{j \in \mathcal{I} \\ j \neq i}} \mathbf{H}_{i,j} & \mathbf{H}_{i,i+1} & \cdots & \mathbf{H}_{i,N_{msc}} \\ & & & \mathbf{H}_{i,i+1} & -\mathbf{H}_{i,i+1} & & \\ & & & \vdots & & \ddots & \\ & & & \mathbf{H}_{i,N_{msc}} & & & -\mathbf{H}_{i,N_{msc}} \end{pmatrix} \end{matrix} \quad (3.24b)$$

$$\mathbf{W}_{msc} \stackrel{\text{def}}{=} \sum_{i \in \mathcal{I}} \nabla^2 F_{msc}(\hat{\mathbf{q}}; i) \quad (3.24c)$$

$$= 2 \begin{pmatrix} - \sum_{\substack{j \in \mathcal{I} \\ j \neq 1}} \mathbf{H}_{1,j} & \mathbf{H}_{1,2} & \cdots & \mathbf{H}_{1,N_{msc}} \\ \mathbf{H}_{2,1} & - \sum_{\substack{j \in \mathcal{I} \\ j \neq i}} \mathbf{H}_{2,j} & & \mathbf{H}_{2,N_{msc}} \\ \vdots & & \ddots & \vdots \\ \mathbf{H}_{N_{msc},1} & \mathbf{H}_{N_{msc},2} & \cdots & - \sum_{\substack{j \in \mathcal{I} \\ j \neq i}} \mathbf{H}_{N_{msc},j} \end{pmatrix} \quad (3.24d)$$

ただし、 $\mathbf{H}(\mathbf{q}^{(i)}, \mathbf{q}^{(j)})$  を  $\mathbf{H}_{i,j}$  と略して記す。また、 $d(\mathbf{q}^{(i)}, \mathbf{q}^{(j)}) = -d(\mathbf{q}^{(j)}, \mathbf{q}^{(i)})$ ,  $\mathbf{H}_{i,j} = \mathbf{H}_{j,i}$  を利用した。

解候補分散項  $\sum_{i \in \mathcal{I}} F_{msc}(\hat{\mathbf{q}}; i)$  による二次計画問題の目的関数 (式 (1.5a)) は次式で表される。

$$\sum_{i \in \mathcal{I}} \left\{ F_{msc}(\hat{\mathbf{q}}_k; i) + \nabla F_{msc}(\hat{\mathbf{q}}_k; i)^T \Delta \hat{\mathbf{q}}_k + \frac{1}{2} \Delta \hat{\mathbf{q}}_k^T \nabla^2 F_{msc}(\hat{\mathbf{q}}_k; i) \Delta \hat{\mathbf{q}}_k \right\} \quad (3.25)$$

$$= \sum_{i \in \mathcal{I}} F_{msc}(\hat{\mathbf{q}}_k; i) + \left\{ \sum_{i \in \mathcal{I}} \nabla F_{msc}(\hat{\mathbf{q}}_k; i) \right\}^T \Delta \hat{\mathbf{q}}_k + \frac{1}{2} \Delta \hat{\mathbf{q}}_k^T \left\{ \sum_{i \in \mathcal{I}} \nabla^2 F_{msc}(\hat{\mathbf{q}}_k; i) \right\} \Delta \hat{\mathbf{q}}_k \quad (3.26)$$

$$= \sum_{i \in \mathcal{I}} F_{msc}(\hat{\mathbf{q}}_k; i) + \mathbf{v}_{msc}^T \Delta \hat{\mathbf{q}}_k + \frac{1}{2} \Delta \hat{\mathbf{q}}_k^T \mathbf{W}_{msc} \Delta \hat{\mathbf{q}}_k \quad (3.27)$$

$W_{msc}$  が必ずしも半正定値行列ではないことに注意する必要がある．以下のようにして  $W_{msc}$  に近い正定値行列を計算し用いることで対処する<sup>10</sup>． $W_{msc}$  が次式のように固有値分解されるとする．

$$W_{msc} = V_{msc} D_{msc} V_{msc}^{-1} \quad (3.28)$$

ただし， $D_{msc}$  は固有値を対角成分にもつ対角行列， $V_{msc}$  は固有ベクトルを並べた行列である．このとき  $W_{msc}$  に近い正定値行列  $\tilde{W}_{msc}$  は次式で得られる．

$$\tilde{W}_{msc} = V_{msc} D_{msc}^+ V_{msc}^{-1} \quad (3.29)$$

ただし， $D_{msc}^+$  は  $D_{msc}$  の対角成分のうち，負のものを 0 で置き換えた対角行列である．

式 (3.7) において，解候補を分散させながら，最終的に本来の目的関数を最小にする解を得るために，SQP のイテレーションごとに，解候補分散項のスケール  $k_{msc}$  を次式のように更新することが有効である．

$$k_{msc} \leftarrow \min(\gamma_{msc} k_{msc}, k_{msc-min}) \quad (3.30)$$

$\gamma_{msc}$  は  $0 < \gamma_{msc} < 1$  なるスケール減少率， $k_{msc-min}$  はスケール最小値を表す．

### 3.2.2 複数解候補を用いた逐次二次計画法の実装

#### sqp-msc-optimization

[class]

```

:super      sqp-optimization
:slots      (_num-msc number of multiple solution candidates  $N_{msc}$ )
              (_config-task-list list of configuration-task instance)
              (_dispersion-scale  $k_{msc}$ )
              (_dispersion-scale-min  $k_{msc-min}$ , minimum of  $k_{msc}$ )
              (_dispersion-scale-decrease-ratio  $\gamma_{msc}$ , decrease ration of  $k_{msc}$ )
              (_config-vector-dist2-min minimum squared distance of configuration vector)
              (_dispersion-matrix buffer for  $W_{msc}$ )

```

複数回候補を用いた逐次二次計画法のクラス．

instant-configuration-task クラスや trajectory-configuration-task クラスの instance (以降, configuration-task と呼ぶ) が与えられた時に, configuration-task のタスク関数ノルム二乗  $\|e(q)\|^2$  を最小にするコンフィギュレーション  $q$  を, 複数の解候補を同時に考慮しながら反復計算により求める．

```

:init &rest args &key (num-msc 3) [method]
      (dispersion-scale 0.01)
      (dispersion-scale-min 0.0)
      (dispersion-scale-decrease-ratio 0.5)
      (config-vector-dist2-min 1.000000e-10)
      &allow-other-keys

```

Initialize instance

```

:config-task-list [method]

```

Return list of configuration-task instance

<sup>10</sup> $W_{msc}$  が対称行列であることから, 以下を参考にした．[https://math.stackexchange.com/questions/648809/how-to-find-closest-positive-definite-matrix-of-non-symmetric-matrix#comment1689831\\_649522](https://math.stackexchange.com/questions/648809/how-to-find-closest-positive-definite-matrix-of-non-symmetric-matrix#comment1689831_649522)

**:dispersion-matrix** [method]

式 (3.23d) 参照 .

return  $\mathbf{W}_{msc} \in \mathbb{R}^{N_{msc} \dim(\mathbf{q}) \times N_{msc} \dim(\mathbf{q})}$

**:dispersion-vector** [method]

式 (3.24d) 参照 .

return  $\mathbf{v}_{msc} \in \mathbb{R}^{N_{msc} \dim(\mathbf{q})}$

## 4 動作生成の拡張

### 4.1 マニピュレーションの動作生成

**robot-object-environment** [class]

:super     **robot-environment**  
 :slots     (\_obj  $\mathcal{O}$ )  
              (\_obj-with-root-virtual  $\hat{\mathcal{O}}$ )

ロボットと物体とロボット・環境間の接触のクラス .

以下を合わせた関節・リンク構造に関するメソッドが定義されている .

1. 浮遊ルートリンクのための仮想関節付きのロボットの関節
2. 物体位置姿勢を表す仮想関節
3. 接触位置を定める仮想関節

関節・リンク構造を定めるために、初期化時に以下を与える

**robot**  $\mathcal{R}$  ロボット (cascaded-link クラスのインスタンス) .

**object**  $\mathcal{O}$  物体 (cascaded-link クラスのインスタンス) . 関節をもたないことを前提とする .

**contact-list**  $\{\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_{N_C}\}$  接触 (2d-planar-contact クラスなどのインスタンス) のリスト .

ロボット  $\mathcal{R}$  に、浮遊ルートリンクの変位に対応する仮想関節を付加した仮想関節付きロボット  $\hat{\mathcal{R}}$  を内部で保持する . 同様に、物体  $\mathcal{O}$  に、物体の変位に対応する仮想関節を付加した仮想関節付き物体  $\hat{\mathcal{O}}$  を内部で保持する .

**:init**  $\mathcal{E}_{key}$  (*robot*) [method]

(*object*)

(*contact-list*)

(*root-virtual-mode* :6dof)

(*root-virtual-joint-class-list*)

(*root-virtual-joint-axis-list*)

Initialize instance

**:object**  $\mathcal{E}_{rest}$  *args* [method]

return  $\mathcal{O}$

**:object-with-root-virtual**  $\mathcal{E}_{rest}$  *args* [method]

return  $\hat{\mathcal{O}}$

## instant-manipulation-configuration-task

[class]

```

:super      instant-configuration-task
:slots      (_robot-obj-env robot-object-environment instance)
              (_wrench-obj-vector  $\hat{\mathbf{w}}_{obj}$  [N] [Nm])
              (_num-contact-obj  $N_{cnt-obj} := |\mathcal{T}^{cnt-trg-obj}|$ )
              (_num-act-react  $N_{act-react} := |\mathcal{P}^{act-react}|$ )
              (_dim-wrench-obj  $dim(\hat{\mathbf{w}}_{obj}) = 6N_{cnt-obj}$ )
              (_contact-target-coords-obj-list  $\mathcal{T}^{cnt-trg-obj}$ )
              (_contact-constraint-obj-list list of contact-constraint instance for object)
              (_act-react-pair-list  $\mathcal{P}^{act-react}$ )

```

マニピュレーションにける瞬時コンフィギュレーション  $q^{(l)}$  と瞬時タスク関数  $e^{(l)}(q^{(l)})$  のクラス。マニピュレーション対象の物体の瞬時コンフィギュレーションや瞬時タスク関数を含む。

このクラスの説明で用いる全ての変数は、時間ステップ  $l$  を表す添字をつけて  $x^{(l)}$  と表されるべきだが、このクラス内の説明では省略して  $x$  と表す。また、以降では、説明文やメソッド名で、“瞬時” や “instant” を省略する。

コンフィギュレーション  $q$  の取得・更新、タスク関数  $e(q)$  の取得、タスク関数のヤコビ行列  $\frac{\partial e(q)}{\partial q}$  の取得、コンフィギュレーションの等式・不等式制約  $A, b, C, d$  の取得のためのメソッドが定義されている。コンフィギュレーション・タスク関数を定めるために、instant-configuration-task の設定に加えて、初期化時に以下を与える

- ロボット・物体・環境

robot-object-environment ロボット・物体・環境を表す robot-object-environment クラスのインスタンス

- 物体の接触拘束

contact-target-coords-obj-list  $\mathcal{T}^{cnt-trg-obj}$  物体の接触目標位置姿勢リスト

contact-constraint-obj-list 物体の接触レンチ制約リスト

- 作用・反作用の拘束

act-react-pair-list  $\mathcal{P}^{act-react}$  作用・反作用の関係にあるロボット・物体の接触目標位置姿勢ペアのリスト

コンフィギュレーション  $q$  は以下から構成される。

$\theta \in \mathbb{R}^{N_{var-joint}}$  時変関節角度 [rad] [m]

$\hat{\mathbf{w}} \in \mathbb{R}^{6N_{cnt}}$  ロボットの接触レンチ [N] [Nm]

$\hat{\mathbf{w}}_{obj} \in \mathbb{R}^{6N_{cnt-obj}}$  物体の接触レンチ [N] [Nm]

$\tau \in \mathbb{R}^{N_{drive-joint}}$  関節駆動トルク [Nm] [N]

$\phi \in \mathbb{R}^{N_{invar-joint}}$  時不変関節角度 [rad] [m]

$\hat{\mathbf{w}}$  は次式のように、全接触部位でのワールド座標系での力・モーメントを並べたベクトルである。

$$\hat{\mathbf{w}} = \begin{pmatrix} \mathbf{w}_1^T & \mathbf{w}_2^T & \cdots & \mathbf{w}_{N_{cnt}}^T \end{pmatrix}^T \quad (4.1)$$

$$= \begin{pmatrix} \mathbf{f}_1^T & \mathbf{n}_1^T & \mathbf{f}_2^T & \mathbf{n}_2^T & \cdots & \mathbf{f}_{N_{cnt}}^T & \mathbf{n}_{N_{cnt}}^T \end{pmatrix}^T \quad (4.2)$$

タスク関数  $e(q)$  は以下から構成される.

$$e^{kin}(\mathbf{q}) \in \mathbb{R}^{6N_{kin}} \quad \text{幾何到達拘束} \quad [\text{rad}] \quad [\text{m}]$$

$$e^{eom-trans}(\mathbf{q}) \in \mathbb{R}^3 \quad \text{ロボットの力の釣り合い [N]}$$

$$\mathbf{e}^{com-rot}(\mathbf{q}) \in \mathbb{R}^3 \text{ ロボットのモーメントの釣り合い } [\text{Nm}]$$

$$e^{eom-trans-obj}(\mathbf{q}) \in \mathbb{R}^3$$
 物体の力の釣り合い [N]

$$\mathbf{e}^{com-rot-obj}(\mathbf{q}) \in \mathbb{R}^3 \text{ 物体のモーメントの釣り合い } [\text{Nm}]$$

$$\mathbf{e}^{trq}(\mathbf{q}) \in \mathbb{R}^{N_{drive-joint}} \quad \text{関節トルクの釣り合い} \quad [\text{rad}] \quad [\text{m}]$$

$$e^{posture}(\mathbf{q}) \in \mathbb{R}^{N_{posture-joint}} \quad \text{関節角目標 [rad] [m]}$$

```
:init &rest args &key (contact-target-coords-obj-list) [method]
      (contact-constraint-obj-list)
      (act-react-pair-list)
      &allow-other-keys
```

Initialize instance

```
:robot-obj-env [method]
    return robot-object-environment instance
```

```

:wrench-obj [method]
    return  $\hat{\boldsymbol{w}}_{obj}$ 

```

```

:num-contact-obj [method]
    return  $N_{cnt-obj} := |\mathcal{T}^{cnt-trg-obj}|$ 

```

:dim-variant-config [method]

$$\dim(\mathbf{q}_{var}) := \dim(\boldsymbol{\theta}) + \dim(\hat{\mathbf{w}}) + \dim(\hat{\mathbf{w}}_{obj}) + \dim(\boldsymbol{\tau}) \quad (4.3)$$

$$= N_{var-joint} + 6N_{cnt} + 6N_{cnt-obj} + N_{drive-joint} \quad (4.4)$$

```

return  $\dim(\mathbf{q}_{var})$ 

```

:dim-task [method]

$$\begin{aligned} \dim(\mathbf{e}) \quad &:= \dim(\mathbf{e}^{kin}) + \dim(\mathbf{e}^{com-trans}) + \dim(\mathbf{e}^{com-rot}) + \dim(\mathbf{e}^{com-trans-obj}) \\ &+ \dim(\mathbf{e}^{com-rot-obj}) + \dim(\mathbf{e}^{trq}) + \dim(\mathbf{e}^{posture}) \end{aligned} \quad (4.5)$$

$$= 6N_{kin} + 3 + 3 + 3 + 3 + N_{drive-joint} + N_{posture-joint} \quad (4.6)$$

```

return  $\dim(\mathbf{e})$ 

```

```
:variant-config-vector [method]
```

$$\text{return } \mathbf{q}_{var} := \begin{pmatrix} \theta \\ \hat{w} \\ \hat{w}_{obj} \\ \tau \end{pmatrix}$$

```

:config-vector
[method]

```

$$\text{return } \mathbf{q} := \begin{pmatrix} \mathbf{q}_{var} \\ \mathbf{q}_{invar} \end{pmatrix} = \begin{pmatrix} \boldsymbol{\theta} \\ \hat{\mathbf{w}} \\ \hat{\mathbf{w}}_{obj} \\ \boldsymbol{\tau} \\ \boldsymbol{\phi} \end{pmatrix}$$

**:set-wrench-obj** *wrench-obj-new* *ℰkey* (*relative?* *nil*) [method]  
 Set  $\hat{\mathbf{w}}_{obj}$ .

**:set-variant-config** *variant-config-new* *ℰkey* (*relative?* *nil*) [method]  
 (*apply-to-robot?* *t*)  
 Set  $\mathbf{q}_{var}$ .

**:contact-target-coords-obj-list** [method]

$$T_m^{cnt-trg-obj} = \{\mathbf{p}_m^{cnt-trg-obj}, \mathbf{R}_m^{cnt-trg-obj}\} \quad (m = 1, 2, \dots, N_{cnt-obj}) \quad (4.7)$$

$$\text{return } \mathcal{T}^{cnt-trg-obj} := \{T_1^{cnt-trg-obj}, T_2^{cnt-trg-obj}, \dots, T_{N_{cnt-obj}}^{cnt-trg-obj}\}$$

**:contact-constraint-obj-list** [method]  
 return list of contact-constraint instance for object

**:wrench-obj-list** [method]  
 return  $\{\mathbf{w}_{obj,1}, \mathbf{w}_{obj,2}, \dots, \mathbf{w}_{obj,N_{obj}}\}$

**:force-obj-list** [method]  
 return  $\{\mathbf{f}_{obj,1}, \mathbf{f}_{obj,2}, \dots, \mathbf{f}_{obj,N_{cnt-obj}}\}$

**:moment-obj-list** [method]  
 return  $\{\mathbf{n}_{obj,1}, \mathbf{n}_{obj,2}, \dots, \mathbf{n}_{obj,N_{cnt-obj}}\}$

**:mg-obj-vec** [method]  
 return  $m_{obj}\mathbf{g}$

**:cog-obj** *ℰkey* (*update?* *t*) [method]  
 return  $\mathbf{p}_{Gobj}(\mathbf{q})$

**:eom-trans-obj-task-value** *ℰkey* (*update?* *t*) [method]

$$\mathbf{e}^{eom-trans-obj}(\mathbf{q}) = \mathbf{e}^{eom-trans-obj}(\hat{\mathbf{w}}_{obj}) \quad (4.8)$$

$$= \sum_{m=1}^{N_{cnt-obj}} \mathbf{f}_{obj,m} - m_{obj}\mathbf{g} \quad (4.9)$$

$$\text{return } \mathbf{e}^{eom-trans-obj}(\mathbf{q}) \in \mathbb{R}^3$$

**:eom-rot-obj-task-value** *ℰkey* (*update?* *t*) [method]

$$\mathbf{e}^{eom-rot-obj}(\mathbf{q}) = \mathbf{e}^{eom-rot-obj}(\boldsymbol{\theta}, \hat{\mathbf{w}}_{obj}, \boldsymbol{\phi}) \quad (4.10)$$

$$= \sum_{m=1}^{N_{cnt-obj}} \{(\mathbf{p}_m^{cnt-trg-obj}(\boldsymbol{\theta}, \boldsymbol{\phi}) - \mathbf{p}_{Gobj}(\boldsymbol{\theta}, \boldsymbol{\phi})) \times \mathbf{f}_{obj,m} + \mathbf{n}_{obj,m}\} \quad (4.11)$$



return  $e^{eom-rot-obj}(\mathbf{q}) \in \mathbb{R}^3$

**:task-value**  $\mathcal{E}_{key}$  (update?  $t$ )

[method]

$$\text{return } e(\mathbf{q}) := \begin{pmatrix} e^{kin}(\mathbf{q}) \\ e^{eom-trans}(\mathbf{q}) \\ e^{eom-rot}(\mathbf{q}) \\ e^{eom-trans-obj}(\mathbf{q}) \\ e^{eom-rot-obj}(\mathbf{q}) \\ e^{trq}(\mathbf{q}) \\ e^{posture}(\mathbf{q}) \end{pmatrix} = \begin{pmatrix} e^{kin}(\boldsymbol{\theta}, \phi) \\ e^{eom-trans}(\hat{\mathbf{w}}) \\ e^{eom-rot}(\boldsymbol{\theta}, \hat{\mathbf{w}}, \phi) \\ e^{eom-trans-obj}(\hat{\mathbf{w}}_{obj}) \\ e^{eom-rot-obj}(\boldsymbol{\theta}, \hat{\mathbf{w}}_{obj}, \phi) \\ e^{trq}(\boldsymbol{\theta}, \hat{\mathbf{w}}, \tau, \phi) \\ e^{posture}(\boldsymbol{\theta}) \end{pmatrix}$$

**:eom-trans-obj-task-jacobian-with-wrench-obj**

[method]

$$\frac{\partial e^{eom-trans-obj}}{\partial \hat{\mathbf{w}}_{obj}} = \begin{pmatrix} \frac{\partial e^{eom-trans-obj}}{\partial \mathbf{f}_{obj,1}} & \frac{\partial e^{eom-trans-obj}}{\partial \mathbf{n}_{obj,1}} & \cdots & \frac{\partial e^{eom-trans-obj}}{\partial \mathbf{f}_{obj,N_{cnt-obj}}} & \frac{\partial e^{eom-trans-obj}}{\partial \mathbf{n}_{obj,N_{cnt-obj}}} \end{pmatrix} \quad (4.12)$$

$$= \begin{pmatrix} \mathbf{I}_3 & \mathbf{O}_3 & \cdots & \mathbf{I}_3 & \mathbf{O}_3 \end{pmatrix} \quad (4.13)$$

return  $\frac{\partial e^{eom-trans-obj}}{\partial \hat{\mathbf{w}}_{obj}} \in \mathbb{R}^{3 \times 6N_{cnt-obj}}$

**:eom-rot-obj-task-jacobian-with-theta**

[method]

$$\frac{\partial e^{eom-rot-obj}}{\partial \boldsymbol{\theta}} = \sum_{m=1}^{N_{cnt-obj}} \left\{ -[\mathbf{f}_{obj,m} \times] \left( \mathbf{J}_{\theta,m}^{cnt-trg-obj}(\boldsymbol{\theta}, \phi) - \mathbf{J}_{Gobj\theta}(\boldsymbol{\theta}, \phi) \right) \right\} \quad (4.14)$$

$$= \left[ \left( \sum_{m=1}^{N_{cnt-obj}} \mathbf{f}_{obj,m} \right) \times \right] \mathbf{J}_{Gobj\theta}(\boldsymbol{\theta}, \phi) - \sum_{m=1}^{N_{cnt-obj}} [\mathbf{f}_{obj,m} \times] \mathbf{J}_{\theta,m}^{cnt-trg-obj}(\boldsymbol{\theta}, \phi) \quad (4.15)$$

$\sum_{m=1}^{N_{cnt-obj}} \mathbf{f}_{obj,m} = m_{obj} \mathbf{g}$  つまり, eom-trans-obj-task が成立すると仮定すると次式が成り立つ.

$$\frac{\partial e^{eom-rot-obj}}{\partial \boldsymbol{\theta}} = [m_{obj} \mathbf{g} \times] \mathbf{J}_{Gobj\theta}(\boldsymbol{\theta}, \phi) - \sum_{m=1}^{N_{cnt-obj}} [\mathbf{f}_{obj,m} \times] \mathbf{J}_{\theta,m}^{cnt-trg-obj}(\boldsymbol{\theta}, \phi) \quad (4.16)$$

return  $\frac{\partial e^{eom-rot-obj}}{\partial \boldsymbol{\theta}} \in \mathbb{R}^{3 \times N_{var-joint}}$

**:eom-rot-obj-task-jacobian-with-wrench-obj**

[method]

$$\frac{\partial e^{eom-rot-obj}}{\partial \hat{\mathbf{w}}_{obj}} = \begin{pmatrix} \frac{\partial e^{eom-rot-obj}}{\partial \mathbf{f}_{obj,1}} & \frac{\partial e^{eom-rot-obj}}{\partial \mathbf{n}_{obj,1}} & \cdots & \frac{\partial e^{eom-rot-obj}}{\partial \mathbf{f}_{obj,N_{cnt-obj}}} & \frac{\partial e^{eom-rot-obj}}{\partial \mathbf{n}_{obj,N_{cnt-obj}}} \end{pmatrix} \quad (4.17)$$

$$\frac{\partial e^{eom-rot-obj}}{\partial \mathbf{f}_{obj,m}} = [(\mathbf{p}_m^{cnt-trg-obj}(\boldsymbol{\theta}, \phi) - \mathbf{p}_{Gobj}(\boldsymbol{\theta}, \phi)) \times] \quad (m = 1, 2, \dots, N_{cnt-obj}) \quad (4.18)$$

$$\frac{\partial e^{eom-rot-obj}}{\partial \mathbf{n}_{obj,m}} = \mathbf{I}_3 \quad (m = 1, 2, \dots, N_{cnt-obj}) \quad (4.19)$$

return  $\frac{\partial e^{eom-rot-obj}}{\partial \hat{\mathbf{w}}_{obj}} \in \mathbb{R}^{3 \times 6N_{cnt-obj}}$

**:eom-rot-obj-task-jacobian-with-phi**

[method]

$$\frac{\partial \mathbf{e}^{eom-rot-obj}}{\partial \phi} = \sum_{m=1}^{N_{cnt-obj}} \left\{ -[\mathbf{f}_{obj,m} \times] \left( \mathbf{J}_{\phi,m}^{cnt-trg-obj}(\boldsymbol{\theta}, \phi) - \mathbf{J}_{Gobj\phi}(\boldsymbol{\theta}, \phi) \right) \right\} \quad (4.20)$$

$$= \left[ \left( \sum_{m=1}^{N_{cnt-obj}} \mathbf{f}_{obj,m} \right) \times \right] \mathbf{J}_{Gobj\phi}(\boldsymbol{\theta}, \phi) - \sum_{m=1}^{N_{cnt-obj}} [\mathbf{f}_{obj,m} \times] \mathbf{J}_{\phi,m}^{cnt-trg-obj}(\boldsymbol{\theta}, \phi) \quad (4.21)$$

$\sum_{m=1}^{N_{cnt-obj}} \mathbf{f}_{obj,m} = m_{obj} \mathbf{g}$  つまり, eom-trans-obj-task が成立すると仮定すると次式が成り立つ .

$$\frac{\partial \mathbf{e}^{eom-rot-obj}}{\partial \phi} = [m_{obj} \mathbf{g} \times] \mathbf{J}_{Gobj\phi}(\boldsymbol{\theta}, \phi) - \sum_{m=1}^{N_{cnt-obj}} [\mathbf{f}_{obj,m} \times] \mathbf{J}_{\phi,m}^{cnt-trg-obj}(\boldsymbol{\theta}, \phi) \quad (4.22)$$

return  $\frac{\partial \mathbf{e}^{eom-rot-obj}}{\partial \phi} \in \mathbb{R}^{3 \times N_{invar-joint}}$

**:variant-task-jacobian**

[method]

$$\frac{\partial \mathbf{e}}{\partial \mathbf{q}_{var}} = \begin{matrix} & N_{var-joint} & 6N_{cnt} & 6N_{cnt-obj} & N_{drive-joint} \\ \begin{matrix} 6N_{kin} \\ 3 \\ 3 \\ 3 \\ 3 \\ N_{drive-joint} \\ N_{posture-joint} \end{matrix} & \begin{pmatrix} \frac{\partial \mathbf{e}^{kin}}{\partial \boldsymbol{\theta}} & & & \\ & \frac{\partial \mathbf{e}^{eom-trans}}{\partial \hat{\mathbf{w}}} & & \\ & \frac{\partial \mathbf{e}^{eom-rot}}{\partial \boldsymbol{\theta}} & \frac{\partial \hat{\mathbf{w}}}{\partial \mathbf{e}^{eom-rot}} & \\ & & \frac{\partial \mathbf{e}^{eom-trans-obj}}{\partial \hat{\mathbf{w}}_{obj}} & \\ & \frac{\partial \mathbf{e}^{eom-rot-obj}}{\partial \boldsymbol{\theta}} & \frac{\partial \hat{\mathbf{w}}_{obj}}{\partial \mathbf{e}^{eom-rot-obj}} & \\ & \frac{\partial \mathbf{e}^{trq}}{\partial \boldsymbol{\theta}} & \frac{\partial \mathbf{e}^{trq}}{\partial \hat{\mathbf{w}}} & \\ & \frac{\partial \mathbf{e}^{posture}}{\partial \boldsymbol{\theta}} & & \frac{\partial \mathbf{e}^{trq}}{\partial \boldsymbol{\tau}} \end{pmatrix} \end{matrix} \quad (4.23)$$

return  $\frac{\partial \mathbf{e}}{\partial \mathbf{q}_{var}} \in \mathbb{R}^{(6N_{kin}+3+3+3+3+N_{drive-joint}+N_{posture-joint}) \times (N_{var-joint}+6N_{cnt}+6N_{cnt-obj}+N_{drive-joint})}$

**:invariant-task-jacobian**

[method]

$$\frac{\partial \mathbf{e}}{\partial \mathbf{q}_{invar}} = \begin{matrix} & N_{invar-joint} \\ \begin{matrix} 6N_{kin} \\ 3 \\ 3 \\ 3 \\ 3 \\ N_{drive-joint} \\ N_{posture-joint} \end{matrix} & \begin{pmatrix} \frac{\partial \mathbf{e}^{kin}}{\partial \phi} \\ & \frac{\partial \mathbf{e}^{eom-rot}}{\partial \phi} \\ & & \frac{\partial \mathbf{e}^{eom-rot-obj}}{\partial \phi} \\ & & \frac{\partial \phi}{\partial \mathbf{e}^{trq}} \\ & & \frac{\partial \phi}{\partial \phi} \end{pmatrix} \end{matrix} \quad (4.24)$$

return  $\frac{\partial \mathbf{e}}{\partial \mathbf{q}_{invar}} \in \mathbb{R}^{(6N_{kin}+3+3+3+3+N_{drive-joint}+N_{posture-joint}) \times N_{invar-joint}}$

**:task-jacobian**

[method]

$$\frac{\partial \mathbf{e}}{\partial \mathbf{q}} = \begin{pmatrix} \frac{\partial \mathbf{e}}{\partial \mathbf{q}_{\text{var}}} & \frac{\partial \mathbf{e}}{\partial \mathbf{q}_{\text{invar}}} \end{pmatrix} \quad (4.25)$$

$$= \begin{matrix} 6N_{\text{kin}} \\ 3 \\ 3 \\ 3 \\ 3 \\ N_{\text{drive-joint}} \\ N_{\text{posture-joint}} \end{matrix} \begin{pmatrix} N_{\text{var-joint}} & 6N_{\text{cnt}} & 6N_{\text{cnt-obj}} & N_{\text{drive-joint}} & N_{\text{invar-joint}} \\ \frac{\partial \mathbf{e}^{\text{kin}}}{\partial \boldsymbol{\theta}} & & & & \frac{\partial \mathbf{e}^{\text{kin}}}{\partial \boldsymbol{\phi}} \\ & \frac{\partial \mathbf{e}^{\text{com-trans}}}{\partial \hat{\mathbf{w}}} & & & \\ \frac{\partial \mathbf{e}^{\text{com-rot}}}{\partial \boldsymbol{\theta}} & \frac{\partial \mathbf{e}^{\text{com-rot}}}{\partial \hat{\mathbf{w}}} & & & \frac{\partial \mathbf{e}^{\text{com-rot}}}{\partial \boldsymbol{\phi}} \\ & & \frac{\partial \mathbf{e}^{\text{com-trans-obj}}}{\partial \hat{\mathbf{w}}_{\text{obj}}} & & \\ \frac{\partial \mathbf{e}^{\text{com-rot-obj}}}{\partial \boldsymbol{\theta}} & & \frac{\partial \mathbf{e}^{\text{com-rot-obj}}}{\partial \hat{\mathbf{w}}_{\text{obj}}} & & \frac{\partial \mathbf{e}^{\text{com-rot-obj}}}{\partial \boldsymbol{\phi}} \\ & \frac{\partial \mathbf{e}^{\text{trq}}}{\partial \boldsymbol{\theta}} & \frac{\partial \mathbf{e}^{\text{trq}}}{\partial \hat{\mathbf{w}}} & \frac{\partial \mathbf{e}^{\text{trq}}}{\partial \boldsymbol{\tau}} & \frac{\partial \mathbf{e}^{\text{trq}}}{\partial \boldsymbol{\phi}} \\ \frac{\partial \mathbf{e}^{\text{posture}}}{\partial \boldsymbol{\theta}} & & & & \end{pmatrix} \quad (4.26)$$

$$\text{return } \frac{\partial \mathbf{e}}{\partial \mathbf{q}} \in \mathbb{R}^{(6N_{\text{kin}}+3+3+3+3+N_{\text{drive-joint}}+N_{\text{posture-joint}}) \times (N_{\text{var-joint}}+6N_{\text{cnt}}+6N_{\text{cnt-obj}}+N_{\text{drive-joint}}+N_{\text{invar-joint}})}$$

**:wrench-obj-inequality-constraint-matrix** *ℰkey (update? t)* [method]

物体の接触レンチ  $\mathbf{w}_{\text{obj}} \in \mathbb{R}^6$  が満たすべき制約（非負制約，摩擦制約，圧力中心制約）が次式のように表されるとする．

$$\mathbf{C}_{\mathbf{w}_{\text{obj}}} \mathbf{w}_{\text{obj}} \geq \mathbf{d}_{\mathbf{w}_{\text{obj}}} \quad (4.27)$$

$N_{\text{cnt-obj}}$  箇所の接触部位の接触レンチを並べたベクトル  $\hat{\mathbf{w}}_{\text{obj}}$  の不等式制約は次式で表される．

$$\mathbf{C}_{\mathbf{w}_{\text{obj}},m} (\mathbf{w}_{\text{obj},m} + \Delta \mathbf{w}_{\text{obj},m}) \geq \mathbf{d}_{\mathbf{w}_{\text{obj}},m} \quad (m = 1, 2, \dots, N_{\text{cnt-obj}}) \quad (4.28)$$

$$\Leftrightarrow \mathbf{C}_{\mathbf{w}_{\text{obj}},m} \Delta \mathbf{w}_{\text{obj},m} \geq \mathbf{d}_{\mathbf{w}_{\text{obj}},m} - \mathbf{C}_{\mathbf{w}_{\text{obj}},m} \mathbf{w}_{\text{obj},m} \quad (m = 1, 2, \dots, N_{\text{cnt-obj}}) \quad (4.29)$$

$$\Leftrightarrow \begin{pmatrix} \mathbf{C}_{\mathbf{w}_{\text{obj}},1} & & & \\ & \mathbf{C}_{\mathbf{w}_{\text{obj}},2} & & \\ & & \ddots & \\ & & & \mathbf{C}_{\mathbf{w}_{\text{obj}},N_{\text{cnt-obj}}} \end{pmatrix} \begin{pmatrix} \Delta \mathbf{w}_{\text{obj},1} \\ \Delta \mathbf{w}_{\text{obj},2} \\ \vdots \\ \Delta \mathbf{w}_{\text{obj},N_{\text{cnt-obj}}} \end{pmatrix} \geq \begin{pmatrix} \mathbf{d}_{\mathbf{w}_{\text{obj}},1} - \mathbf{C}_{\mathbf{w}_{\text{obj}},1} \mathbf{w}_{\text{obj},1} \\ \mathbf{d}_{\mathbf{w}_{\text{obj}},2} - \mathbf{C}_{\mathbf{w}_{\text{obj}},2} \mathbf{w}_{\text{obj},2} \\ \vdots \\ \mathbf{d}_{\mathbf{w}_{\text{obj}},N_{\text{cnt-obj}}} - \mathbf{C}_{\mathbf{w}_{\text{obj}},N_{\text{cnt-obj}}} \mathbf{w}_{\text{obj},N_{\text{cnt-obj}}} \end{pmatrix} \quad (4.30)$$

$$\Leftrightarrow \mathbf{C}_{\hat{\mathbf{w}}_{\text{obj}}} \Delta \hat{\mathbf{w}}_{\text{obj}} \geq \mathbf{d}_{\hat{\mathbf{w}}_{\text{obj}}} \quad (4.31)$$

$$\text{return } \mathbf{C}_{\hat{\mathbf{w}}_{\text{obj}}} := \begin{pmatrix} \mathbf{C}_{\mathbf{w}_{\text{obj}},1} & & & \\ & \mathbf{C}_{\mathbf{w}_{\text{obj}},2} & & \\ & & \ddots & \\ & & & \mathbf{C}_{\mathbf{w}_{\text{obj}},N_{\text{cnt-obj}}} \end{pmatrix} \in \mathbb{R}^{N_{\text{wrench-obj-ineq}} \times 6N_{\text{cnt-obj}}}$$

**:wrench-obj-inequality-constraint-vector** *ℰkey (update? t)* [method]

$$\text{return } \mathbf{d}_{\hat{\mathbf{w}}_{\text{obj}}} := \begin{pmatrix} \mathbf{d}_{\mathbf{w}_{\text{obj}},1} - \mathbf{C}_{\mathbf{w}_{\text{obj}},1} \mathbf{w}_{\text{obj},1} \\ \mathbf{d}_{\mathbf{w}_{\text{obj}},2} - \mathbf{C}_{\mathbf{w}_{\text{obj}},2} \mathbf{w}_{\text{obj},2} \\ \vdots \\ \mathbf{d}_{\mathbf{w}_{\text{obj}},N_{\text{cnt-obj}}} - \mathbf{C}_{\mathbf{w}_{\text{obj}},N_{\text{cnt-obj}}} \mathbf{w}_{\text{obj},N_{\text{cnt-obj}}} \end{pmatrix} \in \mathbb{R}^{N_{\text{wrench-obj-ineq}}}$$

**:variant-config-inequality-constraint-matrix** *ℰkey (update? nil)* [method]

$$\begin{cases} \mathbf{C}_\theta \Delta \boldsymbol{\theta} \geq \mathbf{d}_\theta \\ \mathbf{C}_{\hat{\mathbf{w}}} \Delta \hat{\mathbf{w}} \geq \mathbf{d}_{\hat{\mathbf{w}}} \\ \mathbf{C}_{\hat{\mathbf{w}}_{obj}} \Delta \hat{\mathbf{w}}_{obj} \geq \mathbf{d}_{\hat{\mathbf{w}}_{obj}} \\ \mathbf{C}_\tau \Delta \boldsymbol{\tau} \geq \mathbf{d}_\tau \end{cases} \quad (4.32)$$

$$\Leftrightarrow \begin{pmatrix} \mathbf{C}_\theta & & & \\ & \mathbf{C}_{\hat{\mathbf{w}}} & & \\ & & \mathbf{C}_{\hat{\mathbf{w}}_{obj}} & \\ & & & \mathbf{C}_\tau \end{pmatrix} \begin{pmatrix} \Delta \boldsymbol{\theta} \\ \Delta \hat{\mathbf{w}} \\ \Delta \hat{\mathbf{w}}_{obj} \\ \Delta \boldsymbol{\tau} \end{pmatrix} \geq \begin{pmatrix} \mathbf{d}_\theta \\ \mathbf{d}_{\hat{\mathbf{w}}} \\ \mathbf{d}_{\hat{\mathbf{w}}_{obj}} \\ \mathbf{d}_\tau \end{pmatrix} \quad (4.33)$$

$$\Leftrightarrow \mathbf{C}_{var} \Delta \mathbf{q}_{var} \geq \mathbf{d}_{var} \quad (4.34)$$

$$\text{return } \mathbf{C}_{var} := \begin{pmatrix} \mathbf{C}_\theta & & & \\ & \mathbf{C}_{\hat{\mathbf{w}}} & & \\ & & \mathbf{C}_{\hat{\mathbf{w}}_{obj}} & \\ & & & \mathbf{C}_\tau \end{pmatrix} \in \mathbb{R}^{N_{var-ineq} \times \dim(\mathbf{q}_{var})}$$

**:variant-config-inequality-constraint-vector** *ℓkey (update? t)* [method]

$$\text{return } \mathbf{d}_{var} := \begin{pmatrix} \mathbf{d}_\theta \\ \mathbf{d}_{\hat{\mathbf{w}}} \\ \mathbf{d}_{\hat{\mathbf{w}}_{obj}} \\ \mathbf{d}_\tau \end{pmatrix} \in \mathbb{R}^{N_{var-ineq}}$$

**:act-react-equality-constraint-matrix** *ℓkey (update? nil)* [method]

ロボット・物体間の接触レンチに関する作用・反作用の法則は次式のように表される．

$$\hat{\mathbf{w}}_{i(m)} + \hat{\mathbf{w}}_{obj,j(m)} = \mathbf{0} \quad (m = 1, 2, \dots, N_{act-react}) \quad (4.35)$$

$$\Leftrightarrow \mathbf{A}_{act-react,robot,m} \hat{\mathbf{w}} + \mathbf{A}_{act-react,obj,m} \hat{\mathbf{w}}_{obj} = \mathbf{0} \quad (m = 1, 2, \dots, N_{act-react}) \quad (4.36)$$

$$\text{where } \mathbf{A}_{act-react,robot,m} = \begin{pmatrix} \mathbf{O}_6 & \mathbf{O}_6 & \cdots & \mathbf{I}_6 & \cdots & \mathbf{O}_6 & \mathbf{O}_6 \end{pmatrix} \in \mathbb{R}^{6 \times 6N_{cnt}} \quad (4.37)$$

$$\mathbf{A}_{act-react,obj,m} = \begin{pmatrix} \mathbf{O}_6 & \mathbf{O}_6 & \cdots & \mathbf{I}_6 & \cdots & \mathbf{O}_6 & \mathbf{O}_6 \end{pmatrix} \in \mathbb{R}^{6 \times 6N_{cnt-obj}} \quad (4.38)$$

$$\Leftrightarrow \mathbf{A}_{act-react,robot} \hat{\mathbf{w}} + \mathbf{A}_{act-react,obj} \hat{\mathbf{w}}_{obj} = \mathbf{0} \quad (4.39)$$

$$\text{where } \mathbf{A}_{act-react,robot} = \begin{pmatrix} \mathbf{A}_{act-react,robot,1} \\ \vdots \\ \mathbf{A}_{act-react,robot,N_{act-react}} \end{pmatrix} \in \mathbb{R}^{6N_{act-react} \times 6N_{cnt}} \quad (4.40)$$

$$\mathbf{A}_{act-react,obj} = \begin{pmatrix} \mathbf{A}_{act-react,obj,1} \\ \vdots \\ \mathbf{A}_{act-react,obj,N_{act-react}} \end{pmatrix} \in \mathbb{R}^{6N_{act-react} \times 6N_{cnt-obj}} \quad (4.41)$$

$$\Leftrightarrow \mathbf{A}_{act-react} \begin{pmatrix} \hat{\mathbf{w}} \\ \hat{\mathbf{w}}_{obj} \end{pmatrix} = \mathbf{0} \in \mathbb{R}^{6N_{act-react}} \quad (4.42)$$

$$\text{where } \mathbf{A}_{act-react} = \begin{pmatrix} \mathbf{A}_{act-react,robot} & \mathbf{A}_{act-react,obj} \end{pmatrix} \in \mathbb{R}^{6N_{act-react} \times (6N_{cnt} + 6N_{cnt-obj})} \quad (4.43)$$

$$\Leftrightarrow \mathbf{A}_{act-react} \begin{pmatrix} \hat{\mathbf{w}} + \Delta \hat{\mathbf{w}} \\ \hat{\mathbf{w}}_{obj} + \Delta \hat{\mathbf{w}}_{obj} \end{pmatrix} = \mathbf{0} \quad (4.44)$$

$$\Leftrightarrow \mathbf{A}_{act-react} \begin{pmatrix} \Delta \hat{\mathbf{w}} \\ \Delta \hat{\mathbf{w}}_{obj} \end{pmatrix} = \mathbf{b}_{act-react} \quad (4.45)$$

$$\text{where } \mathbf{b}_{act-react} = -\mathbf{A}_{act-react} \begin{pmatrix} \hat{\mathbf{w}} \\ \hat{\mathbf{w}}_{obj} \end{pmatrix} \quad (4.46)$$

$i(m), j(m)$  は作用・反作用の関係にある接触レンチの  $m$  番目の対におけるロボット，物体の接触レンチのインデックスである．

return  $\mathbf{A}_{act-react} \in \mathbb{R}^{6N_{act-react} \times (6N_{cnt} + 6N_{cnt-obj})}$

**:act-react-equality-constraint-vector**  $\mathcal{E}key$  (update?  $t$ ) [method]

return  $\mathbf{b}_{act-react} \in \mathbb{R}^{6N_{act-react}}$

**:variant-config-equality-constraint-matrix**  $\mathcal{E}key$  (update?  $nil$ ) [method]

$$\mathbf{A}_{act-react} \begin{pmatrix} \Delta \hat{\mathbf{w}} \\ \Delta \hat{\mathbf{w}}_{obj} \end{pmatrix} = \mathbf{b}_{act-react} \quad (4.47)$$

$$\Leftrightarrow \begin{pmatrix} \mathbf{O} & \mathbf{A}_{act-react} & \mathbf{O} \end{pmatrix} \begin{pmatrix} \Delta \boldsymbol{\theta} \\ \Delta \hat{\mathbf{w}} \\ \Delta \hat{\mathbf{w}}_{obj} \\ \Delta \boldsymbol{\tau} \end{pmatrix} = \mathbf{b}_{act-react} \quad (4.48)$$

$$\Leftrightarrow \mathbf{A}_{var} \Delta \mathbf{q}_{var} = \mathbf{b}_{var} \quad (4.49)$$

return  $\mathbf{A}_{var} := \begin{pmatrix} \mathbf{O} & \mathbf{A}_{act-react} & \mathbf{O} \end{pmatrix} \in \mathbb{R}^{6N_{act-react} \times \dim(\mathbf{q}_{var})}$

**:variant-config-equality-constraint-vector**  $\mathcal{E}key$  (*update?*  $t$ ) [method]

return  $\mathbf{b}_{var} := \mathbf{b}_{act-react} \in \mathbb{R}^{6N_{act-react}}$

**:invariant-config-equality-constraint-matrix**  $\mathcal{E}key$  (*update?*  $nil$ ) [method]

return  $\mathbf{A}_{invar} \in \mathbb{R}^{0 \times \dim(\mathbf{q}_{invar})}$  (no equality constraint)

**:invariant-config-equality-constraint-vector**  $\mathcal{E}key$  (*update?*  $t$ ) [method]

return  $\mathbf{b}_{invar} \in \mathbb{R}^0$  (no equality constraint)

**:config-equality-constraint-matrix**  $\mathcal{E}key$  (*update?*  $nil$ ) [method]

$$\mathbf{A}_{var} \Delta \mathbf{q}_{var} = \mathbf{b}_{var} \quad (4.50)$$

$$\Leftrightarrow \begin{pmatrix} \mathbf{A}_{var} & \mathbf{O} \end{pmatrix} \begin{pmatrix} \Delta \mathbf{q}_{var} \\ \Delta \mathbf{q}_{invar} \end{pmatrix} = \mathbf{b}_{var} \quad (4.51)$$

$$\Leftrightarrow \mathbf{A} \Delta \mathbf{q} = \mathbf{b} \quad (4.52)$$

return  $\mathbf{A} := \begin{pmatrix} \mathbf{A}_{var} & \mathbf{O} \end{pmatrix} \in \mathbb{R}^{N_{eq} \times \dim(\mathbf{q})}$

**:config-equality-constraint-vector**  $\mathcal{E}key$  (*update?*  $t$ ) [method]

return  $\mathbf{b} := \mathbf{b}_{var} \in \mathbb{R}^{N_{eq}}$

**:torque-regular-matrix**  $\mathcal{E}key$  (*update?*  $nil$ ) [method]

(*only-variant?*  $nil$ )

二次形式の正則化項として次式を考える .

$$F_{tau}(\mathbf{q}) = \left\| \frac{\boldsymbol{\tau}}{\boldsymbol{\tau}_{max}} \right\|^2 \quad (\text{ベクトルの要素ごとの割り算を表す}) \quad (4.53)$$

$$= \boldsymbol{\tau}^T \bar{\mathbf{W}}_{trq} \boldsymbol{\tau} \quad (4.54)$$

ここで ,

$$\bar{\mathbf{W}}_{trq} := \begin{pmatrix} \frac{1}{\tau_{max,1}^2} & & & \\ & \frac{1}{\tau_{max,2}^2} & & \\ & & \ddots & \\ & & & \frac{1}{\tau_{max,N_{drive-joint}}^2} \end{pmatrix} \in \mathbb{R}^{\dim(\boldsymbol{\tau}) \times \dim(\boldsymbol{\tau})} \quad (4.55)$$

*only-variant?* is true:

$$\mathbf{W}_{trq} := \begin{matrix} \dim(\boldsymbol{\theta}) \\ \dim(\hat{\mathbf{w}}) \\ \dim(\hat{\mathbf{w}}_{obj}) \\ \dim(\boldsymbol{\tau}) \end{matrix} \begin{pmatrix} \dim(\boldsymbol{\theta}) & \dim(\hat{\mathbf{w}}) & \dim(\hat{\mathbf{w}}_{obj}) & \dim(\boldsymbol{\tau}) \\ & & & \\ & & & \\ & & & \bar{\mathbf{W}}_{trq} \end{pmatrix} \in \mathbb{R}^{\dim(\mathbf{q}_{var}) \times \dim(\mathbf{q}_{var})} \quad (4.56)$$

otherwise:

$$\mathbf{W}_{trq} := \begin{matrix} \dim(\boldsymbol{\theta}) \\ \dim(\hat{\mathbf{w}}) \\ \dim(\hat{\mathbf{w}}_{obj}) \\ \dim(\boldsymbol{\tau}) \\ \dim(\boldsymbol{\phi}) \end{matrix} \begin{pmatrix} \dim(\boldsymbol{\theta}) & \dim(\hat{\mathbf{w}}) & \dim(\hat{\mathbf{w}}_{obj}) & \dim(\boldsymbol{\tau}) & \dim(\boldsymbol{\phi}) \\ & & & & \\ & & & & \\ & & & & \\ & & & \bar{\mathbf{W}}_{trq} & \end{pmatrix} \in \mathbb{R}^{\dim(\mathbf{q}) \times \dim(\mathbf{q})} \quad (4.57)$$

```

return  $\mathbf{W}_{trq}$ 

```

```
:torque-regular-vector key (update? t) [method]
(only-variant? nil)
```

$$\bar{\mathbf{v}}_{trq} := \bar{\mathbf{W}}_{trq} \boldsymbol{\tau} \quad (4.58)$$

$$= \begin{pmatrix} \frac{\tau_1}{\tau_{\max,1}^2} \\ \frac{\tau_2}{\tau_{\max,2}^2} \\ \vdots \\ \frac{\tau_{\dim(\mathbf{T})}}{\tau_{\max,\dim(\mathbf{T})}^2} \end{pmatrix} \in \mathbb{R}^{\dim(\mathbf{T})} \quad (4.59)$$

*only-variant?* is true:

$$\mathbf{v}_{trq} := \begin{pmatrix} 1 \\ \dim(\boldsymbol{\theta}) \\ \dim(\hat{\mathbf{w}}) \\ \dim(\hat{\mathbf{w}}_{obj}) \\ \dim(\boldsymbol{\tau}) \\ \bar{\mathbf{v}}_{trq} \end{pmatrix} \in \mathbb{R}^{\dim(\mathbf{q}_{var})} \quad (4.60)$$

*otherwise:*

$$\mathbf{v}_{trq} := \begin{pmatrix} 1 \\ \dim(\boldsymbol{\theta}) \\ \dim(\hat{\mathbf{w}}) \\ \dim(\hat{\mathbf{w}}_{obj}) \\ \dim(\boldsymbol{\tau}) \\ \dim(\phi) \end{pmatrix} \in \mathbb{R}^{\dim(\mathbf{q})} \quad (4.61)$$

$$\text{return } \mathbf{v}_{trq}$$

```
:collision-inequality-constraint-matrix key (update? nil) [method]
```

$$\mathbf{C}_{col} := N_{col} \begin{pmatrix} \dim(\boldsymbol{\theta}) & \dim(\hat{\mathbf{w}}) & \dim(\hat{\mathbf{w}}_{obj}) & \dim(\boldsymbol{\tau}) & \dim(\boldsymbol{\phi}) \\ \mathbf{C}_{col,\theta} & \mathbf{O} & \mathbf{O} & \mathbf{O} & \mathbf{C}_{col,\phi} \end{pmatrix} \quad (4.62)$$

return  $\mathbf{C}_{col} \in \mathbb{R}^{N_{col} \times \dim(\mathbf{q})}$

<b>:update-viewer</b>	[method]
Update viewer.	

<b>:print-status</b>	[method]
Print status.	

## 4.2 B スプラインを用いた関節軌道生成

### 4.2.1 B スプラインを用いた関節軌道生成の理論

一般の B スプライン基底関数の定義

B スプライン基底関数は以下で定義される .

$$b_{i,0}(t) \stackrel{\text{def}}{=} \begin{cases} 1 & \text{if } t_i \leq t < t_{i+1} \\ 0 & \text{otherwise} \end{cases} \quad (4.63)$$

$$b_{i,n}(t) \stackrel{\text{def}}{=} \frac{t - t_i}{t_{i+n} - t_i} b_{i,n-1}(t) + \frac{t_{i+n+1} - t}{t_{i+n+1} - t_{i+1}} b_{i+1,n-1}(t) \quad (4.64)$$

$t_i$  はノットと呼ばれる .

使用区間を指定してノットを一様とする場合の B スプライン基底関数

$t_s, t_f$  を B スプラインの使用区間の初期 , 終端時刻とする .

$n < m$  とする .

$$t_n = t_s \quad (4.65)$$

$$t_m = t_f \quad (4.66)$$

とする .  $t_i$  ( $0 \leq i \leq n+m$ ) が等間隔に並ぶとすると ,

$$t_i = \frac{i-n}{m-n}(t_f - t_s) + t_s \quad (4.67)$$

$$= hi + \frac{mt_s - nt_f}{m-n} \quad (4.68)$$

ただし ,

$$h \stackrel{\text{def}}{=} \frac{t_f - t_s}{m-n} \quad (4.69)$$

式 (4.68) を式 (4.63) , 式 (4.64) に代入すると , B スプライン基底関数は次式で得られる .

$$b_{i,0}(t) = \begin{cases} 1 & \text{if } t_i \leq t < t_{i+1} \\ 0 & \text{otherwise} \end{cases} \quad (4.70)$$

$$b_{i,n}(t) = \frac{(t - t_i)b_{i,n-1}(t) + (t_{i+n+1} - t)b_{i+1,n-1}(t)}{nh} \quad (4.71)$$

以降では ,  $n$  を B スプラインの次数 ,  $m$  を制御点の個数と呼ぶ .

### B スプラインの凸包性

式 (4.70) , 式 (4.71) で定義される B スプライン基底関数  $b_{i,n}(t)$  は次式のように凸包性を持つ .

$$\sum_{i=0}^{m-1} b_{i,n}(t) = 1 \quad (t_s \leq t \leq t_f) \quad (4.72)$$

$$0 \leq b_{i,n}(t) \leq 1 \quad (i = 0, 1, \dots, m-1, t_s \leq t \leq t_f) \quad (4.73)$$



## B スプラインの微分

B スプライン基底関数の微分に関して次式が成り立つ<sup>11</sup> .

$$\dot{\mathbf{b}}_n(t) = \frac{d\mathbf{b}_n(t)}{dt} = \mathbf{D}\mathbf{b}_{n-1}(t) \quad (4.74)$$

ただし ,

$$\mathbf{b}_n(t) \stackrel{\text{def}}{=} \begin{pmatrix} b_{0,n}(t) \\ b_{1,n}(t) \\ \vdots \\ b_{m-1,n}(t) \end{pmatrix} \in \mathbb{R}^m \quad (4.75)$$

$$\mathbf{D} \stackrel{\text{def}}{=} \frac{1}{h} \begin{pmatrix} 1 & -1 & & & \mathbf{O} \\ & 1 & -1 & & \\ & & \ddots & \ddots & \\ & & & \ddots & -1 \\ \mathbf{O} & & & & 1 \end{pmatrix} \in \mathbb{R}^{m \times m} \quad (4.76)$$

したがって ,  $k$  階微分に関して次式が成り立つ .

$$\mathbf{b}_n^{(k)}(t) = \frac{d^{(k)}\mathbf{b}_n(t)}{dt^{(k)}} = \mathbf{D}^k \mathbf{b}_{n-k}(t) \quad (4.77)$$

## B スプラインによる関節角軌道の表現

$j$  番目の関節角軌道  $\theta_j(t)$  を次式で表す .

$$\theta_j(t) \stackrel{\text{def}}{=} \sum_{i=0}^{m-1} p_{j,i} b_{i,n}(t) = \mathbf{p}_j^T \mathbf{b}_n(t) \in \mathbb{R} \quad (t_s \leq t \leq t_f) \quad (4.78)$$

ただし ,

$$\mathbf{p}_j = \begin{pmatrix} p_{j,0} \\ p_{j,1} \\ \vdots \\ p_{j,m-1} \end{pmatrix} \in \mathbb{R}^m, \quad \mathbf{b}_n(t) = \begin{pmatrix} b_{0,n}(t) \\ b_{1,n}(t) \\ \vdots \\ b_{m-1,n}(t) \end{pmatrix} \in \mathbb{R}^m \quad (4.79)$$

以降では ,  $\mathbf{p}_j$  を制御点 ,  $\mathbf{b}_n(t)$  を基底関数と呼ぶ . 制御点  $\mathbf{p}_j$  を決定すると関節角軌道が定まる . 制御点  $\mathbf{p}_j$  を動作計画の設計変数とする .

$j = 1, 2, \dots, N_{\text{joint}}$  番目の関節角軌道を並べたベクトル関数は ,

$$\boldsymbol{\theta}(t) \stackrel{\text{def}}{=} \begin{pmatrix} \theta_1(t) \\ \theta_2(t) \\ \vdots \\ \theta_{N_{\text{joint}}}(t) \end{pmatrix} = \begin{pmatrix} \mathbf{p}_1^T \mathbf{b}_n(t) \\ \mathbf{p}_2^T \mathbf{b}_n(t) \\ \vdots \\ \mathbf{p}_{N_{\text{joint}}}^T \mathbf{b}_n(t) \end{pmatrix} = \begin{pmatrix} \mathbf{p}_1^T \\ \mathbf{p}_2^T \\ \vdots \\ \mathbf{p}_{N_{\text{joint}}}^T \end{pmatrix} \mathbf{b}_n(t) = \mathbf{P} \mathbf{b}_n(t) \in \mathbb{R}^{N_{\text{joint}}} \quad (4.80)$$

ただし ,

$$\mathbf{P} \stackrel{\text{def}}{=} \begin{pmatrix} \mathbf{p}_1^T \\ \mathbf{p}_2^T \\ \vdots \\ \mathbf{p}_{N_{\text{joint}}}^T \end{pmatrix} \in \mathbb{R}^{N_{\text{joint}} \times m} \quad (4.81)$$

<sup>11</sup> 数学的帰納法で証明できる . <http://mat.fsv.cvut.cz/gcg/sbornik/prochazkova.pdf>

式 (4.80) は，制御点を縦に並べたベクトルとして分離して，以下のようにも表現できる．

$$\boldsymbol{\theta}(t) = \begin{pmatrix} \theta_1(t) \\ \theta_2(t) \\ \vdots \\ \theta_{N_{joint}}(t) \end{pmatrix} = \begin{pmatrix} \mathbf{b}_n^T(t) \mathbf{p}_1 \\ \mathbf{b}_n^T(t) \mathbf{p}_2 \\ \vdots \\ \mathbf{b}_n^T(t) \mathbf{p}_{N_{joint}} \end{pmatrix} = \begin{pmatrix} \mathbf{b}_n^T(t) & & \mathbf{O} \\ & \mathbf{b}_n^T(t) & \\ & & \ddots \\ \mathbf{O} & & & \mathbf{b}_n^T(t) \end{pmatrix} \begin{pmatrix} \mathbf{p}_1 \\ \mathbf{p}_2 \\ \vdots \\ \mathbf{p}_{N_{joint}} \end{pmatrix} = \mathbf{B}_n(t) \mathbf{p} \in \mathbb{R}^{N_{joint}} \quad (4.82)$$

ただし，

$$\mathbf{B}_n(t) \stackrel{\text{def}}{=} \begin{pmatrix} \mathbf{b}_n^T(t) & & \mathbf{O} \\ & \mathbf{b}_n^T(t) & \\ & & \ddots \\ \mathbf{O} & & & \mathbf{b}_n^T(t) \end{pmatrix} \in \mathbb{R}^{N_{joint} \times m N_{joint}}, \quad \mathbf{p} \stackrel{\text{def}}{=} \begin{pmatrix} \mathbf{p}_1 \\ \mathbf{p}_2 \\ \vdots \\ \mathbf{p}_{N_{joint}} \end{pmatrix} \in \mathbb{R}^{m N_{joint}} \quad (4.83)$$

## B スプラインによる関節角軌道の微分

式 (4.80) と式 (4.74) から，関節角速度軌道は次式で得られる．

$$\dot{\boldsymbol{\theta}}(t) = \mathbf{P} \dot{\mathbf{b}}_n(t) \quad (4.84)$$

$$= \mathbf{P} \mathbf{D} \mathbf{b}_{n-1}(t) \quad (4.85)$$

$$= \begin{pmatrix} \mathbf{p}_1^T \\ \vdots \\ \mathbf{p}_{N_{joint}}^T \end{pmatrix} \mathbf{D} \mathbf{b}_{n-1}(t) \quad (4.86)$$

$$= \begin{pmatrix} \mathbf{p}_1^T \mathbf{D} \mathbf{b}_{n-1}(t) \\ \vdots \\ \mathbf{p}_{N_{joint}}^T \mathbf{D} \mathbf{b}_{n-1}(t) \end{pmatrix} \quad (4.87)$$

$$= \begin{pmatrix} \mathbf{b}_{n-1}^T(t) \mathbf{D}^T \mathbf{p}_1 \\ \vdots \\ \mathbf{b}_{n-1}^T(t) \mathbf{D}^T \mathbf{p}_{N_{joint}} \end{pmatrix} \quad (4.88)$$

$$= \begin{pmatrix} \mathbf{b}_{n-1}^T(t) \mathbf{D}^T & & \mathbf{O} \\ & \ddots & \\ \mathbf{O} & & \mathbf{b}_{n-1}^T(t) \mathbf{D}^T \end{pmatrix} \begin{pmatrix} \mathbf{p}_1 \\ \vdots \\ \mathbf{p}_{N_{joint}} \end{pmatrix} \quad (4.89)$$

$$= \begin{pmatrix} \mathbf{b}_{n-1}^T(t) & & \mathbf{O} \\ & \ddots & \\ \mathbf{O} & & \mathbf{b}_{n-1}^T(t) \end{pmatrix} \begin{pmatrix} \mathbf{D}^T & & \mathbf{O} \\ & \ddots & \\ \mathbf{O} & & \mathbf{D}^T \end{pmatrix} \begin{pmatrix} \mathbf{p}_1 \\ \vdots \\ \mathbf{p}_{N_{joint}} \end{pmatrix} \quad (4.90)$$

$$= \mathbf{B}_{n-1}(t) \hat{\mathbf{D}}_1 \mathbf{p} \quad (4.91)$$

ただし，

$$\hat{\mathbf{D}}_1 = \begin{pmatrix} \mathbf{D}^T & & \mathbf{O} \\ & \mathbf{D}^T & \\ & & \ddots \\ \mathbf{O} & & & \mathbf{D}^T \end{pmatrix} \in \mathbb{R}^{m N_{joint} \times m N_{joint}} \quad (4.92)$$

同様に、関節角軌道の  $k$  階微分は次式で得られる。

$$\boldsymbol{\theta}^{(k)}(t) = \frac{d^{(k)}\boldsymbol{\theta}(t)}{dt^{(k)}} \quad (4.93)$$

$$= \boldsymbol{P}\boldsymbol{D}^k\boldsymbol{b}_{n-k}(t) \quad (4.94)$$

$$= \boldsymbol{B}_{n-k}(t)\hat{\boldsymbol{D}}_k\boldsymbol{p} \quad (4.95)$$

ただし、

$$\hat{\boldsymbol{D}}_k = \begin{pmatrix} (\boldsymbol{D}^k)^T & & \boldsymbol{O} \\ & \ddots & \\ \boldsymbol{O} & & (\boldsymbol{D}^k)^T \end{pmatrix} = (\hat{\boldsymbol{D}}_1)^k \in \mathbb{R}^{mN_{joint} \times mN_{joint}} \quad (4.96)$$

計算時間は式 (4.94) のほうが式 (4.95) より速い。

#### エンドエフェクタ位置姿勢拘束のタスク関数

関節角  $\boldsymbol{\theta} \in \mathbb{R}^{N_{joint}}$  からエンドエフェクタ位置姿勢  $\boldsymbol{r} \in \mathbb{R}^6$  への写像を  $\boldsymbol{f}(\boldsymbol{\theta})$  で表す。

$$\boldsymbol{r} = \boldsymbol{f}(\boldsymbol{\theta}) \quad (4.97)$$

関節角軌道が式 (4.82) で表現されるとき、エンドエフェクタ軌道は次式で表される。

$$\boldsymbol{r}(t) = \boldsymbol{f}(\boldsymbol{\theta}(t)) = \boldsymbol{f}(\boldsymbol{B}_n(t)\boldsymbol{p}) \quad (4.98)$$

$l = 1, 2, \dots, N_{tm}$  について、時刻  $t_l$  でエンドエフェクタの位置姿勢が  $\boldsymbol{r}_l$  であるタスクのタスク関数は次式で表される。以降では、 $t_l$  をタイミングと呼ぶ。

$$\boldsymbol{e}(\boldsymbol{p}, \boldsymbol{t}) \stackrel{\text{def}}{=} \begin{pmatrix} \boldsymbol{e}_1(\boldsymbol{p}, t) \\ \boldsymbol{e}_2(\boldsymbol{p}, t) \\ \vdots \\ \boldsymbol{e}_{N_{tm}}(\boldsymbol{p}, t) \end{pmatrix} = \begin{pmatrix} \boldsymbol{r}_1 - \boldsymbol{f}(\boldsymbol{\theta}(t_1)) \\ \boldsymbol{r}_2 - \boldsymbol{f}(\boldsymbol{\theta}(t_2)) \\ \vdots \\ \boldsymbol{r}_{N_{tm}} - \boldsymbol{f}(\boldsymbol{\theta}(t_{N_{tm}})) \end{pmatrix} = \begin{pmatrix} \boldsymbol{r}_1 - \boldsymbol{f}(\boldsymbol{B}_n(t_1)\boldsymbol{p}) \\ \boldsymbol{r}_2 - \boldsymbol{f}(\boldsymbol{B}_n(t_2)\boldsymbol{p}) \\ \vdots \\ \boldsymbol{r}_{N_{tm}} - \boldsymbol{f}(\boldsymbol{B}_n(t_{N_{tm}})\boldsymbol{p}) \end{pmatrix} \in \mathbb{R}^{6N_{tm}} \quad (4.99)$$

ただし、

$$\boldsymbol{e}_l(\boldsymbol{p}, t) \stackrel{\text{def}}{=} \boldsymbol{r}_l - \boldsymbol{f}(\boldsymbol{\theta}(t_l)) = \boldsymbol{r}_l - \boldsymbol{f}(\boldsymbol{B}_n(t_l)\boldsymbol{p}) \in \mathbb{R}^6 \quad (l = 1, 2, \dots, N_{tm}) \quad (4.100)$$

$$\boldsymbol{t} \stackrel{\text{def}}{=} \begin{pmatrix} t_1 \\ t_2 \\ \vdots \\ t_{N_{tm}} \end{pmatrix} \in \mathbb{R}^{N_{tm}} \quad (4.101)$$

このタスクを実現する関節角軌道は、次の評価関数を最小にする制御点  $\boldsymbol{p}$ 、タイミング  $\boldsymbol{t}$  を求めることで導出することができる。

$$F(\boldsymbol{p}, \boldsymbol{t}) \stackrel{\text{def}}{=} \frac{1}{2} \|\boldsymbol{e}(\boldsymbol{p}, \boldsymbol{t})\|^2 \quad (4.102)$$

$$= \frac{1}{2} \sum_{l=1}^{N_{tm}} \|\boldsymbol{r}_l - \boldsymbol{f}(\boldsymbol{\theta}(t_l))\|^2 \quad (4.103)$$

$$= \frac{1}{2} \sum_{l=1}^{N_{tm}} \|\boldsymbol{r}_l - \boldsymbol{f}(\boldsymbol{B}_n(t_l)\boldsymbol{p})\|^2 \quad (4.104)$$

また,  $l$  番目の幾何拘束の許容誤差を  $e_{tol,l} \geq 0 \in \mathbb{R}^6$  とする場合, タスク関数  $\tilde{e}_l(\mathbf{p}, t)$  は次式で表される.

$$\tilde{e}_{l,i}(\mathbf{p}, t) \stackrel{\text{def}}{=} \begin{cases} e_{l,i}(\mathbf{p}, t) - e_{tol,l,i} & e_{l,i}(\mathbf{p}, t) > e_{tol,l,i} \\ e_{l,i}(\mathbf{p}, t) + e_{tol,l,i} & e_{l,i}(\mathbf{p}, t) < -e_{tol,l,i} \\ 0 & \text{otherwise} \end{cases} \quad (i = 1, 2, \dots, 6) \quad (4.105)$$

$\tilde{e}_{l,i}(\mathbf{p}, t)$  は  $\tilde{e}_l(\mathbf{p}, t)$  の  $i$  番目の要素である.  $e_{l,i}(\mathbf{p}, t)$  は  $e(\mathbf{p}, t)$  の  $i$  番目の要素である.

#### タスク関数を制御点で微分したヤコビ行列

式 (4.104) を目的関数とする最適化問題を Gauss-Newton 法, Levenberg-Marquardt 法や逐次二次計画法で解く場合, タスク関数 (4.99) のヤコビ行列が必要となる.

各時刻でのエンドエフェクタ位置姿勢拘束のタスク関数  $e_l(\mathbf{p}, t)$  の制御点  $\mathbf{p}$  に対するヤコビ行列は次式で求められる.

$$\frac{\partial e_l(\mathbf{p}, t)}{\partial \mathbf{p}} = \frac{\partial}{\partial \mathbf{p}} \{ \mathbf{r}_l - \mathbf{f}(\mathbf{B}_n(t_l)\mathbf{p}) \} \quad (4.106)$$

$$= -\frac{\partial}{\partial \mathbf{p}} \mathbf{f}(\mathbf{B}_n(t_l)\mathbf{p}) \quad (4.107)$$

$$= -\left. \frac{\partial \mathbf{f}}{\partial \boldsymbol{\theta}} \right|_{\boldsymbol{\theta}=\boldsymbol{\theta}(t_l)} \frac{\partial \boldsymbol{\theta}}{\partial \mathbf{p}} \quad (4.108)$$

$$= -\mathbf{J}(\boldsymbol{\theta}(t_l)) \frac{\partial}{\partial \mathbf{p}} \{ \mathbf{B}_n(t_l)\mathbf{p} \} \quad (4.109)$$

$$= -\mathbf{J}(\boldsymbol{\theta}(t_l)) \mathbf{B}_n(t_l) \quad (4.110)$$

途中の変形で,  $\boldsymbol{\theta}(\mathbf{p}; t) = \mathbf{B}_n(t)\mathbf{p}$  であることを利用した. ただし,

$$\mathbf{J} \stackrel{\text{def}}{=} \frac{\partial \mathbf{f}}{\partial \boldsymbol{\theta}} \quad (4.111)$$

#### タスク関数をタイミングで微分したヤコビ行列

各時刻でのエンドエフェクタ位置姿勢拘束のタスク関数  $e_l(\mathbf{p}, t)$  のタイミング  $t$  に対するヤコビ行列は次式で求められる.

$$\frac{\partial e_l(\mathbf{p}, t)}{\partial t_l} = \frac{\partial}{\partial t_l} \{ \mathbf{r}_l - \mathbf{f}(\mathbf{P}\mathbf{b}_n(t_l)) \} \quad (4.112)$$

$$= -\frac{\partial}{\partial t_l} \mathbf{f}(\mathbf{P}\mathbf{b}_n(t_l)) \quad (4.113)$$

$$= -\left. \frac{\partial \mathbf{f}}{\partial \boldsymbol{\theta}} \right|_{\boldsymbol{\theta}=\boldsymbol{\theta}(t_l)} \frac{\partial \boldsymbol{\theta}}{\partial t_l} \quad (4.114)$$

$$= -\mathbf{J}(\boldsymbol{\theta}(t_l)) \frac{\partial}{\partial t_l} \{ \mathbf{P}\mathbf{b}_n(t_l) \} \quad (4.115)$$

$$= -\mathbf{J}(\boldsymbol{\theta}(t_l)) \mathbf{P}\dot{\mathbf{b}}_n(t_l) \quad (4.116)$$

$$= -\mathbf{J}(\boldsymbol{\theta}(t_l)) \mathbf{P}\mathbf{D}\mathbf{b}_{n-1}(t_l) \quad (4.117)$$

途中の変形で,  $\boldsymbol{\theta}(\mathbf{p}; t) = \mathbf{P}\mathbf{b}_n(t)$  であることを利用した.

## 初期・終端関節速度・加速度のタスク関数とヤコビ行列

初期，終端時刻の関節速度，加速度はゼロであるべきである．タスク関数は次式となる．

$$e_{sv}(\mathbf{p}, t) \stackrel{\text{def}}{=} \dot{\boldsymbol{\theta}}(t_s) \quad (4.118)$$

$$= \mathbf{B}_{n-1}(t_s) \hat{\mathbf{D}}_1 \mathbf{p} \quad (4.119)$$

$$= \mathbf{PD} \mathbf{b}_{n-1}(t_s) \quad (4.120)$$

$$e_{fv}(\mathbf{p}, t) \stackrel{\text{def}}{=} \dot{\boldsymbol{\theta}}(t_f) \quad (4.121)$$

$$= \mathbf{B}_{n-1}(t_f) \hat{\mathbf{D}}_1 \mathbf{p} \quad (4.122)$$

$$= \mathbf{PD} \mathbf{b}_{n-1}(t_f) \quad (4.123)$$

$$e_{sa}(\mathbf{p}, t) \stackrel{\text{def}}{=} \ddot{\boldsymbol{\theta}}(t_s) \quad (4.124)$$

$$= \mathbf{B}_{n-2}(t_s) \hat{\mathbf{D}}_2 \mathbf{p} \quad (4.125)$$

$$= \mathbf{PD}^2 \mathbf{b}_{n-2}(t_s) \quad (4.126)$$

$$e_{fa}(\mathbf{p}, t) \stackrel{\text{def}}{=} \ddot{\boldsymbol{\theta}}(t_f) \quad (4.127)$$

$$= \mathbf{B}_{n-2}(t_f) \hat{\mathbf{D}}_2 \mathbf{p} \quad (4.128)$$

$$= \mathbf{PD}^2 \mathbf{b}_{n-2}(t_f) \quad (4.129)$$

制御点で微分したヤコビ行列は次式で表される．

$$\frac{\partial e_{sv}(\mathbf{p}, t)}{\partial \mathbf{p}} = \mathbf{B}_{n-1}(t_s) \hat{\mathbf{D}}_1 \quad (4.130)$$

$$\frac{\partial e_{fv}(\mathbf{p}, t)}{\partial \mathbf{p}} = \mathbf{B}_{n-1}(t_f) \hat{\mathbf{D}}_1 \quad (4.131)$$

$$\frac{\partial e_{sa}(\mathbf{p}, t)}{\partial \mathbf{p}} = \mathbf{B}_{n-2}(t_s) \hat{\mathbf{D}}_2 \quad (4.132)$$

$$\frac{\partial e_{fa}(\mathbf{p}, t)}{\partial \mathbf{p}} = \mathbf{B}_{n-2}(t_f) \hat{\mathbf{D}}_2 \quad (4.133)$$

初期時刻，終端時刻で微分したヤコビ行列は次式で表される．

$$\frac{\partial e_{sv}(\mathbf{p}, t)}{\partial t_s} = \mathbf{PD} \frac{\partial \mathbf{b}_{n-1}(t_s)}{\partial t_s} = \mathbf{PD}^2 \mathbf{b}_{n-2}(t_s) \quad (4.134)$$

$$\frac{\partial e_{fv}(\mathbf{p}, t)}{\partial t_f} = \mathbf{PD} \frac{\partial \mathbf{b}_{n-1}(t_f)}{\partial t_f} = \mathbf{PD}^2 \mathbf{b}_{n-2}(t_f) \quad (4.135)$$

$$\frac{\partial e_{sa}(\mathbf{p}, t)}{\partial t_s} = \mathbf{PD}^2 \frac{\partial \mathbf{b}_{n-2}(t_s)}{\partial t_s} = \mathbf{PD}^3 \mathbf{b}_{n-3}(t_s) \quad (4.136)$$

$$\frac{\partial e_{fa}(\mathbf{p}, t)}{\partial t_f} = \mathbf{PD}^2 \frac{\partial \mathbf{b}_{n-2}(t_f)}{\partial t_f} = \mathbf{PD}^3 \mathbf{b}_{n-3}(t_f) \quad (4.137)$$

## 関節角上下限制約

式 (4.78) の関節角軌道定義において，

$$\mathbf{p}_j \leq \theta_{max,j} \mathbf{1}_m \quad (4.138)$$

のとき，B スプラインの凸包性 (式 (4.72)，式 (4.73)) より次式が成り立つ．ただし， $\mathbf{1}_m \in \mathbb{R}^m$  は全要素が 1

の  $m$  次元ベクトルである .

$$\theta_j(t) = \sum_{i=0}^{m-1} p_{j,i} b_{i,n}(t) \quad (4.139)$$

$$\leq \sum_{i=0}^{m-1} \theta_{max,j} b_{i,n}(t) \quad (4.140)$$

$$= \theta_{max,j} \sum_{i=0}^{m-1} b_{i,n}(t) \quad (4.141)$$

$$= \theta_{max,j} \quad (4.142)$$

同様に ,  $\theta_{min,j} \mathbf{1}_m \leq \mathbf{p}_j$  とすれば ,  $\theta_{min,j} \leq \theta_j(t)$  が成り立つ .

したがって ,  $j$  番目の関節角の上下限を  $\theta_{max,j}, \theta_{min,j}$  とすると , 次式の制約を制御点に課すことで , 関節角上下限制約を満たす関節角軌道が得られる .

$$\theta_{min,j} \mathbf{1}_m \leq \mathbf{p}_j \leq \theta_{max,j} \mathbf{1}_m \quad (j = 1, 2, \dots, N_{joint}) \quad (4.143)$$

つまり ,

$$\hat{\mathbf{E}} \boldsymbol{\theta}_{min} \leq \mathbf{p} \leq \hat{\mathbf{E}} \boldsymbol{\theta}_{max} \quad (4.144)$$

$$\Leftrightarrow \begin{pmatrix} \mathbf{I} \\ -\mathbf{I} \end{pmatrix} \mathbf{p} \geq \begin{pmatrix} \hat{\mathbf{E}} \boldsymbol{\theta}_{min} \\ -\hat{\mathbf{E}} \boldsymbol{\theta}_{max} \end{pmatrix} \quad (4.145)$$

ただし ,

$$\hat{\mathbf{E}} \stackrel{\text{def}}{=} \begin{pmatrix} \mathbf{1}_m & & \mathbf{0}_m \\ & \mathbf{1}_m & \\ & & \ddots \\ \mathbf{0}_m & & \mathbf{1}_m \end{pmatrix} \in \mathbb{R}^{m N_{joint} \times N_{joint}} \quad (4.146)$$

これは , 逐次二次計画法の中で , 次式の不等式制約となる .

$$\begin{pmatrix} \mathbf{I} \\ -\mathbf{I} \end{pmatrix} \Delta \mathbf{p} \geq \begin{pmatrix} \hat{\mathbf{E}} \boldsymbol{\theta}_{min} - \mathbf{p} \\ -\hat{\mathbf{E}} \boldsymbol{\theta}_{max} + \mathbf{p} \end{pmatrix} \quad (4.147)$$

関節角速度・角加速度上下限制約

式 (4.78) と式 (4.74) より , 関節角速度軌道 , 角加速度軌道は次式で表される .

$$\dot{\theta}_j(t) = \mathbf{p}_j^T \dot{\mathbf{b}}_n(t) = \mathbf{p}_j^T \mathbf{D} \mathbf{b}_{n-1}(t) = (\mathbf{D}^T \mathbf{p}_j)^T \mathbf{b}_{n-1}(t) \in \mathbb{R} \quad (t_s \leq t \leq t_f) \quad (4.148)$$

$$\ddot{\theta}_j(t) = \mathbf{p}_j^T \ddot{\mathbf{b}}_n(t) = \mathbf{p}_j^T \mathbf{D}^2 \mathbf{b}_{n-2}(t) = ((\mathbf{D}^2)^T \mathbf{p}_j)^T \mathbf{b}_{n-2}(t) \in \mathbb{R} \quad (t_s \leq t \leq t_f) \quad (4.149)$$

$j$  番目の関節角速度 , 角加速度の上限を  $v_{max,j}, a_{max,j}$  とする . 関節角上下限制約の導出と同様に考えると , 次式の制約を制御点に課すことで , 関節角速度・角加速度上下限制約を満たす関節角軌道が得られる .

$$-v_{max,j} \mathbf{1}_m \leq \mathbf{D}^T \mathbf{p}_j \leq v_{max,j} \mathbf{1}_m \quad (j = 1, 2, \dots, N_{joint}) \quad (4.150)$$

$$-a_{max,j} \mathbf{1}_m \leq (\mathbf{D}^2)^T \mathbf{p}_j \leq a_{max,j} \mathbf{1}_m \quad (j = 1, 2, \dots, N_{joint}) \quad (4.151)$$

つまり,

$$-\hat{\mathbf{E}}\mathbf{v}_{max} \leq \hat{\mathbf{D}}_1\mathbf{p} \leq \hat{\mathbf{E}}\mathbf{v}_{max} \quad (4.152)$$

$$\Leftrightarrow \begin{pmatrix} \hat{\mathbf{D}}_1 \\ -\hat{\mathbf{D}}_1 \end{pmatrix} \mathbf{p} \geq \begin{pmatrix} -\hat{\mathbf{E}}\mathbf{v}_{max} \\ -\hat{\mathbf{E}}\mathbf{v}_{max} \end{pmatrix} \quad (4.153)$$

$$-\hat{\mathbf{E}}\mathbf{a}_{max} \leq \hat{\mathbf{D}}_2\mathbf{p} \leq \hat{\mathbf{E}}\mathbf{a}_{max} \quad (4.154)$$

$$\Leftrightarrow \begin{pmatrix} \hat{\mathbf{D}}_2 \\ -\hat{\mathbf{D}}_2 \end{pmatrix} \mathbf{p} \geq \begin{pmatrix} -\hat{\mathbf{E}}\mathbf{a}_{max} \\ -\hat{\mathbf{E}}\mathbf{a}_{max} \end{pmatrix} \quad (4.155)$$

これは, 逐次二次計画法の中で, 次式の不等式制約となる.

$$\begin{pmatrix} \hat{\mathbf{D}}_1 \\ -\hat{\mathbf{D}}_1 \end{pmatrix} \Delta\mathbf{p} \geq \begin{pmatrix} -\hat{\mathbf{E}}\mathbf{v}_{max} - \hat{\mathbf{D}}_1\mathbf{p} \\ -\hat{\mathbf{E}}\mathbf{v}_{max} + \hat{\mathbf{D}}_1\mathbf{p} \end{pmatrix} \quad (4.156)$$

$$\begin{pmatrix} \hat{\mathbf{D}}_2 \\ -\hat{\mathbf{D}}_2 \end{pmatrix} \Delta\mathbf{p} \geq \begin{pmatrix} -\hat{\mathbf{E}}\mathbf{a}_{max} - \hat{\mathbf{D}}_2\mathbf{p} \\ -\hat{\mathbf{E}}\mathbf{a}_{max} + \hat{\mathbf{D}}_2\mathbf{p} \end{pmatrix} \quad (4.157)$$

タイミング上下限制約

タイミングが初期, 終端時刻の間に含まれる制約は次式で表される.

$$t_s \leq t_l \leq t_f \quad (l = 1, 2, \dots, N_{tm}) \quad (4.158)$$

$$\Leftrightarrow t_s \mathbf{1} \leq \mathbf{t} \leq t_f \mathbf{1} \quad (4.159)$$

$$\Leftrightarrow \begin{pmatrix} \mathbf{I} \\ -\mathbf{I} \end{pmatrix} \mathbf{t} \geq \begin{pmatrix} t_s \mathbf{1} \\ -t_f \mathbf{1} \end{pmatrix} \quad (4.160)$$

これは, 逐次二次計画法の中で, 次式の不等式制約となる.

$$\begin{pmatrix} \mathbf{I} \\ -\mathbf{I} \end{pmatrix} \Delta\mathbf{t} \geq \begin{pmatrix} t_s \mathbf{1} - \mathbf{t} \\ -t_f \mathbf{1} + \mathbf{t} \end{pmatrix} \quad (4.161)$$

$$(4.162)$$

また, タイミングの順序が入れ替わることを許容しない場合, その制約は次式で表される.

$$t_l \leq t_{l+1} \quad (l = 1, 2, \dots, N_{tm} - 1) \quad (4.163)$$

$$\Leftrightarrow -t_l + t_{l+1} \geq 0 \quad (l = 1, 2, \dots, N_{tm} - 1) \quad (4.164)$$

$$\Leftrightarrow \mathbf{D}_{tm} \mathbf{t} \geq \mathbf{0} \quad (4.165)$$

ただし,

$$\mathbf{D}_{tm} = \begin{pmatrix} -1 & 1 & & & \mathbf{O} \\ & -1 & 1 & & \\ & & & \ddots & \\ \mathbf{O} & & & & -1 & 1 \end{pmatrix} \in \mathbb{R}^{(N_{tm}-1) \times N_{tm}} \quad (4.166)$$

これは, 逐次二次計画法の中で, 次式の不等式制約となる.

$$\mathbf{D}_{tm} \Delta\mathbf{t} \geq -\mathbf{D}_{tm} \mathbf{t} \quad (4.167)$$

### 関節角微分二乗積分最小化

関節角微分の二乗積分は次式で得られる．

$$F_{sqT,k}(\mathbf{p}) = \int_{t_s}^{t_f} \left\| \boldsymbol{\theta}^{(k)}(t) \right\|^2 dt \quad (4.168)$$

$$= \int_{t_s}^{t_f} \left\| \mathbf{B}_{n-k}(t) \hat{\mathbf{D}}_k \mathbf{p} \right\|^2 dt \quad (4.169)$$

$$= \int_{t_s}^{t_f} \left( \mathbf{B}_{n-k}(t) \hat{\mathbf{D}}_k \mathbf{p} \right)^T \left( \mathbf{B}_{n-k}(t) \hat{\mathbf{D}}_k \mathbf{p} \right) dt \quad (4.170)$$

$$= \mathbf{p}^T \left\{ \int_{t_s}^{t_f} \left( \mathbf{B}_{n-k}(t) \hat{\mathbf{D}}_k \right)^T \mathbf{B}_{n-k}(t) \hat{\mathbf{D}}_k dt \right\} \mathbf{p} \quad (4.171)$$

$$= \mathbf{p}^T \mathbf{H}_k \mathbf{p} \quad (4.172)$$

ただし，

$$\mathbf{H}_k = \int_{t_s}^{t_f} \left( \mathbf{B}_{n-k}(t) \hat{\mathbf{D}}_k \right)^T \mathbf{B}_{n-k}(t) \hat{\mathbf{D}}_k dt \quad (4.173)$$

$$\mathbf{B}_{n-k}(t) \hat{\mathbf{D}}_k = \begin{pmatrix} \mathbf{b}_{n-k}^T(t) & \mathbf{O} \\ & \ddots \\ \mathbf{O} & \mathbf{b}_{n-k}^T(t) \end{pmatrix} \begin{pmatrix} (\mathbf{D}^k)^T & \mathbf{O} \\ & \ddots \\ \mathbf{O} & (\mathbf{D}^k)^T \end{pmatrix} \quad (4.174)$$

$$= \begin{pmatrix} \mathbf{b}_{n-k}^T(t) (\mathbf{D}^k)^T & \mathbf{O} \\ & \ddots \\ \mathbf{O} & \mathbf{b}_{n-k}^T(t) (\mathbf{D}^k)^T \end{pmatrix} \quad (4.175)$$

$$= \begin{pmatrix} \left( \mathbf{D}^k \mathbf{b}_{n-k}(t) \right)^T & \mathbf{O} \\ & \ddots \\ \mathbf{O} & \left( \mathbf{D}^k \mathbf{b}_{n-k}(t) \right)^T \end{pmatrix} \quad (4.176)$$

$$\left( \mathbf{B}_{n-k}(t) \hat{\mathbf{D}}_k \right)^T \mathbf{B}_{n-k}(t) = \begin{pmatrix} \left( \mathbf{D}^k \mathbf{b}_{n-k}(t) \right)^T & \mathbf{O} \\ & \ddots \\ \mathbf{O} & \left( \mathbf{D}^k \mathbf{b}_{n-k}(t) \right)^T \end{pmatrix}^T \begin{pmatrix} \left( \mathbf{D}^k \mathbf{b}_{n-k}(t) \right)^T & \mathbf{O} \\ & \ddots \\ \mathbf{O} & \left( \mathbf{D}^k \mathbf{b}_{n-k}(t) \right)^T \end{pmatrix} \quad (4.177)$$

$$= \begin{pmatrix} \left( \mathbf{D}^k \mathbf{b}_{n-k}(t) \right) \left( \mathbf{D}^k \mathbf{b}_{n-k}(t) \right)^T & & \\ & \ddots & \\ \mathbf{O} & & \left( \mathbf{D}^k \mathbf{b}_{n-k}(t) \right) \left( \mathbf{D}^k \mathbf{b}_{n-k}(t) \right)^T \end{pmatrix} \quad (4.178)$$

これを逐次二次計画問題において，二次形式の正則化項として目的関数に加えることで，滑らかな動作が生成されることが期待される．

### 動作期間の最小化

動作期間  $(t_f - t_s)$  の二乗は次式で表される．

$$F_{duration}(\mathbf{t}) = |t_1 - t_{N_{tm}}|^2 \quad (4.179)$$

$$= \mathbf{t}^T \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \mathbf{t} \quad (4.180)$$



ただし, 初期時刻  $t_s = t_1$ , 終端時刻  $t_f = t_{N_{tm}}$  がタイミングベクトル  $t$  の最初, 最後の要素であるとする. これを逐次二次計画問題において, 二次形式の正則化項として目的関数に加えることで, 短時間でタスクを実現する動作が生成されることが期待される.

#### 4.2.2 B スプラインを用いた関節軌道生成の実装

### bspline-configuration-task

[class]

```

:super    propertied-object
:slots    (_robot robot instance)
           (_control-vector  $\mathbf{p}$ )
           (_timing-vector  $\mathbf{t}$ )
           (_num-kin  $N_{kin} := |\mathcal{T}^{kin-trg}| = |\mathcal{T}^{kin-att}|$ )
           (_num-joint  $N_{joint} := |\mathcal{J}|$ )
           (_num-control-point  $N_{ctrl}$ )
           (_num-timing  $N_{tm}$ )
           (_bspline-order B-spline order,  $n$ )
           (_dim-control-vector  $dim(\mathbf{p})$ )
           (_dim-timing-vector  $dim(\mathbf{t})$ )
           (_dim-config  $dim(\mathbf{q})$ )
           (_dim-task  $dim(\mathbf{e})$ )
           (_num-collision  $N_{col} :=$  number of collision check pairs)
           (_stationery-start-finish-task-scale  $k_{stat}$ )
           (_first-diff-square-integration-regular-scale  $k_{sqr,1}$ )
           (_second-diff-square-integration-regular-scale  $k_{sqr,2}$ )
           (_third-diff-square-integration-regular-scale  $k_{sqr,3}$ )
           (_motion-duration-regular-scale  $k_{duration}$ )
           (_norm-regular-scale-max  $k_{max,p}$ )
           (_norm-regular-scale-offset  $k_{off,p}$ )
           (_timing-norm-regular-scale-max  $k_{max,t}$ )
           (_timing-norm-regular-scale-offset  $k_{off,t}$ )
           (_joint-list  $\mathcal{J}$ )
           (_start-time  $t_s$ )
           (_finish-time  $t_f$ )
           (_kin-time-list  $\{t_1^{kin-tm}, t_2^{kin-tm}, \dots, t_{N_{kin}}^{kin-tm}\}$ )
           (_kin-variable-timing-list list of bool. t for variable timing.)
           (_kin-target-coords-list  $\mathcal{T}^{kin-trg}$ )
           (_kin-attention-coords-list  $\mathcal{T}^{kin-att}$ )
           (_kin-pos-tolerance-list list of position tolerance  $e_{tol,pos}$  [m])
           (_kin-rot-tolerance-list list of rotation tolerance  $e_{tol,rot}$  [rad])
           (_joint-angle-margin margin of  $\theta$  [deg] [mm])
           (_collision-pair-list list of bodyset-link or body pair)
           (_keep-timing-order? whether to keep order of timing  $\mathbf{t}$  or not)
           (_bspline-matrix buffer for  $\mathbf{B}_n(t)$ )

```

(\_diff-mat buffer for  $D^k$ )  
 (\_diff-mat-list buffer for  $\{D^1, D^2, \dots, D^K\}$ )  
 (\_extended-diff-mat-list buffer for  $\{\hat{D}_1, \hat{D}_2, \dots, \hat{D}_K\}$ )  
 (\_task-jacobi buffer for  $\frac{\partial e}{\partial q}$ )  
 (\_regular-mat buffer for  $W_{reg}$ )  
 (\_regular-vec buffer for  $v_{reg}$ )

B スプラインを利用した軌道生成のためのコンフィギュレーション  $q$  とタスク関数  $e(q)$  のクラス .

コンフィギュレーション  $q$  の取得・更新, タスク関数  $e(q)$  の取得, タスク関数のヤコビ行列  $\frac{\partial e(q)}{\partial q}$  の取得, コンフィギュレーションの等式・不等式制約  $A, b, C, d$  の取得のためのメソッドが定義されている .

コンフィギュレーション・タスク関数を定めるために, 初期化時に以下を与える

- ロボット

robot ロボットのインスタンス

joint-list  $\mathcal{J}$  関節

- B スプラインのパラメータ

start-time  $t_s$  B スプラインの使用区間の初期時刻

finish-time  $t_f$  B スプラインの使用区間の終端時刻

num-control-point  $N_{ctrl}$  制御点の個数

bspline-order  $n$  B スプラインの次数

- 幾何拘束

kin-target-coords-list  $\mathcal{T}^{kin-try}$  幾何到達目標位置姿勢リスト

kin-attention-coords-list  $\mathcal{T}^{kin-att}$  幾何到達着目位置姿勢リスト

kin-time-list  $\{t_1^{kin-tm}, t_2^{kin-tm}, \dots, t_{N_{kin}}^{kin-tm}\}$  幾何到達タイミングリスト

kin-variable-timing-list 幾何到達タイミングが可変か (t), 固定か (nil) のリスト . このリスト内の  $t$  の個数がタイミング  $t$  の次元  $N_{tm}$  となる .

コンフィギュレーション  $q$  は以下から構成される .

$$q := \begin{pmatrix} p \\ t \end{pmatrix} \quad (4.181)$$

$p \in \mathbb{R}^{N_{ctrl}N_{joint}}$  制御点 (B スプライン基底関数の山の高さ) [rad] [m]

$t \in \mathbb{R}^{N_{tm}}$  タイミング (幾何拘束タスクの課される時刻) [sec]

タスク関数  $e(q)$  は以下から構成される .

$$e(q) := \begin{pmatrix} e^{kin}(q) \\ e^{stat}(q) \end{pmatrix} \in \mathbb{R}^{6N_{kin}+4N_{joint}} \quad (4.182)$$

$e^{kin}(q) \in \mathbb{R}^{6N_{kin}}$  幾何到達拘束 [rad] [m]

$e^{stat}(q) \in \mathbb{R}^{4N_{joint}}$  初期, 終端時刻静止拘束 [rad][rad/s][rad/s<sup>2</sup>][m][m/s][m/s<sup>2</sup>]

<b>:init</b> $\mathcal{E}_{key}$ (name)	[method]
(robot) (joint-list (send robot :joint-list)) (start-time 0.0) (finish-time 10.0) (num-control-point 10) (bspline-order 3) (kin-time-list) (kin-variable-timing-list (make-list (length kin-time-list) :initial-element nil)) (kin-target-coords-list) (kin-attention-coords-list) (kin-pos-tolerance-list (make-list (length kin-time-list) :initial-element 0.0)) (kin-rot-tolerance-list (make-list (length kin-time-list) :initial-element 0.0)) (joint-angle-margin 3.0) (collision-pair-list) (keep-timing-order? t) (stationery-start-finish-task-scale 0.0) (first-diff-square-integration-regular-scale 0.0) (second-diff-square-integration-regular-scale 0.0) (third-diff-square-integration-regular-scale 0.0) (motion-duration-regular-scale 0.0) (norm-regular-scale-max 1.000000e-05) (norm-regular-scale-offset 1.000000e-07) (timing-norm-regular-scale-max 1.000000e-05) (timing-norm-regular-scale-offset 1.000000e-07)	
Initialize instance	
<b>:robot</b>	[method]
return robot instance	
<b>:joint-list</b>	[method]
return $\mathcal{J}$	
<b>:num-kin</b>	[method]
return $N_{kin} :=  \mathcal{T}^{kin-trg}  =  \mathcal{T}^{kin-att} $	
<b>:num-joint</b>	[method]
return $N_{joint} :=  \mathcal{J} $	
<b>:num-control-point</b>	[method]
return $N_{ctrl}$	
<b>:num-timing</b>	[method]
return $N_{tm}$	
<b>:num-collision</b>	[method]
return $N_{col} :=$ number of collision check pairs	
<b>:dim-config</b>	[method]
return $dim(\mathbf{q}) := dim(\mathbf{p}) + dim(\mathbf{t}) = N_{ctrl}N_{joint} + N_{tm}$	

**:dim-task** [method]

return  $\dim(\mathbf{e}) := \dim(\mathbf{e}^{kin}) + \dim(\mathbf{e}^{stat}) = 6N_{kin} + 4N_{joint}$

**:control-vector** [method]

return control vector  $\mathbf{p}$

**:timing-vector** [method]

return timing vector  $\mathbf{t}$

**:config-vector** [method]

return  $\mathbf{q} := \begin{pmatrix} \mathbf{p} \\ \mathbf{t} \end{pmatrix}$

**:set-control-vector** *control-vector-new* *ℰkey* (*relative?* *nil*) [method]

Set  $\mathbf{p}$ .

**:set-timing-vector** *timing-vector-new* *ℰkey* (*relative?* *nil*) [method]

Set  $\mathbf{t}$ .

**:set-config** *config-new* *ℰkey* (*relative?* *nil*) [method]

Set  $\mathbf{q}$ .

**:bspline-vector** *tm* *ℰkey* (*order-offset* 0) [method]

$$\mathbf{b}_n(t) := \begin{pmatrix} b_{0,n}(t) \\ b_{1,n}(t) \\ \vdots \\ b_{N_{ctrl}-1,n}(t) \end{pmatrix} \in \mathbb{R}^{N_{ctrl}} \quad (4.183)$$

return  $\mathbf{b}_n(t)$

**:bspline-matrix** *tm* *ℰkey* (*order-offset* 0) [method]

$$\mathbf{B}_n(t) := \begin{pmatrix} \mathbf{b}_n^T(t) & & & \mathbf{O} \\ & \mathbf{b}_n^T(t) & & \\ & & \ddots & \\ \mathbf{O} & & & \mathbf{b}_n^T(t) \end{pmatrix} \in \mathbb{R}^{N_{joint} \times N_{ctrl} N_{joint}} \quad (4.184)$$

return  $\mathbf{B}_n(t)$

**:differential-matrix** *ℰkey* (*diff-order* 1) [method]

$$\mathbf{D} := \frac{1}{h} \begin{pmatrix} 1 & -1 & & & \mathbf{O} \\ & 1 & -1 & & \\ & & \ddots & \ddots & \\ & & & \ddots & -1 \\ \mathbf{O} & & & & 1 \end{pmatrix} \in \mathbb{R}^{N_{ctrl} \times N_{ctrl}} \quad (4.185)$$

return  $\mathbf{D}^k$

**:extended-differential-matrix** *ℰkey (diff-order 1)*

[method]

$$\hat{\mathbf{D}}_k := \begin{pmatrix} (\mathbf{D}^k)^T & & \mathbf{O} \\ & \ddots & \\ \mathbf{O} & & (\mathbf{D}^k)^T \end{pmatrix} \in \mathbb{R}^{N_{ctrl}N_{joint} \times N_{ctrl}N_{joint}} \quad (4.186)$$

return  $\hat{\mathbf{D}}_k$

**:bspline-differential-matrix** *tm ℰkey (diff-order 1)*

[method]

return  $\mathbf{B}_{n-k}(t)\hat{\mathbf{D}}_k \in \mathbb{R}^{N_{joint} \times N_{ctrl}N_{joint}}$

**:control-matrix**

[method]

$$\mathbf{P} := \begin{pmatrix} \mathbf{p}_1^T \\ \mathbf{p}_2^T \\ \vdots \\ \mathbf{p}_{n_{joint}}^T \end{pmatrix} \in \mathbb{R}^{N_{joint} \times N_{ctrl}} \quad (4.187)$$

return  $\mathbf{P}$

**:theta** *tm*

[method]

return  $\boldsymbol{\theta}(t) = \mathbf{B}_n(t)\mathbf{p}$  [rad][m]

**:theta-dot** *tm ℰkey (diff-order 1)*

[method]

return  $\boldsymbol{\theta}^{(k)}(t) = \frac{d^{(k)}\boldsymbol{\theta}(t)}{dt^{(k)}} = \mathbf{P}\mathbf{D}^k\mathbf{b}_{n-k}(t)$  [rad/s<sup>k</sup>][m/s<sup>k</sup>]

**:theta-dot-numerical** *tm ℰkey (diff-order 1)*

[method]

(delta-time 0.05)

return  $\boldsymbol{\theta}^{(k)}(t) = \frac{d^{(k)}\boldsymbol{\theta}(t)}{dt^{(k)}} = \frac{\boldsymbol{\theta}^{(k-1)}(t + \Delta t) - \boldsymbol{\theta}^{(k-1)}(t)}{\Delta t}$  [rad/s<sup>k</sup>][m/s<sup>k</sup>]

**:apply-theta-to-robot** *tm*

[method]

apply  $\boldsymbol{\theta}(t)$  to robot.

**:kin-target-coords-list**

[method]

$$\mathbf{T}_m^{kin-trg} = \{\mathbf{p}_l^{kin-trg}, \mathbf{R}_l^{kin-trg}\} \quad (l = 1, 2, \dots, N_{kin}) \quad (4.188)$$

return  $\mathcal{T}^{kin-trg} := \{T_1^{kin-trg}, T_2^{kin-trg}, \dots, T_{N_{kin}}^{kin-trg}\}$

**:kin-attention-coords-list**

[method]

$$\mathbf{T}_m^{kin-att} = \{\mathbf{p}_l^{kin-att}, \mathbf{R}_l^{kin-att}\} \quad (l = 1, 2, \dots, N_{kin}) \quad (4.189)$$

return  $\mathcal{T}^{kin-att} := \{T_1^{kin-att}, T_2^{kin-att}, \dots, T_{N_{kin}}^{kin-att}\}$

**:kin-start-time**

[method]

return  $t_s^{kin} := t_1^{kin-tm}$

**:kin-finish-time** [method]

return  $t_f^{kin} := t_{N_{kin}}^{kin-tm}$

**:motion-duration** [method]

return  $(t_{N_{kin}}^{kin-tm} - t_1^{kin-tm})$

**:kinematics-task-value**  $\mathcal{E}key$  (update?  $t$ ) [method]

$$\mathbf{e}^{kin}(\mathbf{q}) = \mathbf{e}^{kin}(\mathbf{p}, t) \quad (4.190)$$

$$= \begin{pmatrix} \mathbf{e}_1^{kin}(\mathbf{p}, t) \\ \mathbf{e}_2^{kin}(\mathbf{p}, t) \\ \vdots \\ \mathbf{e}_{N_{kin}}^{kin}(\mathbf{p}, t) \end{pmatrix} \quad (4.191)$$

$$\mathbf{e}_l^{kin}(\mathbf{p}, t) = \begin{pmatrix} \mathbf{p}_l^{kin-trg} - \mathbf{p}_l^{kin-att}(\mathbf{p}, t) \\ a(\mathbf{R}_l^{kin-trg} \mathbf{R}_l^{kin-att}(\mathbf{p}, t)^T) \end{pmatrix} \in \mathbb{R}^6 \quad (l = 1, 2, \dots, N_{kin}) \quad (4.192)$$

$a(\mathbf{R})$  は姿勢行列  $\mathbf{R}$  の等価角軸ベクトルを表す .

return  $\mathbf{e}^{kin}(\mathbf{q}) \in \mathbb{R}^{6N_{kin}}$

**:stationery-start-finish-task-value**  $\mathcal{E}key$  (update?  $t$ ) [method]

$$\mathbf{e}^{stat}(\mathbf{q}) = \mathbf{e}^{stat}(\mathbf{p}, t) \quad (4.193)$$

$$= \begin{pmatrix} \mathbf{e}_{sv}^{stat}(\mathbf{p}, t) \\ \mathbf{e}_{fv}^{stat}(\mathbf{p}, t) \\ \mathbf{e}_{sa}^{stat}(\mathbf{p}, t) \\ \mathbf{e}_{fa}^{stat}(\mathbf{p}, t) \end{pmatrix} \quad (4.194)$$

$$\mathbf{e}_{sv}^{stat}(\mathbf{p}, t) := \dot{\boldsymbol{\theta}}(t_s^{kin}) \quad (4.195)$$

$$\mathbf{e}_{fv}^{stat}(\mathbf{p}, t) := \dot{\boldsymbol{\theta}}(t_f^{kin}) \quad (4.196)$$

$$\mathbf{e}_{sa}^{stat}(\mathbf{p}, t) := \ddot{\boldsymbol{\theta}}(t_s^{kin}) \quad (4.197)$$

$$\mathbf{e}_{fa}^{stat}(\mathbf{p}, t) := \ddot{\boldsymbol{\theta}}(t_f^{kin}) \quad (4.198)$$

return  $\mathbf{e}^{stat}(\mathbf{q}) \in \mathbb{R}^{4N_{joint}}$

**:task-value**  $\mathcal{E}key$  (update?  $t$ ) [method]

return  $\mathbf{e}(\mathbf{q}) := \begin{pmatrix} \mathbf{e}^{kin}(\mathbf{q}) \\ k_{stat} \mathbf{e}^{stat}(\mathbf{q}) \end{pmatrix} \in \mathbb{R}^{6N_{kin} + 4N_{joint}}$

**:kinematics-task-jacobian-with-control-vector** [method]

式 (4.110) より , タスク関数  $\mathbf{e}^{kin}$  を制御点  $\mathbf{p}$  で微分したヤコビ行列は次式で得られる .

$$\frac{\partial \mathbf{e}^{kin}}{\partial \mathbf{p}} = \begin{pmatrix} \frac{\partial \mathbf{e}_1^{kin}}{\partial \mathbf{p}} \\ \frac{\partial \mathbf{e}_2^{kin}}{\partial \mathbf{p}} \\ \vdots \\ \frac{\partial \mathbf{e}_{N_{kin}}^{kin}}{\partial \mathbf{p}} \end{pmatrix} \quad (4.199)$$

$$\frac{\partial \mathbf{e}_l^{kin}}{\partial \mathbf{p}} = -\mathbf{J}^{kin-att}(\boldsymbol{\theta}(t_l^{kin-tm})) \mathbf{B}_n(t_l^{kin-tm}) \quad (l = 1, 2, \dots, N_{kin}) \quad (4.200)$$

$$\text{return } \frac{\partial \mathbf{e}^{kin}}{\partial \mathbf{p}} \in \mathbb{R}^{6N_{kin} \times N_{ctrl}N_{joint}}$$

**:kinematics-task-jacobian-with-timing-vector**

[method]

式 (4.117) より , タスク関数  $e^{kin}$  をタイミング  $t$  で微分したヤコビ行列は次式で得られる .

$$\frac{\partial \mathbf{e}^{kin}}{\partial \mathbf{t}} = \begin{pmatrix} \frac{\partial \mathbf{e}_1^{kin}}{\partial t} \\ \frac{\partial \mathbf{e}_2^{kin}}{\partial t} \\ \vdots \\ \frac{\partial \mathbf{e}_{N_{kin}}^{kin}}{\partial t} \end{pmatrix} \quad (4.201)$$

$\frac{\partial \mathbf{e}_i^{kin}}{\partial t}$  の  $i$  番目の列ベクトル  $\left[ \frac{\partial \mathbf{e}_i^{kin}}{\partial t} \right]_i \in \mathbb{R}^6$  は次式で表される ( $i = 1, 2, \dots, N_{tm}$ ) .

$$\left[ \frac{\partial \mathbf{e}_i^{kin}}{\partial t} \right]_i = \begin{cases} -\mathbf{J}^{kin-att}(\boldsymbol{\theta}(t_l^{kin-tm})) \mathbf{P} \mathbf{D} \mathbf{b}_{n-1}(t_l^{kin-tm}) & t_l^{kin-tm} \text{ and } t_i \text{ is identical} \\ \mathbf{0} & \text{otherwise} \end{cases} \quad (4.202)$$

$$\text{return } \frac{\partial \mathbf{e}^{kin}}{\partial \mathbf{t}} \in \mathbb{R}^{6N_{kin} \times N_{tm}}$$

**:stationery-start-finish-task-jacobian-with-control-vector**

[method]

式 (4.130) , 式 (4.131) , 式 (4.132) , 式 (4.133) より , タスク関数  $e^{stat}$  を制御点  $\mathbf{p}$  で微分したヤコビ行列は次式で得られる .

$$\frac{\partial \mathbf{e}^{stat}}{\partial \mathbf{p}} = \begin{pmatrix} \frac{\partial \mathbf{e}_{sv}^{stat}}{\partial \mathbf{p}} \\ \frac{\partial \mathbf{e}_{fv}^{stat}}{\partial \mathbf{p}} \\ \frac{\partial \mathbf{e}_{sa}^{stat}}{\partial \mathbf{p}} \\ \frac{\partial \mathbf{e}_{fa}^{stat}}{\partial \mathbf{p}} \end{pmatrix} \quad (4.203)$$

$$\frac{\partial \mathbf{e}_{sv}^{stat}(\mathbf{p}, \mathbf{t})}{\partial \mathbf{p}} = \mathbf{B}_{n-1}(t_s^{kin}) \hat{\mathbf{D}}_1 \quad (4.204)$$

$$\frac{\partial \mathbf{e}_{fv}^{stat}(\mathbf{p}, \mathbf{t})}{\partial \mathbf{p}} = \mathbf{B}_{n-1}(t_f^{kin}) \hat{\mathbf{D}}_1 \quad (4.205)$$

$$\frac{\partial \mathbf{e}_{sa}^{stat}(\mathbf{p}, \mathbf{t})}{\partial \mathbf{p}} = \mathbf{B}_{n-2}(t_s^{kin}) \hat{\mathbf{D}}_2 \quad (4.206)$$

$$\frac{\partial \mathbf{e}_{fa}^{stat}(\mathbf{p}, \mathbf{t})}{\partial \mathbf{p}} = \mathbf{B}_{n-2}(t_f^{kin}) \hat{\mathbf{D}}_2 \quad (4.207)$$

$$\text{return } \frac{\partial \mathbf{e}^{stat}}{\partial \mathbf{p}} \in \mathbb{R}^{4N_{joint} \times N_{ctrl}N_{joint}}$$

**:stationery-start-finish-task-jacobian-with-timing-vector**

[method]

式 (4.134) , 式 (4.135) , 式 (4.136) , 式 (4.137) より , タスク関数  $e^{stat}$  をタイミング  $t$  で微分したヤコビ行列は次式で得られる .

$$\frac{\partial \mathbf{e}^{stat}}{\partial \mathbf{t}} = \begin{pmatrix} \frac{\partial \mathbf{e}_{sv}^{stat}}{\partial t} \\ \frac{\partial \mathbf{e}_{fv}^{stat}}{\partial t} \\ \frac{\partial \mathbf{e}_{sa}^{stat}}{\partial t} \\ \frac{\partial \mathbf{e}_{fa}^{stat}}{\partial t} \end{pmatrix} \quad (4.208)$$

$\frac{\partial \mathbf{e}_x^{stat}}{\partial \mathbf{t}}$  の  $i$  番目の列ベクトル  $\left[ \frac{\partial \mathbf{e}_x^{stat}}{\partial \mathbf{t}} \right]_i \in \mathbb{R}^{N_{joint}}$  は次式で表される ( $x \in \{sv, fv, sa, fa\}, i = 1, 2, \dots, N_{tm}$ ) .

$$\left[ \frac{\partial \mathbf{e}_{sv}^{stat}(\mathbf{p}, \mathbf{t})}{\partial \mathbf{t}} \right]_i = \begin{cases} PD^2 \mathbf{b}_{n-2}(t_s^{kin}) & t_s^{kin} \text{ and } t_i \text{ is identical} \\ \mathbf{0} & \text{otherwise} \end{cases} \quad (4.209)$$

$$\left[ \frac{\partial \mathbf{e}_{fv}^{stat}(\mathbf{p}, \mathbf{t})}{\partial \mathbf{t}} \right]_i = \begin{cases} PD^2 \mathbf{b}_{n-2}(t_f^{kin}) & t_f^{kin} \text{ and } t_i \text{ is identical} \\ \mathbf{0} & \text{otherwise} \end{cases} \quad (4.210)$$

$$\left[ \frac{\partial \mathbf{e}_{sa}^{stat}(\mathbf{p}, \mathbf{t})}{\partial \mathbf{t}} \right]_i = \begin{cases} PD^3 \mathbf{b}_{n-3}(t_s^{kin}) & t_s^{kin} \text{ and } t_i \text{ is identical} \\ \mathbf{0} & \text{otherwise} \end{cases} \quad (4.211)$$

$$\left[ \frac{\partial \mathbf{e}_{fa}^{stat}(\mathbf{p}, \mathbf{t})}{\partial \mathbf{t}} \right]_i = \begin{cases} PD^3 \mathbf{b}_{n-3}(t_f^{kin}) & t_f^{kin} \text{ and } t_i \text{ is identical} \\ \mathbf{0} & \text{otherwise} \end{cases} \quad (4.212)$$

return  $\frac{\partial \mathbf{e}^{stat}}{\partial \mathbf{t}} \in \mathbb{R}^{4N_{joint} \times N_{tm}}$

**:task-jacobian**

[method]

$$\frac{\partial \mathbf{e}}{\partial \mathbf{q}} = \begin{matrix} 6N_{kin} & N_{ctrl}N_{joint} & N_{tm} \\ 4N_{joint} & \begin{pmatrix} \frac{\partial \mathbf{e}^{kin}}{\partial \mathbf{p}} & \frac{\partial \mathbf{e}^{kin}}{\partial \mathbf{t}} \\ k_{stat} \frac{\partial \mathbf{e}^{stat}}{\partial \mathbf{p}} & k_{stat} \frac{\partial \mathbf{e}^{stat}}{\partial \mathbf{t}} \end{pmatrix} \end{matrix} \quad (4.213)$$

return  $\frac{\partial \mathbf{e}}{\partial \mathbf{q}} = \mathbb{R}^{(6N_{kin} + 4N_{joint}) \times (N_{ctrl}N_{joint} + N_{tm})}$

**:theta-max-vector**  $\mathcal{E}_{key}$  (update? nil)

[method]

return  $\boldsymbol{\theta}_{max} \in \mathbb{R}^{N_{joint}}$

**:theta-min-vector**  $\mathcal{E}_{key}$  (update? nil)

[method]

return  $\boldsymbol{\theta}_{min} \in \mathbb{R}^{N_{joint}}$

**:theta-inequality-constraint-matrix**  $\mathcal{E}_{key}$  (update? nil)

[method]

式 (4.144) より , 関節角度上下制限約は次式で表される .

$$\hat{\mathbf{E}}\boldsymbol{\theta}_{min} \leq \mathbf{p} + \Delta\mathbf{p} \leq \hat{\mathbf{E}}\boldsymbol{\theta}_{max} \quad (4.214)$$

$$\Leftrightarrow \begin{pmatrix} \mathbf{I} \\ -\mathbf{I} \end{pmatrix} \Delta\mathbf{p} \geq \begin{pmatrix} \hat{\mathbf{E}}\boldsymbol{\theta}_{min} - \mathbf{p} \\ -\hat{\mathbf{E}}\boldsymbol{\theta}_{max} + \mathbf{p} \end{pmatrix} \quad (4.215)$$

$$\Leftrightarrow \mathbf{C}_\theta \Delta\mathbf{p} \geq \mathbf{d}_\theta \quad (4.216)$$

ただし ,

$$\hat{\mathbf{E}} := \begin{pmatrix} \mathbf{1}_{N_{ctrl}} & & \mathbf{0}_{N_{ctrl}} \\ & \mathbf{1}_{N_{ctrl}} & \\ & & \ddots \\ \mathbf{0}_{N_{ctrl}} & & \mathbf{1}_{N_{ctrl}} \end{pmatrix} \in \mathbb{R}^{N_{ctrl}N_{joint} \times N_{joint}} \quad (4.217)$$

$\mathbf{1}_{N_{ctrl}} \in \mathbb{R}^{N_{ctrl}}$  は全要素が 1 の  $N_{ctrl}$  次元ベクトルである .

return  $\mathbf{C}_\theta := \begin{pmatrix} \mathbf{I} \\ -\mathbf{I} \end{pmatrix} \in \mathbb{R}^{2N_{ctrl}N_{joint} \times N_{ctrl}N_{joint}}$

**:theta-inequality-constraint-vector**  $\mathcal{E}_{key}$  (update? t)

[method]

return  $\mathbf{d}_\theta := \begin{pmatrix} \hat{\mathbf{E}}\boldsymbol{\theta}_{min} - \mathbf{p} \\ -\hat{\mathbf{E}}\boldsymbol{\theta}_{max} + \mathbf{p} \end{pmatrix} \in \mathbb{R}^{2N_{ctrl}N_{joint}}$



**:velocity-max-vector**  $\mathcal{E}key$  (*update?* *nil*) [method]  
 return  $\mathbf{v}_{max} \in \mathbb{R}^{N_{joint}}$

**:velocity-inequality-constraint-matrix**  $\mathcal{E}key$  (*update?* *nil*) [method]  
 式 (4.152) より , 関節速度上下限制約は次式で表される .

$$-\hat{\mathbf{E}}\mathbf{v}_{max} \leq \hat{\mathbf{D}}_1(\mathbf{p} + \Delta\mathbf{p}) \leq \hat{\mathbf{E}}\mathbf{v}_{max} \quad (4.218)$$

$$\Leftrightarrow \begin{pmatrix} \hat{\mathbf{D}}_1 \\ -\hat{\mathbf{D}}_1 \end{pmatrix} \Delta\mathbf{p} \geq \begin{pmatrix} -\hat{\mathbf{E}}\mathbf{v}_{max} - \hat{\mathbf{D}}_1\mathbf{p} \\ -\hat{\mathbf{E}}\mathbf{v}_{max} + \hat{\mathbf{D}}_1\mathbf{p} \end{pmatrix} \quad (4.219)$$

$$\Leftrightarrow \mathbf{C}_{\dot{\theta}}\Delta\mathbf{p} \geq \mathbf{d}_{\dot{\theta}} \quad (4.220)$$

$$\text{return } \mathbf{C}_{\dot{\theta}} := \begin{pmatrix} \hat{\mathbf{D}}_1 \\ -\hat{\mathbf{D}}_1 \end{pmatrix} \in \mathbb{R}^{2N_{ctrl}N_{joint} \times N_{ctrl}N_{joint}}$$

**:velocity-inequality-constraint-vector**  $\mathcal{E}key$  (*update?* *t*) [method]

$$\text{return } \mathbf{d}_{\dot{\theta}} := \begin{pmatrix} -\hat{\mathbf{E}}\mathbf{v}_{max} - \hat{\mathbf{D}}_1\mathbf{p} \\ -\hat{\mathbf{E}}\mathbf{v}_{max} + \hat{\mathbf{D}}_1\mathbf{p} \end{pmatrix} \in \mathbb{R}^{2N_{ctrl}N_{joint}}$$

**:acceleration-max-vector**  $\mathcal{E}key$  (*update?* *nil*) [method]

$$\text{return } \mathbf{a}_{max} \in \mathbb{R}^{N_{joint}}$$

**:acceleration-inequality-constraint-matrix**  $\mathcal{E}key$  (*update?* *nil*) [method]

式 (4.154) より , 関節加速度上下限制約は次式で表される .

$$-\hat{\mathbf{E}}\mathbf{a}_{max} \leq \hat{\mathbf{D}}_2(\mathbf{p} + \Delta\mathbf{p}) \leq \hat{\mathbf{E}}\mathbf{a}_{max} \quad (4.221)$$

$$\Leftrightarrow \begin{pmatrix} \hat{\mathbf{D}}_2 \\ -\hat{\mathbf{D}}_2 \end{pmatrix} \Delta\mathbf{p} \geq \begin{pmatrix} -\hat{\mathbf{E}}\mathbf{a}_{max} - \hat{\mathbf{D}}_2\mathbf{p} \\ -\hat{\mathbf{E}}\mathbf{a}_{max} + \hat{\mathbf{D}}_2\mathbf{p} \end{pmatrix} \quad (4.222)$$

$$\Leftrightarrow \mathbf{C}_{\ddot{\theta}}\Delta\mathbf{p} \geq \mathbf{d}_{\ddot{\theta}} \quad (4.223)$$

$$\text{return } \mathbf{C}_{\ddot{\theta}} := \begin{pmatrix} \hat{\mathbf{D}}_2 \\ -\hat{\mathbf{D}}_2 \end{pmatrix} \in \mathbb{R}^{2N_{ctrl}N_{joint} \times N_{ctrl}N_{joint}}$$

**:acceleration-inequality-constraint-vector**  $\mathcal{E}key$  (*update?* *t*) [method]

$$\text{return } \mathbf{d}_{\ddot{\theta}} := \begin{pmatrix} -\hat{\mathbf{E}}\mathbf{a}_{max} - \hat{\mathbf{D}}_2\mathbf{p} \\ -\hat{\mathbf{E}}\mathbf{a}_{max} + \hat{\mathbf{D}}_2\mathbf{p} \end{pmatrix} \in \mathbb{R}^{2N_{ctrl}N_{joint}}$$

**:control-vector-inequality-constraint-matrix**  $\mathcal{E}key$  (*update?* *nil*) [method]

$$\begin{cases} \mathbf{C}_{\theta}\Delta\mathbf{p} \geq \mathbf{d}_{\theta} \\ \mathbf{C}_{\dot{\theta}}\Delta\mathbf{p} \geq \mathbf{d}_{\dot{\theta}} \\ \mathbf{C}_{\ddot{\theta}}\Delta\mathbf{p} \geq \mathbf{d}_{\ddot{\theta}} \end{cases} \quad (4.224)$$

$$\Leftrightarrow \begin{pmatrix} \mathbf{C}_{\theta} \\ \mathbf{C}_{\dot{\theta}} \\ \mathbf{C}_{\ddot{\theta}} \end{pmatrix} \Delta\mathbf{p} \geq \begin{pmatrix} \mathbf{d}_{\theta} \\ \mathbf{d}_{\dot{\theta}} \\ \mathbf{d}_{\ddot{\theta}} \end{pmatrix} \quad (4.225)$$

$$\Leftrightarrow \mathbf{C}_p\Delta\mathbf{p} \geq \mathbf{d}_p \quad (4.226)$$

$$\text{return } \mathbf{C}_p := \begin{pmatrix} \mathbf{C}_{\theta} \\ \mathbf{C}_{\dot{\theta}} \\ \mathbf{C}_{\ddot{\theta}} \end{pmatrix} \in \mathbb{R}^{N_{p-ineq} \times \dim(\mathbf{p})}$$

**:control-vector-inequality-constraint-vector**  $\mathcal{E}key$  (*update?* *t*) [method]

$$\text{return } \mathbf{d}_p := \begin{pmatrix} \mathbf{d}_\theta \\ \mathbf{d}_{\dot{\theta}} \\ \mathbf{d}_{\ddot{\theta}} \end{pmatrix} \in \mathbb{R}^{N_{p-ineq}}$$

**:timing-vector-inequality-constraint-matrix**  $\mathcal{E}_{key} \text{ (update? nil)}$  [method]

式 (4.159) より，タイミングが B スプラインの初期，終端時刻の間に含まれる制約は次式で表される．

$$t_s \mathbf{1} \leq \mathbf{t} + \Delta \mathbf{t} \leq t_f \mathbf{1} \quad (4.227)$$

$$\Leftrightarrow \begin{pmatrix} \mathbf{I} \\ -\mathbf{I} \end{pmatrix} \Delta \mathbf{t} \geq \begin{pmatrix} t_s \mathbf{1} - \mathbf{t} \\ -t_f \mathbf{1} + \mathbf{t} \end{pmatrix} \quad (4.228)$$

また，式 (4.165) より，タイミングの順序が入れ替わることを許容しない場合，その制約は次式で表される．

$$\mathbf{D}_{tm}(\mathbf{t} + \Delta \mathbf{t}) \geq \mathbf{0} \quad (4.229)$$

$$\Leftrightarrow \mathbf{D}_{tm} \Delta \mathbf{t} \geq -\mathbf{D}_{tm} \mathbf{t} \quad (4.230)$$

ただし，

$$\mathbf{D}_{tm} = \begin{pmatrix} -1 & 1 & & & \mathbf{O} \\ & -1 & 1 & & \\ & & & \ddots & \\ \mathbf{O} & & & & -1 & 1 \end{pmatrix} \in \mathbb{R}^{(N_{tm}-1) \times N_{tm}} \quad (4.231)$$

これらを合わせると，

$$\begin{pmatrix} \mathbf{I} \\ -\mathbf{I} \\ \mathbf{D}_{tm} \end{pmatrix} \Delta \mathbf{t} \geq \begin{pmatrix} t_s \mathbf{1} - \mathbf{t} \\ -t_f \mathbf{1} + \mathbf{t} \\ -\mathbf{D}_{tm} \mathbf{t} \end{pmatrix} \Leftrightarrow \mathbf{C}_t \Delta \mathbf{p} \geq \mathbf{d}_t \quad (4.232)$$

$$\text{return } \mathbf{C}_t := \begin{pmatrix} \mathbf{I} \\ -\mathbf{I} \\ \mathbf{D}_{tm} \end{pmatrix} \in \mathbb{R}^{(3N_{tm}-1) \times \dim(\mathbf{t})}$$

**:timing-vector-inequality-constraint-vector**  $\mathcal{E}_{key} \text{ (update? t)}$  [method]

$$\text{return } \mathbf{d}_t := \begin{pmatrix} t_s \mathbf{1} - \mathbf{t} \\ -t_f \mathbf{1} + \mathbf{t} \\ -\mathbf{D}_{tm} \mathbf{t} \end{pmatrix} \in \mathbb{R}^{(3N_{tm}-1)}$$

**:config-inequality-constraint-matrix**  $\mathcal{E}_{key} \text{ (update? nil)}$  [method]

$\text{(update-collision? nil)}$

$$\begin{cases} \mathbf{C}_p \Delta \mathbf{p} \geq \mathbf{d}_p \\ \mathbf{C}_t \Delta \mathbf{t} \geq \mathbf{d}_t \end{cases} \quad (4.233)$$

$$\Leftrightarrow \begin{pmatrix} \mathbf{C}_p & \mathbf{C}_t \end{pmatrix} \begin{pmatrix} \Delta \mathbf{p} \\ \Delta \mathbf{t} \end{pmatrix} \geq \begin{pmatrix} \mathbf{d}_p \\ \mathbf{d}_t \end{pmatrix} \quad (4.234)$$

$$\Leftrightarrow \mathbf{C} \Delta \mathbf{q} \geq \mathbf{d} \quad (4.235)$$

$$\text{return } \mathbf{C} := \begin{pmatrix} \mathbf{C}_p \\ \mathbf{C}_t \end{pmatrix} \in \mathbb{R}^{N_{ineq} \times \dim(\mathbf{q})}$$



$$\mathbf{v}_{reg,p} := (k_{sqr,1} \mathbf{H}_{sqr,1} + k_{sqr,2} \mathbf{H}_{sqr,2} + k_{sqr,3} \mathbf{H}_{sqr,3}) \mathbf{p} \quad (4.241)$$

return  $\mathbf{v}_{reg,p} \in \mathbb{R}^{dim(\mathbf{p})}$

**:motion-duration-regular-matrix**

[method]

式 (4.180) より, 動作期間の二乗は次式で得られる .

$$F_{duration}(\mathbf{t}) = |t_1 - t_{N_{tm}}|^2 \quad (4.242)$$

$$= \mathbf{t}^T \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \mathbf{t} \quad (4.243)$$

$$= \mathbf{t}^T \mathbf{H}_{duration} \mathbf{t} \quad (4.244)$$

これは二次形式の正則化項である .

return  $\mathbf{H}_{duration} \in \mathbb{R}^{dim(\mathbf{t}) \times dim(\mathbf{t})}$

**:timing-vector-regular-matrix**

[method]

$$\mathbf{W}_{reg,t} := \min(k_{max,t}, \|\mathbf{e}\|^2 + k_{off,t}) \mathbf{I} + k_{duration} \mathbf{H}_{duration} \quad (4.245)$$

return  $\mathbf{W}_{reg,t} \in \mathbb{R}^{dim(\mathbf{t}) \times dim(\mathbf{t})}$

**:timing-vector-regular-vector**

[method]

$$\mathbf{v}_{reg,t} := k_{duration} \mathbf{H}_{duration} \mathbf{t} \quad (4.246)$$

return  $\mathbf{v}_{reg,t} \in \mathbb{R}^{dim(\mathbf{t})}$

**:regular-matrix**

[method]

$$\mathbf{W}_{reg} := \begin{pmatrix} \mathbf{W}_{reg,p} & \\ & \mathbf{W}_{reg,t} \end{pmatrix} \quad (4.247)$$

return  $\mathbf{W}_{reg} \in \mathbb{R}^{dim(\mathbf{q}) \times dim(\mathbf{q})}$

**:regular-vector**

[method]

$$\mathbf{v}_{reg} := \begin{pmatrix} \mathbf{v}_{reg,p} \\ \mathbf{v}_{reg,t} \end{pmatrix} \quad (4.248)$$

return  $\mathbf{v}_{reg} \in \mathbb{R}^{dim(\mathbf{q}) \times dim(\mathbf{q})}$

**:update-collision-inequality-constraint**

[method]

Not implemented yet.

**:update-viewer** *key (trajectory-delta-time (/ (- \_finish-time \_start-time) 10.0))*

[method]

Update viewer.

**:print-status** [method]

Print status.

**:print-motion-information** [method]

Print motion information.

**:play-animation** *ℰkey (robot)* [method]

*(delta-time (/ (- \_finish-time \_start-time) 100.0))*

*(only-motion-duration? t)*

*(loop? t)*

*(visualize-callback-func)*

Play motion animation.

**:plot-theta-graph** *ℰkey (joint-id nil)* [method]

*(divide-num 200)*

*(plot-numerical? nil)*

*(only-motion-duration? t)*

*(dat-filename /tmp/bspline-configuration-task-plot-theta-graph.dat)*

*(dump-pdf? nil)*

*(dump-filename (ros::resolve-ros-path package://eus-qp/optmotiongen/logs/bspline-configuration-task-plot-theta-graph.pdf))*

Plot graph.

**:generate-angle-vector-sequence** *ℰkey (divide-num 100)* [method]

Generate angle-vector-sequence.

**get-bspline-knot** *i n m x\_min x\_max h* [function]

$$t_i = \frac{i-n}{m-n}(t_f - t_s) + t_s \quad (4.249)$$

$$= hi + \frac{mt_s - nt_f}{m-n} \quad (4.250)$$

return knot  $t_i$  for B-spline function

**bspline-basis-func** *x i n m x\_min x\_max ℰoptional (n-orig n) (m-orig m)* [function]

$$b_{i,0}(t) = \begin{cases} 1 & \text{if } t_i \leq t < t_{i+1} \\ 0 & \text{otherwise} \end{cases} \quad (4.251)$$

$$b_{i,n}(t) = \frac{(t - t_i)b_{i,n-1}(t) + (t_{i+n+1} - t)b_{i+1,n-1}(t)}{nh} \quad (4.252)$$

return B-spline function value  $b_{i,n}(t)$ .

### 4.3 B スプラインを用いた動的動作の生成

**bspline-trajectory** [class]

:super **propertied-object**

:slots (*\_start-time t<sub>s</sub>*)

```

(_finish-time  $t_f$ )
(_num-control-point  $N_{ctrl}$ )
(_bspline-order B-spline order,  $n$ )
(_dim-instant-config  $\dim(\bar{\mathbf{q}}(t))$ )
(_dim-control-vector  $\dim(\mathbf{p}) := N_{ctrl} \dim(\bar{\mathbf{q}})$ )
(_control-vector  $\mathbf{p}$ )
(_bspline-vector buffer for  $\mathbf{b}_n(t)$ )
(_bspline-matrix buffer for  $\mathbf{B}_n(t)$ )
(_diff-mat buffer for  $\mathbf{D}^k$ )
(_diff-mat-list buffer for  $\{\mathbf{D}^1, \mathbf{D}^2, \dots, \mathbf{D}^K\}$ )
(_extended-diff-mat-list buffer for  $\{\hat{\mathbf{D}}_1, \hat{\mathbf{D}}_2, \dots, \hat{\mathbf{D}}_K\}$ )

```

B スプラインを利用した軌道のクラス .

B スプラインベクトル・行列, 制御点ベクトル・ベクトル, 微分行列, 瞬時コンフィギュレーションの取得や制御点ベクトルの更新のためのメソッドが定義されている .

B スプライン軌道を定めるために, 初期化時に以下を与える

**start-time**  $t_s$  初期時刻

**finish-time**  $t_f$  終端時刻

**num-control-point**  $N_{ctrl}$  制御点の個数

**bspline-order**  $n$  B スプラインのオーダー

**dim-instant-config**  $\dim(\bar{\mathbf{q}}(t))$  瞬時コンフィギュレーションの次元

```

:init  $\mathcal{E}_{key}$  ( $name$ ) [method]
    ( $start-time$  0.0)
    ( $finish-time$  10.0)
    ( $num-control-point$  10)
    ( $bspline-order$  3)
    ( $dim-instant-config$  1)
    Initialize instance

:start-time [method]
    return  $t_s$ 

:finish-time [method]
    return  $t_s$ 

:num-control-point [method]
    return  $N_{ctrl}$ 

:dim-instant-config [method]
    return  $\dim(\bar{\mathbf{q}}(t))$ 

:dim-control-vector [method]
    return  $\dim(\mathbf{p}) := N_{ctrl} \dim(\bar{\mathbf{q}})$ 

:bspline-vector  $tm$   $\mathcal{E}_{key}$  ( $order-offset$  0) [method]

```

$$\mathbf{b}_n(t) := \begin{pmatrix} b_{0,n}(t) \\ b_{1,n}(t) \\ \vdots \\ b_{N_{ctrl}-1,n}(t) \end{pmatrix} \in \mathbb{R}^{N_{ctrl}} \quad (4.253)$$

return  $\mathbf{b}_n(t)$

**:bspline-matrix** *tm* *ℰkey* (order-offset 0) [method]

$$\mathbf{B}_n(t) := \begin{pmatrix} \mathbf{b}_n^T(t) & & & \mathbf{O} \\ & \mathbf{b}_n^T(t) & & \\ & & \ddots & \\ \mathbf{O} & & & \mathbf{b}_n^T(t) \end{pmatrix} \in \mathbb{R}^{dim(\bar{\mathbf{q}}) \times N_{ctrl} dim(\bar{\mathbf{q}})} \quad (4.254)$$

return  $\mathbf{B}_n(t)$

**:control-vector** [method]

$$\mathbf{p} := \begin{pmatrix} \mathbf{p}_1 \\ \vdots \\ \mathbf{p}_{dim(\bar{\mathbf{q}})} \end{pmatrix} \in \mathbb{R}^{N_{ctrl} dim(\bar{\mathbf{q}})} \quad (4.255)$$

return  $\mathbf{p}$

**:control-matrix** [method]

$$\mathbf{P} := \begin{pmatrix} \mathbf{p}_1^T \\ \vdots \\ \mathbf{p}_{dim(\bar{\mathbf{q}}}^T \end{pmatrix} \in \mathbb{R}^{dim(\bar{\mathbf{q}}) \times N_{ctrl}} \quad (4.256)$$

return  $\mathbf{P}$

**:differential-matrix** *ℰkey* (diff-order 1) [method]

$$\mathbf{D} := \frac{1}{h} \begin{pmatrix} 1 & -1 & & & \mathbf{O} \\ & 1 & -1 & & \\ & & \ddots & \ddots & \\ & & & \ddots & -1 \\ \mathbf{O} & & & & 1 \end{pmatrix} \in \mathbb{R}^{N_{ctrl} \times N_{ctrl}} \quad (4.257)$$

return  $\mathbf{D}^k$

**:extended-differential-matrix** *ℰkey* (diff-order 1) [method]

$$\hat{\mathbf{D}}_k := \begin{pmatrix} (\mathbf{D}^k)^T & & \mathbf{O} \\ & \ddots & \\ \mathbf{O} & & (\mathbf{D}^k)^T \end{pmatrix} \in \mathbb{R}^{N_{ctrl} dim(\bar{\mathbf{q}}) \times N_{ctrl} dim(\bar{\mathbf{q}})} \quad (4.258)$$

return  $\hat{D}_k$

**:instant-config** *tm*

[method]

$$\text{return } \bar{\mathbf{q}}(t) = \begin{pmatrix} \mathbf{p}_1^T \mathbf{b}_n(t) \\ \vdots \\ \mathbf{p}_{\dim(\bar{\mathbf{q}})}^T \mathbf{b}_n(t) \end{pmatrix} = \mathbf{P} \mathbf{b}_n(t) = \mathbf{B}_n(t) \mathbf{p} \in \mathbb{R}^{\dim(\bar{\mathbf{q}})}$$

**:instant-config-dot** *tm* *ℰkey* (*diff-order 1*)

[method]

$$\text{return } \bar{\mathbf{q}}^{(k)}(t) = \frac{d^{(k)} \bar{\mathbf{q}}(t)}{dt^{(k)}} = \mathbf{P} \mathbf{D}^k \mathbf{b}_{n-k}(t)$$

**:set-control-vector** *control-vector-new* *ℰkey* (*relative? nil*)

[method]

Set  $\mathbf{p} \in \mathbb{R}^{N_{ctrl} \dim(\bar{\mathbf{q}})}$ .

**bspline-dynamic-configuration-task**

[class]

**:super** **propertied-object**

**:slots**

(\_robot-env robot-environment instance)

(\_theta-bst bspline trajectory instance for  $\boldsymbol{\theta}$ )

(\_cog-bst bspline trajectory instance for  $\mathbf{c}$ )

(\_ang-moment-bst bspline trajectory instance for  $\mathbf{L}$ )

(\_wrench-bst bspline trajectory instance for  $\hat{\mathbf{w}}$ )

(\_torque-bst bspline trajectory instance for  $\boldsymbol{\tau}$ )

(\_phi-vector  $\boldsymbol{\phi}$ )

(\_num-kin  $N_{kin} := |\mathcal{T}^{kin-trg}| = |\mathcal{T}^{kin-att}|$ )

(\_num-contact  $N_{cnt} := |\mathcal{T}^{cnt-trg}| = |\mathcal{T}^{cnt-att}|$ )

(\_num-variant-joint  $N_{var-joint} := |\mathcal{J}_{var}|$ )

(\_num-invariant-joint  $N_{invar-joint} := |\mathcal{J}_{invar}|$ )

(\_num-drive-joint  $N_{drive-joint} := |\mathcal{J}_{drive}|$ )

(\_num-posture-joint  $N_{posture-joint} := |\mathcal{J}_{posture}|$ )

(\_num-collision  $N_{col} :=$  number of collision check pairs)

(\_dim-theta-control-vector  $\dim(\mathbf{p}_\theta) := N_{var-joint} N_{\theta-ctrl}$ )

(\_dim-cog-control-vector  $\dim(\mathbf{p}_c) := 3N_{c-ctrl}$ )

(\_dim-ang-moment-control-vector  $\dim(\mathbf{p}_L) := 3N_{L-ctrl}$ )

(\_dim-wrench-control-vector  $\dim(\mathbf{p}_{\hat{w}}) := 6N_{cnt} N_{\hat{w}-ctrl}$ )

(\_dim-torque-control-vector  $\dim(\mathbf{p}_\tau) := N_{drive-joint} N_{\tau-ctrl}$ )

(\_dim-phi  $\dim(\boldsymbol{\phi}) := N_{invar-joint}$ )

(\_dim-config  $\dim(\mathbf{q}) := \dim(\mathbf{p}_\theta) + \dim(\mathbf{p}_c) + \dim(\mathbf{p}_L) + \dim(\mathbf{p}_{\hat{w}}) + \dim(\mathbf{p}_\tau) + \dim(\boldsymbol{\phi})$ )

(\_dim-kin-task  $\dim(\mathbf{e}^{kin})$ )

(\_dim-eom-trans-task  $\dim(\mathbf{e}^{eom-trans})$ )

(\_dim-eom-rot-task  $\dim(\mathbf{e}^{eom-rot})$ )

(\_dim-cog-task  $\dim(\mathbf{e}^{cog})$ )

(\_dim-ang-moment-task  $\dim(\mathbf{e}^{ang-moment})$ )

(\_dim-torque-task  $\dim(\mathbf{e}^{trq})$ )

(\_dim-posture-task  $\dim(\mathbf{e}^{posture})$ )

(\_dim-task  $\dim(\mathbf{e})$ )

(\_kin-scale-mat-list-func function returning list of  $K_{kin}$ )

(\_target-posture-scale  $k_{posture}$ )

(\_torque-regular-scale  $k_{trq}$ )



```

(_stationery-start-finish-task-scale  $k_{stat}$ )
(_first-diff-square-integration-regular-scale  $k_{sqr,1}$ )
(_second-diff-square-integration-regular-scale  $k_{sqr,2}$ )
(_third-diff-square-integration-regular-scale  $k_{sqr,3}$ )
(_norm-regular-scale-max  $k_{max}$ )
(_norm-regular-scale-offset  $k_{off}$ )
(_variant-joint-list  $\mathcal{J}_{var}$ )
(_invariant-joint-list  $\mathcal{J}_{invar}$ )
(_drive-joint-list  $\mathcal{J}_{drive}$ )
(_posture-joint-list  $\mathcal{J}_{posture}$ )
(_kin-time-list time list for kinematics task)
(_eom-time-list time list for eom task)
(_kin-target-coords-list-func function returning  $\mathcal{T}^{kin-trg}$ )
(_kin-attention-coords-list-func function returning  $\mathcal{T}^{kin-att}$ )
(_contact-target-coords-list-func function returning  $\mathcal{T}^{cnt-trg}$ )
(_contact-attention-coords-list-func function returning  $\mathcal{T}^{cnt-att}$ )
(_contact-constraint-list-func function returning list of contact-constraint)
(_posture-joint-angle-list  $\bar{\theta}^{trg}$ )
(_variant-joint-angle-margin margin of  $\theta$  [deg] [mm])
(_invariant-joint-angle-margin margin of  $\phi$  [deg] [mm])
(_collision-pair-list list of bodyset-link or body pair)
(_collision-distance-margin-list list of collision distance margin)
(_task-jacobi buffer for  $\frac{\partial e}{\partial q}$ )
(_collision-theta-inequality-constraint-matrix buffer for  $C_{col,\theta}$ )
(_collision-phi-inequality-constraint-matrix buffer for  $C_{col,\phi}$ )
(_collision-inequality-constraint-vector buffer for  $C_{col}$ )

```

B スプラインを利用した動的動作生成のためのコンフィギュレーション  $q$  とタスク関数  $e(q)$  のクラス .  
 コンフィギュレーション  $q$  の取得・更新, タスク関数  $e(q)$  の取得, タスク関数のヤコビ行列  $\frac{\partial e(q)}{\partial q}$  の取得,  
 コンフィギュレーションの等式・不等式制約  $A, b, C, d$  の取得のためのメソッドが定義されている .  
 コンフィギュレーション・タスク関数を定めるために, 初期化時に以下を与える

- ロボット

robot ロボットのインスタンス

joint-list  $\mathcal{J}$  関節

- B スプラインのパラメータ

start-time  $t_s$  B スプラインの使用区間の初期時刻

finish-time  $t_f$  B スプラインの使用区間の終端時刻

num-control-point  $N_{ctrl}$  制御点の個数

bspline-order  $n$  B スプラインの次数

- 幾何拘束

kin-target-coords-list  $\mathcal{T}^{kin-trg}$  幾何到達目標位置姿勢リスト

kin-attention-coords-list  $\mathcal{T}^{kin-att}$  幾何到達着目位置姿勢リスト

kin-time-list  $\{t_1^{kin-tm}, t_2^{kin-tm}, \dots, t_{N_{kin}}^{kin-tm}\}$  幾何到達タイミングリスト

**kin-variable-timing-list** 幾何到達タイミングが可変か (t) , 固定か (nil) のリスト . このリスト内の t の個数がタイミング t の次元  $N_{tm}$  となる .

動的動作は各瞬間において以下の瞬時状態  $\bar{q}(t)$  を定めることで表現される .

$$\bar{q}(t) := \begin{pmatrix} \theta(t) \\ c(t) \\ L(t) \\ \hat{w}(t) \\ \tau(t) \\ \phi \end{pmatrix} \quad (4.259)$$

$\theta \in \mathbb{R}^{N_{var-joint}}$  時変関節位置 [rad] [m]

$c \in \mathbb{R}^3$  重心位置 [m]

$L \in \mathbb{R}^3$  角運動量 [kgm<sup>2</sup>/s]

$\hat{w} \in \mathbb{R}^{6N_{cnt}}$  接触レンチ [N] [Nm]

$\tau \in \mathbb{R}^{N_{drive-joint}}$  関節トルク [Nm] [N]

$\phi \in \mathbb{R}^{N_{invar-joint}}$  時不変関節位置 [rad] [m]

$\hat{w}$  は次式のように , 全接触部位でのワールド座標系での力・モーメントを並べたベクトルである .

$$\hat{w} = \begin{pmatrix} w_1^T & w_2^T & \cdots & w_{N_{cnt}}^T \end{pmatrix}^T \quad (4.260)$$

$$= \begin{pmatrix} f_1^T & n_1^T & f_2^T & n_2^T & \cdots & f_{N_{cnt}}^T & n_{N_{cnt}}^T \end{pmatrix}^T \quad (4.261)$$

瞬時状態  $\bar{q}(t)$  の軌道を B スプラインで表現することで , 設計対称のコンフィギュレーション  $q$  は以下から構成される .

$$q := \begin{pmatrix} p_\theta \\ p_c \\ p_L \\ p_{\hat{w}} \\ p_\tau \\ \phi \end{pmatrix} \quad (4.262)$$

$p_\theta \in \mathbb{R}^{N_{var-joint} N_{\theta-ctrl}}$  時変関節位置の制御点 [rad] [m]

$p_c \in \mathbb{R}^{3N_{c-ctrl}}$  重心位置の制御点 [m]

$p_L \in \mathbb{R}^{3N_{L-ctrl}}$  角運動量の制御点 [kgm<sup>2</sup>/s]

$p_{\hat{w}} \in \mathbb{R}^{6N_{cnt} N_{\hat{w}-ctrl}}$  接触レンチの制御点 [N] [Nm]

$p_\tau \in \mathbb{R}^{N_{drive-joint} N_{\tau-ctrl}}$  関節トルクの制御点 [Nm] [N]

$\phi \in \mathbb{R}^{N_{invar-joint}}$  時不変関節位置 [rad] [m]

制御点とは , B スプライン基底関数の重み係数を意味する .

瞬時状態  $\bar{q}(t)$  に関するタスク関数  $\bar{e}(\bar{q}(t))$  は以下から構成される .

$$\bar{e}(\bar{q}) = \begin{pmatrix} \bar{e}^{kin}(\theta, \phi) \\ \bar{e}^{eom-trans}(c, \hat{w}) \\ \bar{e}^{eom-rot}(\theta, c, L, \hat{w}, \phi) \\ \bar{e}^{cog}(\theta, c, \phi) \\ \bar{e}^{ang-moment}(\theta, L, \phi) \\ \bar{e}^{trq}(\theta, \hat{w}, \tau, \phi) \\ \bar{e}^{posture}(\theta) \end{pmatrix} \quad (4.263)$$

$\bar{e}^{kin}(\theta, \phi) \in \mathbb{R}^{6N_{kin}}$  幾何到達拘束 [rad] [m]

$$\bar{e}^{kin}(\theta, \phi) = \begin{pmatrix} \bar{e}_1^{kin}(\theta, \phi) \\ \bar{e}_2^{kin}(\theta, \phi) \\ \vdots \\ \bar{e}_{N_{kin}}^{kin}(\theta, \phi) \end{pmatrix} \quad (4.264)$$

$$\bar{e}_m^{kin}(\theta, \phi) = \begin{pmatrix} \mathbf{p}_m^{kin-trg}(\theta, \phi) - \mathbf{p}_m^{kin-att}(\theta, \phi) \\ \mathbf{a} \left( \mathbf{R}_m^{kin-trg}(\theta, \phi) \mathbf{R}_m^{kin-att}(\theta, \phi)^T \right) \end{pmatrix} \in \mathbb{R}^6 \quad (m = 1, 2, \dots, N_{kin}) \quad (4.265)$$

$\mathbf{a}(\mathbf{R})$  は姿勢行列  $\mathbf{R}$  の等価角軸ベクトルを表す .

$\bar{e}^{eom-trans}(\mathbf{c}, \hat{\mathbf{w}}) \in \mathbb{R}^3$  並進運動方程式 [kg m/s<sup>2</sup>]

$$\bar{e}^{eom-trans}(\mathbf{c}, \hat{\mathbf{w}}) = m\ddot{\mathbf{c}} - \left\{ \sum_{m=1}^{N_{cnt}} \mathbf{f}_m - m\mathbf{g} + \sum_{m=1}^{N_{ex}} \mathbf{f}_m^{ex} \right\} \quad (4.266)$$

$\bar{e}^{eom-rot}(\theta, \mathbf{c}, \mathbf{L}, \hat{\mathbf{w}}, \phi) \in \mathbb{R}^3$  回転運動方程式 [kg m<sup>2</sup>/s<sup>2</sup>]

$$\begin{aligned} \bar{e}^{eom-rot}(\theta, \mathbf{c}, \mathbf{L}, \hat{\mathbf{w}}, \phi) = \dot{\mathbf{L}} - & \left( \sum_{m=1}^{N_{cnt}} \{ (\mathbf{p}_m^{cnt-trg}(\theta, \phi) - \mathbf{c}) \times \mathbf{f}_m + \mathbf{n}_m \} \right. \\ & \left. + \sum_{m=1}^{N_{ex}} \{ (\mathbf{p}_m^{ex}(\theta, \phi) - \mathbf{c}) \times \mathbf{f}_m^{ex} + \mathbf{n}_m^{ex} \} \right) \end{aligned} \quad (4.267)$$

$\bar{e}^{cog}(\theta, \mathbf{c}, \phi) \in \mathbb{R}^3$  重心位置 [m]

$$\bar{e}^{cog}(\theta, \mathbf{c}, \phi) = \mathbf{p}_G(\theta, \phi) - \mathbf{c} \quad (4.268)$$

$\bar{e}^{ang-moment}(\theta, \mathbf{L}, \phi) \in \mathbb{R}^3$  角運動量 [kg m<sup>2</sup>/s]

$$\bar{e}^{ang-moment}(\theta, \mathbf{L}, \phi) = \mathbf{L} - \left\{ \mathbf{A}_\theta(\theta, \phi) \dot{\boldsymbol{\theta}} + \mathbf{A}_\phi(\theta, \phi) \dot{\boldsymbol{\phi}} \right\} \quad (4.269)$$

$\bar{e}^{trq}(\theta, \hat{\mathbf{w}}, \boldsymbol{\tau}, \phi) \in \mathbb{R}^{N_{drive-joint}}$  関節トルクの釣り合い [rad] [m]

$$\begin{aligned} \bar{e}^{trq}(\theta, \hat{\mathbf{w}}, \boldsymbol{\tau}, \phi) = & \sum_{m=1}^{N_{cnt}} \{ (\mathbf{p}_m^{cnt-trg}(\theta, \phi) - \mathbf{c}) \times \mathbf{f}_m + \mathbf{n}_m \} \\ & + \sum_{m=1}^{N_{ex}} \{ (\mathbf{p}_m^{ex}(\theta, \phi) - \mathbf{c}) \times \mathbf{f}_m^{ex} + \mathbf{n}_m^{ex} \} \end{aligned} \quad (4.270)$$

$\bar{e}^{posture}(\boldsymbol{\theta}) \in \mathbb{R}^{N_{posture-joint}}$  関節角目標 [rad] [m]

$$\bar{e}^{posture}(\boldsymbol{\theta}) = \left( \bar{\boldsymbol{\theta}}^{trg} - \bar{\boldsymbol{\theta}} \right) \quad (4.271)$$

瞬時状態  $\bar{\mathbf{q}}(t)$  の軌道を B スプラインで表現することで , 設計対称のコンフィギュレーション  $\mathbf{q}$  に関するタスク関数  $e(\mathbf{q})$  は以下から構成される .

$$e(\mathbf{q}) = \begin{pmatrix} e^{kin}(\mathbf{p}_\theta, \phi) \\ e^{eom-trans}(\mathbf{p}_c, \mathbf{p}_{\hat{\mathbf{w}}}) \\ e^{eom-rot}(\mathbf{p}_\theta, \mathbf{p}_c, \mathbf{p}_L, \mathbf{p}_{\hat{\mathbf{w}}}, \phi) \\ e^{cog}(\mathbf{p}_\theta, \mathbf{p}_c, \phi) \\ e^{ang-moment}(\mathbf{p}_\theta, \mathbf{p}_L, \phi) \\ e^{trq}(\mathbf{p}_\theta, \mathbf{p}_{\hat{\mathbf{w}}}, \mathbf{p}_\tau, \phi) \\ e^{posture}(\mathbf{p}_\theta) \end{pmatrix} \quad (4.272)$$

ただし ,

$$\mathbf{e}^*(\mathbf{q}) = \begin{pmatrix} \bar{\mathbf{e}}^*(\bar{\mathbf{q}}(t_1)) \\ \vdots \\ \bar{\mathbf{e}}^*(\bar{\mathbf{q}}(t_{N_{tm}})) \end{pmatrix} \in \mathbb{R}^{N_{tm} \dim(\bar{\mathbf{e}}^*(\bar{\mathbf{q}})(t))} \quad (4.273)$$

**:init** *key* (*name*) [method]

```
(robot-env)
(variant-joint-list (send robot-env :variant-joint-list))
(invariant-joint-list (send robot-env :invariant-joint-list))
(drive-joint-list (send robot-env :drive-joint-list))
(posture-joint-list)
(kin-time-list)
(eom-time-list)
(theta-bst)
(cog-bst)
(ang-moment-bst)
(wrench-bst)
(torque-bst)
(kin-target-coords-list-func)
(kin-attention-coords-list-func)
(contact-target-coords-list-func)
(contact-attention-coords-list-func)
(contact-constraint-list-func)
(posture-joint-angle-list)
(variant-joint-angle-margin 3.0)
(invariant-joint-angle-margin 3.0)
(collision-pair-list)
(collision-distance-margin 0.01)
(collision-distance-margin-list)
(kin-scale-mat-list-func)
(target-posture-scale 0.001)
(torque-regular-scale 1.000000e-04)
(stationery-start-finish-task-scale 0.0)
(first-diff-square-integration-regular-scale 0.0)
(second-diff-square-integration-regular-scale 0.0)
(third-diff-square-integration-regular-scale 0.0)
(norm-regular-scale-max 1.000000e-05)
(norm-regular-scale-offset 1.000000e-07)
```

Initialize instance

**:robot-env** [method]

return robot-environment instance

**:variant-joint-list** [method]

return  $\mathcal{J}_{var}$

**:invariant-joint-list** [method]

return $\mathcal{J}_{invar}$	
<b>:drive-joint-list</b>	[method]
return $\mathcal{J}_{drive}$	
<b>:num-kin</b>	[method]
return $N_{kin} :=  \mathcal{T}^{kin-trg}  =  \mathcal{T}^{kin-att} $	
<b>:num-contact</b>	[method]
return $N_{cnt} :=  \mathcal{T}^{cnt-trg}  =  \mathcal{T}^{cnt-att} $	
<b>:num-variant-joint</b>	[method]
return $N_{var-joint} :=  \mathcal{J}_{var} $	
<b>:num-invariant-joint</b>	[method]
return $N_{invar-joint} :=  \mathcal{J}_{invar} $	
<b>:num-drive-joint</b>	[method]
return $N_{drive-joint} :=  \mathcal{J}_{drive} $	
<b>:num-posture-joint</b>	[method]
return $N_{target-joint} :=  \mathcal{J}_{target} $	
<b>:num-collision</b>	[method]
return $N_{col} :=$ number of collision check pairs	
<b>:dim-config</b>	[method]
return $dim(\mathbf{q})$	
<b>:dim-task</b>	[method]
return $dim(\mathbf{e})$	
<b>:config-vector</b>	[method]
return $\mathbf{q}$	
<b>:set-theta-control-vector</b> <i>control-vector-new</i> <i>ℰkey</i> ( <i>relative?</i> <i>nil</i> )	[method]
Set $\mathbf{p}_\theta$ .	
<b>:set-cog-control-vector</b> <i>control-vector-new</i> <i>ℰkey</i> ( <i>relative?</i> <i>nil</i> )	[method]
Set $\mathbf{p}_c$ .	
<b>:set-ang-moment-control-vector</b> <i>control-vector-new</i> <i>ℰkey</i> ( <i>relative?</i> <i>nil</i> )	[method]
Set $\mathbf{p}_L$ .	
<b>:set-wrench-control-vector</b> <i>control-vector-new</i> <i>ℰkey</i> ( <i>relative?</i> <i>nil</i> )	[method]
Set $\mathbf{p}_{\hat{w}}$ .	
<b>:set-torque-control-vector</b> <i>control-vector-new</i> <i>ℰkey</i> ( <i>relative?</i> <i>nil</i> )	[method]
Set $\mathbf{p}_\tau$ .	
<b>:set-phi</b> <i>phi-new</i> <i>ℰkey</i> ( <i>relative?</i> <i>nil</i> )	[method]
Set $\phi$ .	
<b>:set-config</b> <i>config-new</i> <i>ℰkey</i> ( <i>relative?</i> <i>nil</i> )	[method]

Set  $\mathbf{q}$ .

**:theta**  $tm$   $\mathcal{E}key$  (diff-order 0) [method]  
 return  $\boldsymbol{\theta}(t)$  [rad] [m]

**:cog**  $tm$   $\mathcal{E}key$  (diff-order 0) [method]  
 return  $\mathbf{c}(t)$  [m]

**:ang-moment**  $tm$   $\mathcal{E}key$  (diff-order 0) [method]  
 return  $\mathbf{L}(t)$  [kgm<sup>2</sup>/s]

**:wrench**  $tm$   $\mathcal{E}key$  (diff-order 0) [method]  
 return  $\hat{\mathbf{w}}(t)$  [N] [Nm]

**:torque**  $tm$   $\mathcal{E}key$  (diff-order 0) [method]  
 return  $\boldsymbol{\tau}(t)$  [Nm] [N]

**:phi** [method]  
 return  $\boldsymbol{\phi}$  [rad] [m]

**:apply-config-to-robot**  $tm$  [method]  
 apply  $\mathbf{q}(t)$  to robot.

**:kin-target-coords-list**  $tm$  [method]

$$T_m^{kin-trg} = \{\mathbf{p}_m^{kin-trg}, \mathbf{R}_m^{kin-trg}\} \quad (m = 1, 2, \dots, N_{kin}) \quad (4.274)$$

$$\text{return } \mathcal{T}^{kin-trg} := \{T_1^{kin-trg}, T_2^{kin-trg}, \dots, T_{N_{kin}}^{kin-trg}\}$$

**:kin-attention-coords-list**  $tm$  [method]

$$T_m^{kin-att} = \{\mathbf{p}_m^{kin-att}, \mathbf{R}_m^{kin-att}\} \quad (m = 1, 2, \dots, N_{kin}) \quad (4.275)$$

$$\text{return } \mathcal{T}^{kin-att} := \{T_1^{kin-att}, T_2^{kin-att}, \dots, T_{N_{kin}}^{kin-att}\}$$

**:kin-scale-mat-list**  $tm$  [method]  
 return list of  $K_{kin}$

**:contact-target-coords-list**  $tm$  [method]

$$T_m^{cnt-trg} = \{\mathbf{p}_m^{cnt-trg}, \mathbf{R}_m^{cnt-trg}\} \quad (m = 1, 2, \dots, N_{cnt}) \quad (4.276)$$

$$\text{return } \mathcal{T}^{cnt-trg} := \{T_1^{cnt-trg}, T_2^{cnt-trg}, \dots, T_{N_{cnt}}^{cnt-trg}\}$$

**:contact-attention-coords-list**  $tm$  [method]

$$T_m^{cnt-att} = \{\mathbf{p}_m^{cnt-att}, \mathbf{R}_m^{cnt-att}\} \quad (m = 1, 2, \dots, N_{cnt}) \quad (4.277)$$

$$\text{return } \mathcal{T}^{cnt-att} := \{T_1^{cnt-att}, T_2^{cnt-att}, \dots, T_{N_{cnt}}^{cnt-att}\}$$

**:contact-constraint-list**  $tm$  [method]

return list of contact-constraint

**:wrench-list**  $tm$  [method]

return  $\{\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_{N_{cnt}}\}$

**:force-list**  $tm$  [method]

return  $\{\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_{N_{cnt}}\}$

**:moment-list**  $tm$  [method]

return  $\{\mathbf{n}_1, \mathbf{n}_2, \dots, \mathbf{n}_{N_{cnt}}\}$

**:mass** [method]

return  $m$  [kg]

**:mg-vec** [method]

return  $m\mathbf{g}$  [kg m/s<sup>2</sup>]

**:cog-from-model**  $\mathcal{E}key$  (*update?*  $t$ ) [method]

return  $\mathbf{p}_G(\mathbf{q})$  [m]

**:kinematics-instant-task-value**  $tm$  [method]

$$\bar{\mathbf{e}}^{kin}(\boldsymbol{\theta}(t), \boldsymbol{\phi}) = \begin{pmatrix} \bar{\mathbf{e}}_1^{kin}(\boldsymbol{\theta}(t), \boldsymbol{\phi}) \\ \bar{\mathbf{e}}_2^{kin}(\boldsymbol{\theta}(t), \boldsymbol{\phi}) \\ \vdots \\ \bar{\mathbf{e}}_{\bar{N}_{kin}(t)}^{kin}(\boldsymbol{\theta}(t), \boldsymbol{\phi}) \end{pmatrix} \in \mathbb{R}^{6\bar{N}_{kin}(t)} \quad (4.278)$$

$$\bar{\mathbf{e}}_m^{kin}(\boldsymbol{\theta}, \boldsymbol{\phi}) = \begin{pmatrix} \mathbf{p}_m^{kin-trg}(\boldsymbol{\theta}, \boldsymbol{\phi}) - \mathbf{p}_m^{kin-att}(\boldsymbol{\theta}, \boldsymbol{\phi}) \\ \mathbf{a} \left( \mathbf{R}_m^{kin-trg}(\boldsymbol{\theta}, \boldsymbol{\phi}) \mathbf{R}_m^{kin-att}(\boldsymbol{\theta}, \boldsymbol{\phi})^T \right) \end{pmatrix} \in \mathbb{R}^6 \quad (m = 1, 2, \dots, \bar{N}_{kin}(t)) \quad (4.279)$$

$\mathbf{a}(\mathbf{R})$  は姿勢行列  $\mathbf{R}$  の等価角軸ベクトルを表す .

return  $\bar{\mathbf{e}}^{kin}(\boldsymbol{\theta}(t), \boldsymbol{\phi}) \in \mathbb{R}^{6\bar{N}_{kin}(t)}$

**:kinematics-task-value**  $\mathcal{E}key$  (*update?*  $t$ ) [method]

$$\mathbf{e}^{kin}(\mathbf{p}_\theta, \boldsymbol{\phi}) = \begin{pmatrix} \bar{\mathbf{e}}^{kin}(\boldsymbol{\theta}(t_1), \boldsymbol{\phi}) \\ \vdots \\ \bar{\mathbf{e}}^{kin}(\boldsymbol{\theta}(t_{N_{tm-kin}}), \boldsymbol{\phi}) \end{pmatrix} \in \mathbb{R}^{6N_{kin}} \quad \left( N_{kin} := \sum_{t=1}^{t_{N_{tm-kin}}} \bar{N}_{kin}(t) \right) \quad (4.280)$$

return  $\mathbf{e}^{kin}(\mathbf{p}_\theta, \boldsymbol{\phi})$

**:eom-trans-instant-task-value**  $tm$  [method]

$$\bar{\mathbf{e}}^{eom-trans}(\mathbf{c}(t), \hat{\mathbf{w}}(t)) = m\ddot{\mathbf{c}} - \left\{ \sum_{m=1}^{N_{cnt}} \mathbf{f}_m - m\mathbf{g} \right\} = m\ddot{\mathbf{c}} - \sum_{m=1}^{N_{cnt}} \mathbf{f}_m + m\mathbf{g} \in \mathbb{R}^3 \quad (4.281)$$

return  $\bar{\mathbf{e}}^{eom-trans}(\mathbf{c}(t), \hat{\mathbf{w}}(t))$

**:eom-trans-task-value**  $\mathcal{E}key$  (*update?*  $t$ ) [method]

$$\mathbf{e}^{eom-trans}(\mathbf{p}_c, \mathbf{p}_{\hat{w}}) = \begin{pmatrix} \bar{\mathbf{e}}^{eom-trans}(\mathbf{c}(t_1), \hat{\mathbf{w}}(t_1)) \\ \vdots \\ \bar{\mathbf{e}}^{eom-trans}(\mathbf{c}(t_{N_{tm-eom}}), \hat{\mathbf{w}}(t_{N_{tm-eom}})) \end{pmatrix} \in \mathbb{R}^{3N_{tm-eom}} \quad (4.282)$$

return  $\mathbf{e}^{eom-trans}(\mathbf{p}_c, \mathbf{p}_{\hat{w}})$

**:eom-rot-instant-task-value**  $tm$  [method]

$$\bar{\mathbf{e}}^{eom-rot}(\boldsymbol{\theta}(t), \mathbf{c}(t), \mathbf{L}(t), \hat{\mathbf{w}}(t), \phi) = \dot{\mathbf{L}} - \sum_{m=1}^{N_{cnt}} \{(\mathbf{p}_m^{cnt-trg}(\boldsymbol{\theta}, \phi) - \mathbf{c}) \times \mathbf{f}_m + \mathbf{n}_m\} \in \mathbb{R}^3 \quad (4.283)$$

return  $\bar{\mathbf{e}}^{eom-rot}(\boldsymbol{\theta}(t), \mathbf{c}(t), \mathbf{L}(t), \hat{\mathbf{w}}(t), \phi)$

**:eom-rot-task-value**  $\mathcal{E}_{key}$  (update?  $t$ ) [method]

$$\mathbf{e}^{eom-rot}(\mathbf{p}_{\theta}, \mathbf{p}_c, \mathbf{p}_L, \mathbf{p}_{\hat{w}}, \phi) = \begin{pmatrix} \bar{\mathbf{e}}^{eom-rot}(\boldsymbol{\theta}(t_1), \mathbf{c}(t_1), \mathbf{L}(t_1), \hat{\mathbf{w}}(t_1), \phi) \\ \vdots \\ \bar{\mathbf{e}}^{eom-rot}(\boldsymbol{\theta}(t_{N_{tm-eom}}), \mathbf{c}(t_{N_{tm-eom}}), \mathbf{L}(t_{N_{tm-eom}}), \hat{\mathbf{w}}(t_{N_{tm-eom}}), \phi) \end{pmatrix} \in \mathbb{R}^{3N_{tm-eom}} \quad (4.284)$$

return  $\mathbf{e}^{eom-rot}(\mathbf{p}_{\theta}, \mathbf{p}_c, \mathbf{p}_L, \mathbf{p}_{\hat{w}}, \phi)$

**:cog-instant-task-value**  $tm$  [method]

$$\bar{\mathbf{e}}^{cog}(\boldsymbol{\theta}(t), \mathbf{c}(t), \phi) = \mathbf{p}_G(\boldsymbol{\theta}, \phi) - \mathbf{c} \in \mathbb{R}^3 \quad (4.285)$$

return  $\bar{\mathbf{e}}^{cog}(\boldsymbol{\theta}(t), \mathbf{c}(t), \phi)$

**:cog-task-value**  $\mathcal{E}_{key}$  (update?  $t$ ) [method]

$$\mathbf{e}^{cog}(\mathbf{p}_{\theta}, \mathbf{p}_c, \phi) = \begin{pmatrix} \bar{\mathbf{e}}^{cog}(\boldsymbol{\theta}(t_1), \mathbf{c}(t_1), \phi) \\ \vdots \\ \bar{\mathbf{e}}^{cog}(\boldsymbol{\theta}(t_{N_{tm-eom}}), \mathbf{c}(t_{N_{tm-eom}}), \phi) \end{pmatrix} \in \mathbb{R}^{3N_{tm-eom}} \quad (4.286)$$

return  $\mathbf{e}^{cog}(\mathbf{p}_{\theta}, \mathbf{p}_c, \phi)$

**:ang-moment-instant-task-value**  $tm$  [method]

$$\bar{\mathbf{e}}^{ang-moment}(\boldsymbol{\theta}(t), \mathbf{L}(t), \phi) = \mathbf{L}(t) - \left\{ \mathbf{A}_{\theta}(\boldsymbol{\theta}(t), \phi(t)) \dot{\boldsymbol{\theta}}(t) + \mathbf{A}_{\phi}(\boldsymbol{\theta}(t), \phi(t)) \dot{\phi}(t) \right\} \in \mathbb{R}^3 \quad (4.287)$$

本実装では,  $\mathbf{A}_{\theta} = \mathbf{A}_{\phi} = \mathbf{O}$  と仮定を置く. このとき, タスク関数は次式となる.

$$\bar{\mathbf{e}}^{ang-moment}(\mathbf{L}(t)) = \mathbf{L}(t) \in \mathbb{R}^3 \quad (4.288)$$

return  $\bar{\mathbf{e}}^{ang-moment}(\mathbf{L}(t))$

**:ang-moment-task-value**  $\mathcal{E}_{key}$  (update?  $t$ ) [method]



$$\mathbf{e}^{ang-moment}(\mathbf{p}_L) = \begin{pmatrix} \bar{\mathbf{e}}^{ang-moment}(\mathbf{L}(t_1)) \\ \vdots \\ \bar{\mathbf{e}}^{ang-moment}(\mathbf{L}(t_{N_{tm-com}})) \end{pmatrix} \in \mathbb{R}^{3N_{tm-com}} \quad (4.289)$$

return  $\mathbf{e}^{ang-moment}(\mathbf{p}_L)$

**:task-value**  $\mathcal{E}_{key}$  (update?  $t$ ) [method]

$$\mathbf{e}(\mathbf{q}) = \begin{pmatrix} \mathbf{e}^{kin}(\mathbf{p}_\theta, \phi) \\ \mathbf{e}^{com-trans}(\mathbf{p}_c, \mathbf{p}_{\hat{w}}) \\ \mathbf{e}^{com-rot}(\mathbf{p}_\theta, \mathbf{p}_c, \mathbf{p}_L, \mathbf{p}_{\hat{w}}, \phi) \\ \mathbf{e}^{cog}(\mathbf{p}_\theta, \mathbf{p}_c, \phi) \\ \mathbf{e}^{ang-moment}(\mathbf{p}_\theta, \mathbf{p}_L, \phi) \\ \mathbf{e}^{trq}(\mathbf{p}_\theta, \mathbf{p}_{\hat{w}}, \mathbf{p}_\tau, \phi) \\ \mathbf{e}^{posture}(\mathbf{p}_\theta) \end{pmatrix} \quad (4.290)$$

return  $\mathbf{e}(\mathbf{q}) \in \mathbb{R}^{dim(\mathbf{e})}$

**:kinematics-instant-task-jacobian-with-theta-control-vector**  $tm$  [method]

$$\frac{\partial \bar{\mathbf{e}}^{kin}(\boldsymbol{\theta}(t), \phi)}{\partial \mathbf{p}_\theta} = \frac{\partial \bar{\mathbf{e}}^{kin}(\boldsymbol{\theta}(t), \phi)}{\partial \boldsymbol{\theta}} \frac{\partial \boldsymbol{\theta}(t)}{\partial \mathbf{p}_\theta} \quad (4.291)$$

$$\frac{\partial \bar{\mathbf{e}}^{kin}(\boldsymbol{\theta}(t), \phi)}{\partial \boldsymbol{\theta}} = \begin{pmatrix} \frac{\partial \bar{\mathbf{e}}_1^{kin}(\boldsymbol{\theta}(t), \phi)}{\partial \boldsymbol{\theta}} \\ \vdots \\ \frac{\partial \bar{\mathbf{e}}_{N_{kin}(t)}^{kin}(\boldsymbol{\theta}(t), \phi)}{\partial \boldsymbol{\theta}} \end{pmatrix} = \begin{pmatrix} \mathbf{J}_{1,\theta}(t) \\ \vdots \\ \mathbf{J}_{N_{kin}(t),\theta}(t) \end{pmatrix} \quad (4.292)$$

$$\frac{\partial \boldsymbol{\theta}(t)}{\partial \mathbf{p}_\theta} = \frac{\partial}{\partial \mathbf{p}_\theta} \mathbf{B}_{\theta,n}(t) \mathbf{p}_\theta = \mathbf{B}_{\theta,n}(t) \quad (4.293)$$

return  $\frac{\partial \mathbf{e}^{kin}}{\partial \mathbf{p}_\theta} \in \mathbb{R}^{6N_{kin} \times dim(\mathbf{p}_\theta)}$

**:kinematics-task-jacobian-with-theta-control-vector** [method]

$$\frac{\partial \mathbf{e}^{kin}}{\partial \mathbf{p}_\theta} = \begin{pmatrix} \frac{\partial \bar{\mathbf{e}}^{kin}(\boldsymbol{\theta}(t_1), \phi(t_1))}{\partial \mathbf{p}_\theta} \\ \vdots \\ \frac{\partial \bar{\mathbf{e}}^{kin}(\boldsymbol{\theta}(t_{N_{tm-kin}}), \phi(t_{N_{tm-kin}}))}{\partial \mathbf{p}_\theta} \end{pmatrix} \quad (4.294)$$

return  $\frac{\partial \mathbf{e}^{kin}}{\partial \mathbf{p}_\theta} \in \mathbb{R}^{6N_{kin} \times dim(\mathbf{p}_\theta)}$

**:kinematics-instant-task-jacobian-with-phi**  $tm$  [method]

$$\frac{\partial \bar{\mathbf{e}}^{kin}(\boldsymbol{\theta}(t), \phi)}{\partial \phi} = \frac{\partial \bar{\mathbf{e}}^{kin}(\boldsymbol{\theta}(t), \phi)}{\partial \phi} \quad (4.295)$$

$$\frac{\partial \bar{\mathbf{e}}^{kin}(\boldsymbol{\theta}(t), \phi)}{\partial \phi} = \begin{pmatrix} \frac{\partial \bar{\mathbf{e}}_1^{kin}(\boldsymbol{\theta}(t), \phi)}{\partial \phi} \\ \vdots \\ \frac{\partial \bar{\mathbf{e}}_{N_{kin}(t)}^{kin}(\boldsymbol{\theta}(t), \phi)}{\partial \phi} \end{pmatrix} = \begin{pmatrix} \mathbf{J}_{1,\phi}(t) \\ \vdots \\ \mathbf{J}_{N_{kin}(t),\phi}(t) \end{pmatrix} \quad (4.296)$$

$$\text{return } \frac{\partial \mathbf{e}^{kin}}{\partial \boldsymbol{\phi}} \in \mathbb{R}^{6N_{kin} \times \dim(\boldsymbol{\phi})}$$

**:kinematics-task-jacobian-with-phi**

[method]

$$\frac{\partial \mathbf{e}^{kin}}{\partial \boldsymbol{\phi}} = \begin{pmatrix} \frac{\partial \bar{\mathbf{e}}^{kin}(\boldsymbol{\theta}(t_1), \boldsymbol{\phi}(t_1))}{\partial \boldsymbol{\phi}} \\ \vdots \\ \frac{\partial \bar{\mathbf{e}}^{kin}(\boldsymbol{\theta}(t_{N_{tm-kin}}), \boldsymbol{\phi}(t_{N_{tm-kin}}))}{\partial \boldsymbol{\phi}} \end{pmatrix} \quad (4.297)$$

$$\text{return } \frac{\partial \mathbf{e}^{kin}}{\partial \boldsymbol{\phi}} \in \mathbb{R}^{6N_{kin} \times \dim(\boldsymbol{\phi})}$$

**:eom-trans-instant-task-jacobian-with-cog-control-vector**  $tm$

[method]

$$\frac{\partial \bar{\mathbf{e}}^{eom-trans}(\mathbf{c}(t), \hat{\mathbf{w}}(t))}{\partial \mathbf{p}_c} = m \frac{\partial \ddot{\mathbf{c}}(t)}{\partial \mathbf{p}_c} \quad (4.298)$$

$$= m \frac{\partial}{\partial \mathbf{p}_c} \mathbf{B}_{c,n-2}(t) \hat{\mathbf{D}}_2 \mathbf{p}_c \quad (4.299)$$

$$= m \mathbf{B}_{c,n-2}(t) \hat{\mathbf{D}}_2 \quad (4.300)$$

$$\text{return } \frac{\partial \bar{\mathbf{e}}^{eom-trans}(\mathbf{c}(t), \hat{\mathbf{w}}(t))}{\partial \mathbf{p}_c} \in \mathbb{R}^{3 \times \dim(\mathbf{p}_c)}$$

**:eom-trans-task-jacobian-with-cog-control-vector**

[method]

$$\frac{\partial \mathbf{e}^{eom-trans}}{\partial \mathbf{p}_c} = \begin{pmatrix} \frac{\partial \bar{\mathbf{e}}^{eom-trans}(\mathbf{c}(t_1), \hat{\mathbf{w}}(t_1))}{\partial \mathbf{p}_c} \\ \vdots \\ \frac{\partial \bar{\mathbf{e}}^{eom-trans}(\mathbf{c}(t_{N_{tm-eom}}), \hat{\mathbf{w}}(t_{N_{tm-eom}}))}{\partial \mathbf{p}_c} \end{pmatrix} \quad (4.301)$$

$$\text{return } \frac{\partial \mathbf{e}^{eom-trans}}{\partial \mathbf{p}_c} \in \mathbb{R}^{3N_{tm-eom} \times \dim(\mathbf{p}_c)}$$

**:eom-trans-instant-task-jacobian-with-wrench-control-vector**  $tm$

[method]

$$\frac{\partial \bar{\mathbf{e}}^{eom-trans}(\mathbf{c}(t), \hat{\mathbf{w}}(t))}{\partial \mathbf{p}_{\hat{\mathbf{w}}}} = \frac{\partial \bar{\mathbf{e}}^{eom-trans}(\mathbf{c}(t), \hat{\mathbf{w}}(t))}{\partial \hat{\mathbf{w}}} \frac{\partial \hat{\mathbf{w}}(t)}{\partial \mathbf{p}_{\hat{\mathbf{w}}}} \quad (4.302)$$

$$\frac{\partial \bar{\mathbf{e}}^{eom-trans}(\mathbf{c}(t), \hat{\mathbf{w}}(t))}{\partial \hat{\mathbf{w}}} = \begin{pmatrix} -\mathbf{I}_3 & \mathbf{O}_3 & \cdots & -\mathbf{I}_3 & \mathbf{O}_3 \end{pmatrix} \quad (4.303)$$

$$\frac{\partial \hat{\mathbf{w}}(t)}{\partial \mathbf{p}_{\hat{\mathbf{w}}}} = \frac{\partial}{\partial \mathbf{p}_{\hat{\mathbf{w}}}} \mathbf{B}_{\hat{\mathbf{w}},n}(t) \mathbf{p}_{\hat{\mathbf{w}}} = \mathbf{B}_{\hat{\mathbf{w}},n}(t) \quad (4.304)$$

$$\text{return } \frac{\partial \bar{\mathbf{e}}^{eom-trans}(\mathbf{c}(t), \hat{\mathbf{w}}(t))}{\partial \mathbf{p}_{\hat{\mathbf{w}}}} \in \mathbb{R}^{3 \times \dim(\mathbf{p}_{\hat{\mathbf{w}}})}$$

**:eom-trans-task-jacobian-with-wrench-control-vector**

[method]

$$\frac{\partial \mathbf{e}^{eom-trans}}{\partial \mathbf{p}_{\hat{\mathbf{w}}}} = \begin{pmatrix} \frac{\partial \bar{\mathbf{e}}^{eom-trans}(\mathbf{c}(t_1), \hat{\mathbf{w}}(t_1))}{\partial \mathbf{p}_{\hat{\mathbf{w}}}} \\ \vdots \\ \frac{\partial \bar{\mathbf{e}}^{eom-trans}(\mathbf{c}(t_{N_{tm-eom}}), \hat{\mathbf{w}}(t_{N_{tm-eom}}))}{\partial \mathbf{p}_{\hat{\mathbf{w}}}} \end{pmatrix} \quad (4.305)$$

$$\text{return } \frac{\partial \mathbf{e}^{eom-trans}}{\partial \mathbf{p}_{\hat{\mathbf{w}}}} \in \mathbb{R}^{3N_{tm-eom} \times \dim(\mathbf{p}_{\hat{\mathbf{w}}})}$$

**:eom-rot-instant-task-jacobian-with-theta-control-vector** *tm*

[method]

$$\bar{\mathbf{e}}^{eom-rot}(\boldsymbol{\theta}(t), \mathbf{c}(t), \mathbf{L}(t), \hat{\mathbf{w}}(t), \phi) = \dot{\mathbf{L}} - \sum_{m=1}^{N_{cnt}} \{(\mathbf{p}_m^{cnt-trg}(\boldsymbol{\theta}, \phi) - \mathbf{c}) \times \mathbf{f}_m + \mathbf{n}_m\} \in \mathbb{R}^3 \quad (4.306)$$

$$\frac{\partial \bar{\mathbf{e}}^{eom-rot}(\boldsymbol{\theta}(t), \mathbf{c}(t), \mathbf{L}(t), \hat{\mathbf{w}}(t), \phi)}{\partial \mathbf{p}_\theta} = \frac{\partial \bar{\mathbf{e}}^{eom-rot}(\boldsymbol{\theta}(t), \mathbf{c}(t), \mathbf{L}(t), \hat{\mathbf{w}}(t), \phi)}{\partial \boldsymbol{\theta}} \frac{\partial \boldsymbol{\theta}(t)}{\partial \mathbf{p}_\theta} \quad (4.307)$$

$$\frac{\partial \bar{\mathbf{e}}^{eom-rot}(\boldsymbol{\theta}(t), \mathbf{c}(t), \mathbf{L}(t), \hat{\mathbf{w}}(t), \phi)}{\partial \boldsymbol{\theta}} = \sum_{m=1}^{N_{cnt}} \{[\mathbf{f}_m(t) \times] \mathbf{J}_{m,\theta}^{cnt-trg}(t)\} \quad (4.308)$$

$$\frac{\partial \boldsymbol{\theta}(t)}{\partial \mathbf{p}_\theta} = \frac{\partial}{\partial \mathbf{p}_\theta} \mathbf{B}_{\theta,n}(t) \mathbf{p}_\theta = \mathbf{B}_{\theta,n}(t) \quad (4.309)$$

$$\text{return } \frac{\partial \bar{\mathbf{e}}^{eom-rot}(\boldsymbol{\theta}(t), \mathbf{c}(t), \mathbf{L}(t), \hat{\mathbf{w}}(t), \phi)}{\partial \mathbf{p}_\theta} \in \mathbb{R}^{3 \times \dim(\mathbf{p}_\theta)}$$

**:eom-rot-task-jacobian-with-theta-control-vector**

[method]

$$\frac{\partial \mathbf{e}^{eom-rot}}{\partial \mathbf{p}_\theta} = \begin{pmatrix} \frac{\partial \bar{\mathbf{e}}^{eom-rot}(\boldsymbol{\theta}(t_1), \mathbf{c}(t_1), \mathbf{L}(t_1), \hat{\mathbf{w}}(t_1), \phi)}{\partial \mathbf{p}_\theta} \\ \vdots \\ \frac{\partial \bar{\mathbf{e}}^{eom-rot}(\boldsymbol{\theta}(t_{N_{tm-eom}}), \mathbf{c}(t_{N_{tm-eom}}), \mathbf{L}(t_{N_{tm-eom}}), \hat{\mathbf{w}}(t_{N_{tm-eom}}), \phi)}{\partial \mathbf{p}_\theta} \end{pmatrix} \quad (4.310)$$

$$\text{return } \frac{\partial \mathbf{e}^{eom-rot}}{\partial \mathbf{p}_\theta} \in \mathbb{R}^{3N_{tm-eom} \times \dim(\mathbf{p}_\theta)}$$

**:eom-rot-instant-task-jacobian-with-cog-control-vector** *tm*

[method]

$$\frac{\partial \bar{\mathbf{e}}^{eom-rot}(\boldsymbol{\theta}(t), \mathbf{c}(t), \mathbf{L}(t), \hat{\mathbf{w}}(t), \phi)}{\partial \mathbf{p}_c} = \frac{\partial \bar{\mathbf{e}}^{eom-rot}(\boldsymbol{\theta}(t), \mathbf{c}(t), \mathbf{L}(t), \hat{\mathbf{w}}(t), \phi)}{\partial \mathbf{c}} \frac{\partial \mathbf{c}(t)}{\partial \mathbf{p}_c} \quad (4.311)$$

$$\frac{\partial \bar{\mathbf{e}}^{eom-rot}(\boldsymbol{\theta}(t), \mathbf{c}(t), \mathbf{L}(t), \hat{\mathbf{w}}(t), \phi)}{\partial \mathbf{c}} = - \sum_{m=1}^{N_{cnt}} [\mathbf{f}_m \times] = \left[ \left( - \sum_{m=1}^{N_{cnt}} \mathbf{f}_m \right) \times \right] \quad (4.312)$$

$$\frac{\partial \mathbf{c}(t)}{\partial \mathbf{p}_c} = \frac{\partial}{\partial \mathbf{p}_c} \mathbf{B}_{c,n}(t) \mathbf{p}_c = \mathbf{B}_{c,n}(t) \quad (4.313)$$

$$\text{return } \frac{\partial \bar{\mathbf{e}}^{eom-rot}(\boldsymbol{\theta}(t), \mathbf{c}(t), \mathbf{L}(t), \hat{\mathbf{w}}(t), \phi)}{\partial \mathbf{p}_c} \in \mathbb{R}^{3 \times \dim(\mathbf{p}_c)}$$

**:eom-rot-task-jacobian-with-cog-control-vector**

[method]

$$\frac{\partial \mathbf{e}^{eom-rot}}{\partial \mathbf{p}_c} = \begin{pmatrix} \frac{\partial \bar{\mathbf{e}}^{eom-rot}(\boldsymbol{\theta}(t_1), \mathbf{c}(t_1), \mathbf{L}(t_1), \hat{\mathbf{w}}(t_1), \phi)}{\partial \mathbf{p}_c} \\ \vdots \\ \frac{\partial \bar{\mathbf{e}}^{eom-rot}(\boldsymbol{\theta}(t_{N_{tm-eom}}), \mathbf{c}(t_{N_{tm-eom}}), \mathbf{L}(t_{N_{tm-eom}}), \hat{\mathbf{w}}(t_{N_{tm-eom}}), \phi)}{\partial \mathbf{p}_c} \end{pmatrix} \quad (4.314)$$

$$\text{return } \frac{\partial \mathbf{e}^{eom-rot}}{\partial \mathbf{p}_c} \in \mathbb{R}^{3N_{tm-eom} \times \dim(\mathbf{p}_c)}$$

**:eom-rot-instant-task-jacobian-with-ang-moment-control-vector** *tm*

[method]

$$\frac{\partial \bar{e}^{eom-rot}(\boldsymbol{\theta}(t), \mathbf{c}(t), \mathbf{L}(t), \hat{\mathbf{w}}(t), \phi)}{\partial \mathbf{p}_L} = \frac{\partial \dot{\mathbf{L}}(t)}{\partial \mathbf{p}_L} \quad (4.315)$$

$$= \frac{\partial}{\partial \mathbf{p}_L} \mathbf{B}_{L,n-1}(t) \hat{\mathbf{D}}_1 \mathbf{p}_L \quad (4.316)$$

$$= \mathbf{B}_{L,n-1}(t) \hat{\mathbf{D}}_1 \quad (4.317)$$

$$\text{return } \frac{\partial \bar{e}^{eom-rot}(\boldsymbol{\theta}(t), \mathbf{c}(t), \mathbf{L}(t), \hat{\mathbf{w}}(t), \phi)}{\partial \mathbf{p}_L} \in \mathbb{R}^{3 \times \dim(\mathbf{p}_L)}$$

**:eom-rot-task-jacobian-with-ang-moment-control-vector** [method]

$$\frac{\partial \mathbf{e}^{eom-rot}}{\partial \mathbf{p}_L} = \begin{pmatrix} \frac{\partial \bar{e}^{eom-rot}(\boldsymbol{\theta}(t_1), \mathbf{c}(t_1), \mathbf{L}(t_1), \hat{\mathbf{w}}(t_1), \phi)}{\partial \mathbf{p}_L} \\ \vdots \\ \frac{\partial \bar{e}^{eom-rot}(\boldsymbol{\theta}(t_{N_{tm-eom}}), \mathbf{c}(t_{N_{tm-eom}}), \mathbf{L}(t_{N_{tm-eom}}), \hat{\mathbf{w}}(t_{N_{tm-eom}}), \phi)}{\partial \mathbf{p}_L} \end{pmatrix} \quad (4.318)$$

$$\text{return } \frac{\partial \mathbf{e}^{eom-rot}}{\partial \mathbf{p}_L} \in \mathbb{R}^{3N_{tm-eom} \times \dim(\mathbf{p}_L)}$$

**:eom-rot-instant-task-jacobian-with-wrench-control-vector**  $tm$  [method]

$$\frac{\partial \bar{e}^{eom-rot}(\boldsymbol{\theta}(t), \mathbf{c}(t), \mathbf{L}(t), \hat{\mathbf{w}}(t), \phi)}{\partial \mathbf{p}_{\hat{\mathbf{w}}}} = \frac{\partial \bar{e}^{eom-rot}(\boldsymbol{\theta}(t), \mathbf{c}(t), \mathbf{L}(t), \hat{\mathbf{w}}(t), \phi)}{\partial \hat{\mathbf{w}}} \frac{\partial \hat{\mathbf{w}}(t)}{\partial \mathbf{p}_{\hat{\mathbf{w}}}} \quad (4.319)$$

$$\frac{\partial \bar{e}^{eom-rot}(\boldsymbol{\theta}(t), \mathbf{c}(t), \mathbf{L}(t), \hat{\mathbf{w}}(t), \phi)}{\partial \hat{\mathbf{w}}} = \left( \begin{bmatrix} -(\mathbf{p}_1^{cnt-trg}(\boldsymbol{\theta}, \phi) - \mathbf{c}) \times \\ -\mathbf{I}_3 \quad \cdots \quad -(\mathbf{p}_{N_{cnt}}^{cnt-trg}(\boldsymbol{\theta}, \phi) - \mathbf{c}) \times \end{bmatrix} \right) \quad (4.320)$$

$$\frac{\partial \hat{\mathbf{w}}(t)}{\partial \mathbf{p}_{\hat{\mathbf{w}}}} = \frac{\partial}{\partial \mathbf{p}_{\hat{\mathbf{w}}}} \mathbf{B}_{\hat{\mathbf{w}},n}(t) \mathbf{p}_{\hat{\mathbf{w}}} = \mathbf{B}_{\hat{\mathbf{w}},n}(t) \quad (4.321)$$

$$\text{return } \frac{\partial \bar{e}^{eom-rot}(\boldsymbol{\theta}(t), \mathbf{c}(t), \mathbf{L}(t), \hat{\mathbf{w}}(t), \phi)}{\partial \mathbf{p}_{\hat{\mathbf{w}}}} \in \mathbb{R}^{3 \times \dim(\mathbf{p}_{\hat{\mathbf{w}}})}$$

**:eom-rot-task-jacobian-with-wrench-control-vector** [method]

$$\frac{\partial \mathbf{e}^{eom-rot}}{\partial \mathbf{p}_{\hat{\mathbf{w}}}} = \begin{pmatrix} \frac{\partial \bar{e}^{eom-rot}(\boldsymbol{\theta}(t_1), \mathbf{c}(t_1), \mathbf{L}(t_1), \hat{\mathbf{w}}(t_1), \phi)}{\partial \mathbf{p}_{\hat{\mathbf{w}}}} \\ \vdots \\ \frac{\partial \bar{e}^{eom-rot}(\boldsymbol{\theta}(t_{N_{tm-eom}}), \mathbf{c}(t_{N_{tm-eom}}), \mathbf{L}(t_{N_{tm-eom}}), \hat{\mathbf{w}}(t_{N_{tm-eom}}), \phi)}{\partial \mathbf{p}_{\hat{\mathbf{w}}}} \end{pmatrix} \quad (4.322)$$

$$\text{return } \frac{\partial \mathbf{e}^{eom-rot}}{\partial \mathbf{p}_{\hat{\mathbf{w}}}} \in \mathbb{R}^{3N_{tm-eom} \times \dim(\mathbf{p}_{\hat{\mathbf{w}}})}$$

**:eom-rot-instant-task-jacobian-with-phi**  $tm$  [method]

$$\frac{\partial \bar{e}^{eom-rot}(\boldsymbol{\theta}(t), \mathbf{c}(t), \mathbf{L}(t), \hat{\mathbf{w}}(t), \phi)}{\partial \phi} = \frac{\partial \bar{e}^{eom-rot}(\boldsymbol{\theta}(t), \mathbf{c}(t), \mathbf{L}(t), \hat{\mathbf{w}}(t), \phi)}{\partial \phi} \quad (4.323)$$

$$= \sum_{m=1}^{N_{cnt}} \left\{ [\mathbf{f}_m(t) \times] \mathbf{J}_{m,\phi}^{cnt-trg}(t) \right\} \quad (4.324)$$

$$\text{return } \frac{\partial \bar{e}^{eom-rot}(\boldsymbol{\theta}(t), \mathbf{c}(t), \mathbf{L}(t), \hat{\mathbf{w}}(t), \phi)}{\partial \phi} \in \mathbb{R}^{3 \times \dim(\phi)}$$

:eom-rot-task-jacobian-with-phi

[method]

$$\frac{\partial \mathbf{e}^{eom-rot}}{\partial \phi} = \begin{pmatrix} \frac{\partial \bar{\mathbf{e}}^{eom-rot}(\boldsymbol{\theta}(t_1), \mathbf{c}(t_1), \mathbf{L}(t_1), \hat{\mathbf{w}}(t_1), \phi)}{\partial \phi} \\ \vdots \\ \frac{\partial \bar{\mathbf{e}}^{eom-rot}(\boldsymbol{\theta}(t_{N_{tm-eom}}), \mathbf{c}(t_{N_{tm-eom}}), \mathbf{L}(t_{N_{tm-eom}}), \hat{\mathbf{w}}(t_{N_{tm-eom}}), \phi)}{\partial \phi} \end{pmatrix} \quad (4.325)$$

$$\text{return } \frac{\partial \mathbf{e}^{eom-rot}}{\partial \phi} \in \mathbb{R}^{3N_{tm-eom} \times \dim(\phi)}$$

:cog-instant-task-jacobian-with-theta-control-vector  $tm$ 

[method]

$$\frac{\partial \bar{\mathbf{e}}^{cog}(\boldsymbol{\theta}(t), \mathbf{c}(t), \phi)}{\partial \mathbf{p}_\theta} = \frac{\partial \bar{\mathbf{e}}^{cog}(\boldsymbol{\theta}(t), \mathbf{c}(t), \phi)}{\partial \boldsymbol{\theta}} \frac{\partial \boldsymbol{\theta}(t)}{\partial \mathbf{p}_\theta} \quad (4.326)$$

$$\frac{\partial \bar{\mathbf{e}}^{cog}(\boldsymbol{\theta}(t), \mathbf{c}(t), \phi)}{\partial \boldsymbol{\theta}} = \mathbf{J}_{G,\theta}(t) \quad (4.327)$$

$$\frac{\partial \boldsymbol{\theta}(t)}{\partial \mathbf{p}_\theta} = \frac{\partial}{\partial \mathbf{p}_\theta} \mathbf{B}_{\theta,n}(t) \mathbf{p}_\theta = \mathbf{B}_{\theta,n}(t) \quad (4.328)$$

$$\text{return } \frac{\partial \bar{\mathbf{e}}^{cog}(\boldsymbol{\theta}(t), \mathbf{c}(t), \phi)}{\partial \mathbf{p}_\theta} \in \mathbb{R}^{3 \times \dim(\mathbf{p}_\theta)}$$

:cog-task-jacobian-with-theta-control-vector

[method]

$$\frac{\partial \mathbf{e}^{cog}}{\partial \mathbf{p}_\theta} = \begin{pmatrix} \frac{\partial \bar{\mathbf{e}}^{cog}(\boldsymbol{\theta}(t_1), \mathbf{c}(t_1), \phi)}{\partial \mathbf{p}_\theta} \\ \vdots \\ \frac{\partial \bar{\mathbf{e}}^{cog}(\boldsymbol{\theta}(t_{N_{tm-eom}}), \mathbf{c}(t_{N_{tm-eom}}), \phi)}{\partial \mathbf{p}_\theta} \end{pmatrix} \quad (4.329)$$

$$\text{return } \frac{\partial \mathbf{e}^{cog}}{\partial \mathbf{p}_\theta} \in \mathbb{R}^{3N_{tm-eom} \times \dim(\mathbf{p}_\theta)}$$

:cog-instant-task-jacobian-with-cog-control-vector  $tm$ 

[method]

$$\frac{\partial \bar{\mathbf{e}}^{cog}(\boldsymbol{\theta}(t), \mathbf{c}(t), \phi)}{\partial \mathbf{p}_c} = \frac{\partial \bar{\mathbf{e}}^{cog}(\boldsymbol{\theta}(t), \mathbf{c}(t), \phi)}{\partial \mathbf{c}} \frac{\partial \mathbf{c}(t)}{\partial \mathbf{p}_c} \quad (4.330)$$

$$\frac{\partial \bar{\mathbf{e}}^{cog}(\boldsymbol{\theta}(t), \mathbf{c}(t), \phi)}{\partial \mathbf{c}} = -\mathbf{I}_3 \quad (4.331)$$

$$\frac{\partial \mathbf{c}(t)}{\partial \mathbf{p}_c} = \frac{\partial}{\partial \mathbf{p}_c} \mathbf{B}_{c,n}(t) \mathbf{p}_c = \mathbf{B}_{c,n}(t) \quad (4.332)$$

$$\text{return } \frac{\partial \bar{\mathbf{e}}^{cog}(\boldsymbol{\theta}(t), \mathbf{c}(t), \phi)}{\partial \mathbf{p}_c} \in \mathbb{R}^{3 \times \dim(\mathbf{p}_c)}$$

:cog-task-jacobian-with-cog-control-vector

[method]

$$\frac{\partial \mathbf{e}^{cog}}{\partial \mathbf{p}_c} = \begin{pmatrix} \frac{\partial \bar{\mathbf{e}}^{cog}(\boldsymbol{\theta}(t_1), \mathbf{c}(t_1), \phi)}{\partial \mathbf{p}_c} \\ \vdots \\ \frac{\partial \bar{\mathbf{e}}^{cog}(\boldsymbol{\theta}(t_{N_{tm-eom}}), \mathbf{c}(t_{N_{tm-eom}}), \phi)}{\partial \mathbf{p}_c} \end{pmatrix} \quad (4.333)$$

$$\text{return } \frac{\partial \mathbf{e}^{cog}}{\partial \mathbf{p}_c} \in \mathbb{R}^{3N_{tm-eom} \times \dim(\mathbf{p}_c)}$$

:cog-instant-task-jacobian-with-phi *tm*

[method]

$$\frac{\partial \bar{\mathbf{e}}^{cog}(\boldsymbol{\theta}(t), \mathbf{c}(t), \phi)}{\partial \phi} = \frac{\partial \bar{\mathbf{e}}^{cog}(\boldsymbol{\theta}(t), \mathbf{c}(t), \phi)}{\partial \phi} = \mathbf{J}_{G, \phi}(t) \quad (4.334)$$

$$\text{return } \frac{\partial \bar{\mathbf{e}}^{cog}(\boldsymbol{\theta}(t), \mathbf{c}(t), \phi)}{\partial \phi} \in \mathbb{R}^{3 \times \dim(\phi)}$$

:cog-task-jacobian-with-phi

[method]

$$\frac{\partial \mathbf{e}^{cog}}{\partial \phi} = \begin{pmatrix} \frac{\partial \bar{\mathbf{e}}^{cog}(\boldsymbol{\theta}(t_1), \mathbf{c}(t_1), \phi)}{\partial \phi} \\ \vdots \\ \frac{\partial \bar{\mathbf{e}}^{cog}(\boldsymbol{\theta}(t_{N_{tm-com}}), \mathbf{c}(t_{N_{tm-com}}), \phi)}{\partial \phi} \end{pmatrix} \quad (4.335)$$

$$\text{return } \frac{\partial \mathbf{e}^{cog}}{\partial \phi} \in \mathbb{R}^{3N_{tm-com} \times \dim(\phi)}$$

:ang-moment-instant-task-jacobian-with-ang-moment-control-vector *tm*

[method]

$$\frac{\partial \bar{\mathbf{e}}^{ang-moment}(\mathbf{L}(t))}{\partial \mathbf{p}_L} = \frac{\partial \bar{\mathbf{e}}^{ang-moment}(\mathbf{L}(t))}{\partial \mathbf{L}} \frac{\partial \mathbf{L}(t)}{\partial \mathbf{p}_L} \quad (4.336)$$

$$\frac{\partial \bar{\mathbf{e}}^{ang-moment}(\mathbf{L}(t))}{\partial \mathbf{L}} = \mathbf{I}_3 \quad (4.337)$$

$$\frac{\partial \mathbf{L}(t)}{\partial \mathbf{p}_L} = \frac{\partial}{\partial \mathbf{p}_L} \mathbf{B}_{L,n}(t) \mathbf{p}_L = \mathbf{B}_{L,n}(t) \quad (4.338)$$

$$\text{return } \frac{\partial \bar{\mathbf{e}}^{ang-moment}(\mathbf{L}(t))}{\partial \mathbf{p}_L} \in \mathbb{R}^{3 \times \dim(\mathbf{p}_L)}$$

:ang-moment-task-jacobian-with-ang-moment-control-vector

[method]

$$\frac{\partial \mathbf{e}^{ang-moment}}{\partial \mathbf{p}_L} = \begin{pmatrix} \frac{\partial \bar{\mathbf{e}}^{ang-moment}(\mathbf{L}(t_1))}{\partial \mathbf{p}_L} \\ \vdots \\ \frac{\partial \bar{\mathbf{e}}^{ang-moment}(\mathbf{L}(t_{N_{tm-com}}))}{\partial \mathbf{p}_L} \end{pmatrix} \quad (4.339)$$

$$\text{return } \frac{\partial \mathbf{e}^{ang-moment}}{\partial \mathbf{p}_L} \in \mathbb{R}^{3N_{tm-com} \times \dim(\mathbf{p}_L)}$$

:task-jacobian

[method]

$$\frac{\partial \mathbf{e}}{\partial \mathbf{q}} = \begin{pmatrix} \dim(\mathbf{e}^{kin}(\mathbf{p}_\theta, \phi)) \\ \dim(\mathbf{e}^{eom-trans}(\mathbf{p}_c, \mathbf{p}_{\hat{w}})) \\ \dim(\mathbf{e}^{eom-rot}(\mathbf{p}_\theta, \mathbf{p}_c, \mathbf{p}_L, \mathbf{p}_{\hat{w}}, \phi)) \\ \dim(\mathbf{e}^{cog}(\mathbf{p}_\theta, \mathbf{p}_c, \phi)) \\ \dim(\mathbf{e}^{ang-moment}(\mathbf{p}_\theta, \mathbf{p}_L, \phi)) \\ \dim(\mathbf{e}^{trq}(\mathbf{p}_\theta, \mathbf{p}_{\hat{w}}, \mathbf{p}_\tau, \phi)) \\ \dim(\mathbf{e}^{posture}(\mathbf{p}_\theta)) \end{pmatrix} \begin{pmatrix} \dim(\mathbf{p}_\theta) & \dim(\mathbf{p}_c) & \dim(\mathbf{p}_L) & \dim(\mathbf{p}_{\hat{w}}) & \dim(\mathbf{p}_\tau) & \dim(\phi) \\ \frac{\partial \mathbf{e}^{kin}}{\partial \mathbf{p}_\theta} & & & & & \frac{\partial \mathbf{e}^{kin}}{\partial \phi} \\ & \frac{\partial \mathbf{e}^{eom-trans}}{\partial \mathbf{p}_c} & & \frac{\partial \mathbf{e}^{eom-trans}}{\partial \mathbf{p}_{\hat{w}}} & & \\ \frac{\partial \mathbf{e}^{eom-rot}}{\partial \mathbf{p}_\theta} & \frac{\partial \mathbf{e}^{eom-rot}}{\partial \mathbf{p}_c} & \frac{\partial \mathbf{e}^{eom-rot}}{\partial \mathbf{p}_L} & \frac{\partial \mathbf{e}^{eom-rot}}{\partial \mathbf{p}_{\hat{w}}} & & \frac{\partial \mathbf{e}^{eom-rot}}{\partial \phi} \\ \frac{\partial \mathbf{e}^{cog}}{\partial \mathbf{p}_\theta} & \frac{\partial \mathbf{e}^{cog}}{\partial \mathbf{p}_c} & & & & \frac{\partial \mathbf{e}^{cog}}{\partial \phi} \\ & & \frac{\partial \mathbf{e}^{ang-moment}}{\partial \mathbf{p}_L} & & & \\ & & & & & \end{pmatrix} \quad (4.340)$$

$$\text{return } \frac{\partial \mathbf{e}}{\partial \mathbf{q}} = \mathbb{R}^{\dim(\mathbf{e}) \times \dim(\mathbf{q})}$$

**:convert-instant-inequality-constraint-matrix-for-control-vector** *key* (*instant-ineq-matrix*) [method]  
 (*num-control-point*)  
 (*dim-instant*)

$\mathbf{x}(t) \in \mathbb{R}^{N_x}$  の各成分を B スプラインで次式のように表現する .

$$x_j(t) = \sum_{i=0}^{N_{ctrl}-1} p_{j,i} b_{i,n}(t) = \mathbf{p}_j^T \mathbf{b}_n(t) \quad (j = 1, 2, \dots, N_x) \quad (4.341)$$

ただし ,

$$\mathbf{p}_j = \begin{pmatrix} p_{j,0} \\ p_{j,1} \\ \vdots \\ p_{j,N_{ctrl}-1} \end{pmatrix} \quad (j = 1, 2, \dots, N_x) \quad (4.342)$$

$$\mathbf{b}_n(t) = \begin{pmatrix} b_{0,n}(t) \\ b_{1,n}(t) \\ \vdots \\ b_{N_{ctrl}-1,n}(t) \end{pmatrix} \quad (4.343)$$

$p_{j,i}$  ,  $b_{i,n}(t)$  をそれぞれ制御点 , B スプライン基底関数と呼ぶ . つまり ,

$$\mathbf{x}(t) = \begin{pmatrix} x_1(t) \\ \vdots \\ x_{N_x}(t) \end{pmatrix} = \begin{pmatrix} \sum_{i=0}^{N_{ctrl}-1} p_{1,i} b_{i,n}(t) \\ \sum_{i=0}^{N_{ctrl}-1} p_{2,i} b_{i,n}(t) \\ \vdots \\ \sum_{i=0}^{N_{ctrl}-1} p_{N_x,i} b_{i,n}(t) \end{pmatrix} = \begin{pmatrix} \mathbf{p}_1^T \mathbf{b}_n(t) \\ \mathbf{p}_2^T \mathbf{b}_n(t) \\ \vdots \\ \mathbf{p}_{N_x}^T \mathbf{b}_n(t) \end{pmatrix} = \mathbf{P} \mathbf{b}_n(t) \quad (4.344)$$

ただし ,

$$\mathbf{P} = \begin{pmatrix} \mathbf{p}_1^T \\ \mathbf{p}_2^T \\ \vdots \\ \mathbf{p}_{N_x}^T \end{pmatrix} = \begin{pmatrix} \tilde{\mathbf{p}}_0 & \tilde{\mathbf{p}}_1 & \cdots & \tilde{\mathbf{p}}_{N_{ctrl}-1} \end{pmatrix} \quad (4.345)$$

$$\tilde{\mathbf{p}}_i = \begin{pmatrix} p_{1,i} \\ p_{2,i} \\ \vdots \\ p_{N_x,i} \end{pmatrix} \quad (i = 0, 1, \dots, N_{ctrl} - 1) \quad (4.346)$$

$$\mathbf{c}^T \tilde{\mathbf{p}}_i \geq d \quad (i = 0, 1, \dots, N_{ctrl} - 1) \quad (4.347)$$

つまり ,

$$\begin{pmatrix} c_1 & c_2 & \cdots & c_{N_x} \end{pmatrix} \begin{pmatrix} p_{1,i} \\ p_{2,i} \\ \vdots \\ p_{N_x,i} \end{pmatrix} = \sum_{j=1}^{N_x} c_j p_{j,i} \geq d \quad (i = 0, 1, \dots, N_{ctrl} - 1) \quad (4.348)$$

のとき ,

$$\mathbf{c}^T \mathbf{x}(t) = \begin{pmatrix} c_1 & c_2 & \cdots & c_{N_x} \end{pmatrix} \begin{pmatrix} \sum_{i=0}^{N_{ctrl}-1} p_{1,i} b_{i,n}(t) \\ \sum_{i=0}^{N_{ctrl}-1} p_{2,i} b_{i,n}(t) \\ \vdots \\ \sum_{i=0}^{N_{ctrl}-1} p_{N_x,i} b_{i,n}(t) \end{pmatrix} \quad (4.349)$$

$$= \sum_{j=1}^{N_x} c_j \sum_{i=0}^{N_{ctrl}-1} p_{j,i} b_{i,n}(t) \quad (4.350)$$

$$= \sum_{i=0}^{N_{ctrl}-1} \left( \sum_{j=1}^{N_x} c_j p_{j,i} \right) b_{i,n}(t) \quad (4.351)$$

$$\geq d \sum_{i=0}^{N_{ctrl}-1} b_{i,n}(t) \quad (4.352)$$

$$= d \quad (4.353)$$

したがって ,

$$\mathbf{C} \mathbf{x}(t) \geq \mathbf{d} \quad (4.354)$$

$$\Leftrightarrow \mathbf{C} \tilde{\mathbf{p}}_i \geq \mathbf{d} \quad (i = 0, 1, \dots, N_{ctrl} - 1) \quad (4.355)$$

$$\Leftrightarrow \begin{matrix} N_{ctrl} & N_{ctrl} & & N_{ctrl} \\ N_{ineq} & \begin{pmatrix} \tilde{\mathbf{C}}_{0,1} & \tilde{\mathbf{C}}_{0,2} & \cdots & \tilde{\mathbf{C}}_{0,N_x} \\ \tilde{\mathbf{C}}_{1,1} & \tilde{\mathbf{C}}_{1,2} & \cdots & \tilde{\mathbf{C}}_{1,N_x} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{\mathbf{C}}_{N_{ctrl}-1,1} & \tilde{\mathbf{C}}_{N_{ctrl}-1,2} & \cdots & \tilde{\mathbf{C}}_{N_{ctrl}-1,N_x} \end{pmatrix} & \begin{pmatrix} \mathbf{p}_1 \\ \mathbf{p}_2 \\ \vdots \\ \mathbf{p}_{N_x} \end{pmatrix} \end{matrix} \geq \begin{pmatrix} \mathbf{d} \\ \mathbf{d} \\ \vdots \\ \mathbf{d} \end{pmatrix} \quad (4.356)$$

$$\Leftrightarrow \tilde{\mathbf{C}} \mathbf{p} \geq \tilde{\mathbf{d}} \quad (4.357)$$

ただし ,

$$\mathbf{C} = \begin{pmatrix} c_1 & c_2 & \cdots & c_{N_x} \end{pmatrix} \in \mathbb{R}^{N_{ineq} \times N_x} \quad (4.358)$$

$i$  番目

$$\tilde{\mathbf{C}}_{i,j} = \begin{pmatrix} \mathbf{0} & \cdots & \mathbf{0} & c_j & \mathbf{0} & \cdots & \mathbf{0} \end{pmatrix} \in \mathbb{R}^{N_{ineq} \times N_{ctrl}} \quad (4.359)$$

$$(i = 0, 1, \dots, N_{ctrl} - 1, j = 1, 2, \dots, N_x) \quad (4.360)$$

この関数は  $\mathbf{C} \in \mathbb{R}^{N_{ineq} \times N_x}$  を受け取り ,  $\tilde{\mathbf{C}} \in \mathbb{R}^{N_{ctrl} N_{ineq} \times N_{ctrl} N_x}$  を返す .

**:convert-instant-inequality-constraint-vector-for-control-vector**  $\mathcal{E}key$  (*instant-ineq-vector*) [method]  
(*num-control-point*)

この関数は  $\mathbf{d} \in \mathbb{R}^{N_{ineq}}$  を受け取り ,  $\tilde{\mathbf{d}} \in \mathbb{R}^{N_{ctrl} N_{ineq}}$  を返す .

**:theta-max-vector**  $\mathcal{E}key$  (*update?* *nil*) [method]

return  $\boldsymbol{\theta}_{max} \in \mathbb{R}^{N_{var-joint}}$

**:theta-min-vector**  $\mathcal{E}key$  (*update?* *nil*) [method]

return  $\boldsymbol{\theta}_{min} \in \mathbb{R}^{N_{var-joint}}$

**:theta-instant-inequality-constraint-matrix**  $\mathcal{E}key$  (*update?* *nil*) [method]



$$\boldsymbol{\theta}_{min} \leq \boldsymbol{\theta} \leq \boldsymbol{\theta}_{max} \quad (4.361)$$

$$\Leftrightarrow \begin{pmatrix} \boldsymbol{I} \\ -\boldsymbol{I} \end{pmatrix} \boldsymbol{\theta} \geq \begin{pmatrix} \boldsymbol{\theta}_{min} \\ -\boldsymbol{\theta}_{max} \end{pmatrix} \quad (4.362)$$

$$\Leftrightarrow \boldsymbol{C}_{\boldsymbol{\theta}} \boldsymbol{\theta} \geq \bar{\boldsymbol{d}}_{\boldsymbol{\theta}} \quad (4.363)$$

$$\text{return } \boldsymbol{C}_{\boldsymbol{\theta}} := \begin{pmatrix} \boldsymbol{I} \\ -\boldsymbol{I} \end{pmatrix} \in \mathbb{R}^{2N_{var-joint} \times N_{var-joint}}$$

**:theta-instant-inequality-constraint-vector** *ℰkey (update? nil)* [method]

$$\text{return } \bar{\boldsymbol{d}}_{\boldsymbol{\theta}} := \begin{pmatrix} \boldsymbol{\theta}_{min} \\ -\boldsymbol{\theta}_{max} \end{pmatrix} \in \mathbb{R}^{2N_{var-joint}}$$

**:theta-control-vector-inequality-constraint-matrix** *ℰkey (update? nil)* [method]

$$\boldsymbol{C}_{\boldsymbol{\theta}} \boldsymbol{\theta} \geq \bar{\boldsymbol{d}}_{\boldsymbol{\theta}} \quad (4.364)$$

$$\Leftrightarrow \boldsymbol{C}_{\boldsymbol{p}_{\boldsymbol{\theta}}} \boldsymbol{p}_{\boldsymbol{\theta}} \geq \bar{\boldsymbol{d}}_{\boldsymbol{p}_{\boldsymbol{\theta}}} \quad (4.365)$$

差分形式で表すと次式となる .

$$\boldsymbol{C}_{\boldsymbol{p}_{\boldsymbol{\theta}}} (\boldsymbol{p}_{\boldsymbol{\theta}} + \Delta \boldsymbol{p}_{\boldsymbol{\theta}}) \geq \bar{\boldsymbol{d}}_{\boldsymbol{p}_{\boldsymbol{\theta}}} \quad (4.366)$$

$$\Leftrightarrow \boldsymbol{C}_{\boldsymbol{p}_{\boldsymbol{\theta}}} \Delta \boldsymbol{p}_{\boldsymbol{\theta}} \geq \bar{\boldsymbol{d}}_{\boldsymbol{p}_{\boldsymbol{\theta}}} - \boldsymbol{C}_{\boldsymbol{p}_{\boldsymbol{\theta}}} \boldsymbol{p}_{\boldsymbol{\theta}} \quad (4.367)$$

$$\Leftrightarrow \boldsymbol{C}_{\boldsymbol{p}_{\boldsymbol{\theta}}} \Delta \boldsymbol{p}_{\boldsymbol{\theta}} \geq \boldsymbol{d}_{\boldsymbol{p}_{\boldsymbol{\theta}}} \quad (4.368)$$

$$\text{return } \boldsymbol{C}_{\boldsymbol{p}_{\boldsymbol{\theta}}}$$

**:theta-control-vector-inequality-constraint-vector-without-offset** *ℰkey (update? nil)* [method]

$$\text{return } \bar{\boldsymbol{d}}_{\boldsymbol{p}_{\boldsymbol{\theta}}}$$

**:theta-control-vector-inequality-constraint-vector** *ℰkey (update? t)* [method]

$$\text{return } \boldsymbol{d}_{\boldsymbol{p}_{\boldsymbol{\theta}}} := \bar{\boldsymbol{d}}_{\boldsymbol{p}_{\boldsymbol{\theta}}} - \boldsymbol{C}_{\boldsymbol{p}_{\boldsymbol{\theta}}} \boldsymbol{p}_{\boldsymbol{\theta}}$$

**:cog-control-vector-inequality-constraint-matrix** [method]

todo

**:cog-control-vector-inequality-constraint-vector** [method]

todo

**:ang-moment-max-vector** *ℰkey (update? nil)* [method]

$$\text{return } \boldsymbol{L}_{max} \in \mathbb{R}^3$$

**:ang-moment-instant-inequality-constraint-matrix** *ℰkey (update? nil)* [method]

$$-\boldsymbol{L}_{max} \leq \boldsymbol{L} \leq \boldsymbol{L}_{max} \quad (4.369)$$

$$\Leftrightarrow \begin{pmatrix} \boldsymbol{I} \\ -\boldsymbol{I} \end{pmatrix} \boldsymbol{L} \geq \begin{pmatrix} -\boldsymbol{L}_{max} \\ -\boldsymbol{L}_{max} \end{pmatrix} \quad (4.370)$$

$$\Leftrightarrow \boldsymbol{C}_L \boldsymbol{L} \geq \bar{\boldsymbol{d}}_L \quad (4.371)$$

$$\text{return } \boldsymbol{C}_L := \begin{pmatrix} \boldsymbol{I} \\ -\boldsymbol{I} \end{pmatrix} \in \mathbb{R}^{6 \times 3}$$

**:ang-moment-instant-inequality-constraint-vector** *ℰkey (update? nil)* [method]  
 return  $\bar{\mathbf{d}}_L := \begin{pmatrix} -\mathbf{L}_{max} \\ -\mathbf{L}_{max} \end{pmatrix} \in \mathbb{R}^6$

**:ang-moment-control-vector-inequality-constraint-matrix** *ℰkey (update? nil)* [method]

$$\mathbf{C}_L \mathbf{L} \geq \bar{\mathbf{d}}_L \quad (4.372)$$

$$\Leftrightarrow \mathbf{C}_{p_L} p_L \geq \bar{\mathbf{d}}_{p_L} \quad (4.373)$$

差分形式で表すと次式となる .

$$\mathbf{C}_{p_L} (p_L + \Delta p_L) \geq \bar{\mathbf{d}}_{p_L} \quad (4.374)$$

$$\Leftrightarrow \mathbf{C}_{p_L} \Delta p_L \geq \bar{\mathbf{d}}_{p_L} - \mathbf{C}_{p_L} p_L \quad (4.375)$$

$$\Leftrightarrow \mathbf{C}_{p_L} \Delta p_L \geq \mathbf{d}_{p_L} \quad (4.376)$$

return  $\mathbf{C}_{p_L}$

**:ang-moment-control-vector-inequality-constraint-vector-without-offset** *ℰkey (update? nil)* [method]  
 return  $\bar{\mathbf{d}}_{p_L}$

**:ang-moment-control-vector-inequality-constraint-vector** *ℰkey (update? t)* [method]  
 return  $\mathbf{d}_{p_L} := \bar{\mathbf{d}}_{p_L} - \mathbf{C}_{p_L} p_L$

**:wrench-control-vector-inequality-constraint-matrix** [method]  
 todo

**:wrench-control-vector-inequality-constraint-vector** [method]  
 todo

**:torque-control-vector-inequality-constraint-matrix** [method]  
 todo

**:torque-control-vector-inequality-constraint-vector** [method]  
 todo

**:phi-max-vector** *ℰkey (update? nil)* [method]  
 return  $\phi_{max} \in \mathbb{R}^{N_{invar-joint}}$

**:phi-min-vector** *ℰkey (update? nil)* [method]  
 return  $\phi_{min} \in \mathbb{R}^{N_{invar-joint}}$

**:phi-inequality-constraint-matrix** *ℰkey (update? nil)* [method]

$$\phi_{min} \leq \phi + \Delta \phi \leq \phi_{max} \quad (4.377)$$

$$\Leftrightarrow \begin{pmatrix} \mathbf{I} \\ -\mathbf{I} \end{pmatrix} \Delta \phi \geq \begin{pmatrix} \phi_{min} - \phi \\ -(\phi_{max} - \phi) \end{pmatrix} \quad (4.378)$$

$$\Leftrightarrow \mathbf{C}_\phi \Delta \phi \geq \mathbf{d}_\phi \quad (4.379)$$

return  $\mathbf{C}_\phi := \begin{pmatrix} \mathbf{I} \\ -\mathbf{I} \end{pmatrix} \in \mathbb{R}^{2N_{invar-joint} \times N_{invar-joint}}$

**:phi-inequality-constraint-vector** *ℰkey (update? t)* [method]

return  $\mathbf{d}_\phi := \begin{pmatrix} \phi_{min} - \phi \\ -(\phi_{max} - \phi) \end{pmatrix} \in \mathbb{R}^{2N_{invar-joint}}$

**:config-inequality-constraint-matrix** [method]

$$\begin{cases} \mathbf{C}_{p_\theta} \Delta \mathbf{p}_\theta \geq \mathbf{d}_{p_\theta} \\ \mathbf{C}_{p_c} \Delta \mathbf{p}_c \geq \mathbf{d}_{p_c} \\ \mathbf{C}_{p_L} \Delta \mathbf{p}_L \geq \mathbf{d}_{p_L} \\ \mathbf{C}_{p_{\hat{w}}} \Delta \mathbf{p}_{\hat{w}} \geq \mathbf{d}_{p_{\hat{w}}} \\ \mathbf{C}_{p_\tau} \Delta \mathbf{p}_\tau \geq \mathbf{d}_{p_\tau} \\ \mathbf{C}_\phi \Delta \phi \geq \mathbf{d}_\phi \end{cases} \quad (4.380)$$

$$\Leftrightarrow \begin{pmatrix} \mathbf{C}_{p_\theta} & & & & & \\ & \mathbf{C}_{p_c} & & & & \\ & & \mathbf{C}_{p_L} & & & \\ & & & \mathbf{C}_{p_{\hat{w}}} & & \\ & & & & \mathbf{C}_{p_\tau} & \\ & & & & & \mathbf{C}_\phi \end{pmatrix} \begin{pmatrix} \Delta \mathbf{p}_\theta \\ \Delta \mathbf{p}_c \\ \Delta \mathbf{p}_L \\ \Delta \mathbf{p}_{\hat{w}} \\ \Delta \mathbf{p}_\tau \\ \Delta \phi \end{pmatrix} \geq \begin{pmatrix} \mathbf{d}_{p_\theta} \\ \mathbf{d}_{p_c} \\ \mathbf{d}_{p_L} \\ \mathbf{d}_{p_{\hat{w}}} \\ \mathbf{d}_{p_\tau} \\ \mathbf{d}_\phi \end{pmatrix} \quad (4.381)$$

$$\Leftrightarrow \mathbf{C} \Delta \mathbf{q} \geq \mathbf{d} \quad (4.382)$$

return  $\mathbf{C}$

**:config-inequality-constraint-vector** [method]

return  $\mathbf{d}$

**:config-equality-constraint-matrix** *ℰkey (update? nil)* [method]

return  $\mathbf{A} \in \mathbb{R}^{0 \times \dim(\mathbf{q})}$  (no equality constraint)

**:config-equality-constraint-vector** *ℰkey (update? t)* [method]

return  $\mathbf{b} \in \mathbb{R}^0$  (no equality constraint)

**:regular-matrix** [method]

$$\mathbf{W}_{reg} := \min(k_{max}, \|\mathbf{e}\|^2 + k_{off}) \mathbf{I} + k_{trq} \mathbf{W}_{trq} \quad (4.383)$$

return  $\mathbf{W}_{reg} \in \mathbb{R}^{\dim(\mathbf{q}) \times \dim(\mathbf{q})}$

**:regular-vector** [method]

$$\mathbf{v}_{reg} := k_{trq} \mathbf{v}_{trq} \quad (4.384)$$

return  $\mathbf{v}_{reg} \in \mathbb{R}^{\dim(\mathbf{q})}$

**:update-collision-inequality-constraint** [method]

Not implemented yet.

**:update-viewer** *ℰkey (start-time (send \_theta-bst :start-time))* [method]

*(finish-time (send \_theta-bst :finish-time))*

*(delta-time (/ (- finish-time start-time) 100.0))*

Update viewer.

**:print-setting-information** [method]

Print setting information.

**:print-status** [method]

Print status.

**:play-animation** *ℰkey* (*robot-env*) [method]

(*start-time* (*send* *\_theta-bst* *:start-time*))  
 (*finish-time* (*send* *\_theta-bst* *:finish-time*))  
 (*delta-time* (/ (- *finish-time* *start-time*) 100.0))  
 (*loop?* *t*)  
 (*visualize-callback-func*)

Play motion animation.

**:dump-graph-pdf** *ℰkey* (*start-time* (*send* *\_theta-bst* *:start-time*)) [method]

(*finish-time* (*send* *\_theta-bst* *:finish-time*))  
 (*delta-time* (/ (- *finish-time* *start-time*) 100.0))  
 (*dat-filename* /tmp/bspline-dynamic-configuration-task-graph.dat)  
 (*dump-filename* (*ros::resolve-ros-path* *package://eus\_qp/optmotiongen/logs/bspline-dynam*

Plot graph.

## 5 補足

### 5.1 既存のロボット基礎クラスの拡張

**joint** [class]

**:super** **propertied-object**  
**:slots** (parent-link)  
 (child-link)  
 (joint-angle)  
 (min-angle)  
 (max-angle)  
 (default-coords)  
 (joint-velocity)  
 (joint-acceleration)  
 (joint-torque)  
 (max-joint-velocity)  
 (max-joint-torque)  
 (joint-min-max-table)  
 (joint-min-max-target)

**:child-link** *ℰrest args* [method]

Returns child link of this joint. If any arguments is set, it is passed to the child-link.  
 Override to support the case that child-link is cascaded-link instantiate. Return the root link of child cascaded-link instantiate in that case.

**:axis-vector** [method]

Return joint axis vector. Represented in world coordinates.

return  $\mathbf{a}_i \in \mathbb{R}^3$

**:pos** [method]

Return joint position. Represented in world coordinates.

return  $\mathbf{p}_i \in \mathbb{R}^3$

**bodyset-link** [class]

<b>:super</b>	<b>bodyset</b>
<b>:slots</b>	(rot)
	(pos)
	(parent)
	(descendants)
	(worldcoords)
	(manager)
	(changed)
	(geometry::bodies)
	(joint)
	(parent-link)
	(child-links)
	(analysis-level)
	(default-coords)
	(weight)
	(acentroid)
	(inertia-tensor)
	(angular-velocity)
	(angular-acceleration)
	(spacial-velocity)
	(spacial-acceleration)
	(momentum-velocity)
	(angular-momentum-velocity)
	(momentum)
	(angular-momentum)
	(force)
	(moment)
	(ext-force)
	(ext-moment)

**:mg** [method]

return  $mg = \|\mathbf{mg}\|$

**:mg-vec** [method]  
 return *mg*

**cascaded-link** [class]

:super **cascaded-coords**  
 :slots (rot)  
 (pos)  
 (parent)  
 (descendants)  
 (worldcoords)  
 (manager)  
 (changed)  
 (links)  
 (joint-list)  
 (bodies)  
 (collision-avoidance-links)  
 (end-coords-list)

**:calc-jacobian-from-joint-list** *key* (*union-joint-list*) [method]  
*(move-target)*  
*(joint-list (mapcar #'(lambda (mt) (send-all (send self :link-list (send mt :parent) (transform-coords (mapcar #'(lambda (mt) (make-coords)) move-target)) (translation-axis (mapcar #'(lambda (mt) t) move-target)) (rotation-axis (mapcar #'(lambda (mt) t) move-target))*

*union-joint-list* list of all joints considered in jacobian. column num of jacobian is same with length of union-joint-list.

*move-target* list of move-target.

*joint-list* list of joint-list which is contained in each chain of move-target.

*transform-coords* list of transform-coords of each move-target.

*translation-axis* list of translation-axis of each move-target.

*rotation-axis* list of rotation-axis of each move-target.

Get jacobian matrix from following two information: (1) union-joint-list and (2) list of move-target. One recession compared with :calc-jacobian-from-link-list is that child-reverse is not supported. (Only not implemented yet because I do not need such feature in current application.)

**:calc-cog-jacobian-from-joint-list** *key* (*union-joint-list*) [method]  
*(update-mass-properties t)*  
*(translation-axis :z)*

*union-joint-list* list of all joints considered in jacobian. column num of jacobian is same with length of union-joint-list.

Get CoG jacobian matrix from union-joint-list.

**:find-link-route** *to* *Optional from* [method]

Override to support the case that joint does not exist between links. Change from (send to :parent-link) to (send to :parent).

## 5.2 環境と接触するロボットの関節・リンク構造

### 2d-planar-contact

[class]

```
:super    cascaded-link
:slots    (_contact-coords  $T_{cnt}$ )
           (_contact-pre-coords  $T_{cnt-pre}$ )
```

二次元平面上の長方形領域での接触座標を表す仮想の関節・リンク構造。

```
:init  $\mathcal{E}key$  (name contact) [method]
      (contact-pre-offset 100)
```

Initialize instance

```
:contact-coords  $\mathcal{E}rest$  args [method]
  return  $T_{cnt} := \{\mathbf{p}_{cnt}, \mathbf{R}_{cnt}\}$ 
```

```
:contact-pre-coords  $\mathcal{E}rest$  args [method]
  return  $T_{cnt-pre} := \{\mathbf{p}_{cnt-pre}, \mathbf{R}_{cnt-pre}\}$ 
```

```
:set-from-face  $\mathcal{E}key$  (face) [method]
                (margin 150.0)
```

set coords and min/max joint angle from face.

### robot-environment

[class]

```
:super    cascaded-link
:slots    (_robot  $\mathcal{R}$ )
           (_robot-with-root-virtual  $\hat{\mathcal{R}}$ )
           (_root-virtual-joint-list list of root virtual joint)
           (_contact-list  $\{\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_{N_C}\}$ )
           (_variant-joint-list  $\mathcal{J}_{var}$ )
           (_invariant-joint-list  $\mathcal{J}_{invar}$ )
           (_drive-joint-list  $\mathcal{J}_{drive}$ )
```

ロボットとロボット・環境間の接触のクラス。

以下を合わせた関節・リンク構造に関するメソッドが定義されている。

1. 浮遊ルートリンクのための仮想関節付きのロボットの関節
2. 接触位置を定める仮想関節

関節・リンク構造を定めるために、初期化時に以下を与える

robot  $\mathcal{R}$  ロボット (cascaded-link クラスのインスタンス)。

**contact-list**  $\{\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_{N_C}\}$  接触 (2d-planar-contact クラスなどのインスタンス) のリスト .

ロボット  $R$  に , 浮遊ルートリンクの変位に対応する仮想関節を付加した仮想関節付きロボット  $\hat{\mathcal{R}}$  を内部で保持する .

<b>:init</b> $\mathcal{E}key$ ( <i>robot</i> )	[method]
( <i>contact-list</i> )	
( <i>root-virtual-mode :6dof</i> )	
( <i>root-virtual-joint-class-list</i> )	
( <i>root-virtual-joint-axis-list</i> )	
Initialize instance	
<b>:dissoc-root-virtual</b>	[method]
dissoc root virtual parent/child structure.	
<b>:init-pose</b>	[method]
set zero joint angle.	
<b>:robot</b> $\mathcal{E}rest$ <i>args</i>	[method]
return $\mathcal{R}$	
<b>:robot-with-root-virtual</b> $\mathcal{E}rest$ <i>args</i>	[method]
return $\hat{\mathcal{R}}$	
<b>:contact-list</b> $\mathcal{E}rest$ <i>args</i>	[method]
return $\{\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_{N_C}\}$	
<b>:contact</b> <i>name</i> $\mathcal{E}rest$ <i>args</i>	[method]
return $\mathcal{C}_i$	
<b>:variant-joint-list</b> $\mathcal{E}optional$ ( <i>jl :nil</i> )	[method]
return $\mathcal{J}_{var}$	
<b>:invariant-joint-list</b> $\mathcal{E}optional$ ( <i>jl :nil</i> )	[method]
return $\mathcal{J}_{invar}$	
<b>:drive-joint-list</b> $\mathcal{E}optional$ ( <i>jl :nil</i> )	[method]
return $\mathcal{J}_{drive}$	
<b>:root-virtual-joint-list</b>	[method]
return list of root virtual joint	

### 5.3 irteus の inverse-kinematics 互換関数

<b>cascaded-link</b>	[class]
:super	<b>cascaded-coords</b>
:slots	(rot)
	(pos)
	(parent)
	(descendants)



```

(worldcoords)
(manager)
(changed)
(links)
(joint-list)
(bodies)
(collision-avoidance-links)
(end-coords-list)

```

```

:inverse-kinematics-optmotiongen target-coords key (stop 50) [method]
                                (link-list)
                                (move-target)
                                (debug-view)
                                (revert-if-fail t)
                                (transform-coords target-coords)
                                (translation-axis (cond ((atom move-target) t) (t (make-list 3 0))))
                                (rotation-axis (cond ((atom move-target) t) (t (make-list 3 0))))
                                (thre (cond ((atom move-target) 1) (t (make-list (length move-target) 0))))
                                (rthre (cond ((atom move-target) (deg2rad 1)) (t (make-list (length move-target) 0))))
                                (collision-avoidance-link-pair :nil)
                                (collision-distance-limit 10.0)
                                (obstacles)
                                (min-loop)
                                (root-virtual-mode :fix)
                                (joint-angle-margin 0.0)
                                (posture-joint-list)
                                (posture-joint-angle-list)
                                (target-posture-scale 0.001)
                                (norm-regular-scale-max 0.01)
                                (norm-regular-scale-offset 1.000000e-07)
                                (pre-process-func)
                                (post-process-func)
                                allow-other-keys

```

Solve inverse kinematics problem with sqp optimization. `;;` target-coords, move-target, rotation-axis, translation-axis `;;` -`;` both list and atom OK. target-coords : The coordinate of the target that returns coordinates. Use a list of targets to solve the IK relative to multiple end links simultaneously. Function is not available to target-coords. link-list : List of links to control. When the target-coords is list, this should be a list of lists. move-target : Specify end-effector coordinate. When the target-coords is list, this should be list too. stop : Maximum number for IK iteration. Default is 50. debug-view : Set t to show debug message and visualization. Use :no-message to just show the irtview image. Default is nil. revert-if-fail : Set nil to keep the angle posture of IK solve iteration. Default is t, which return to original position when IK fails. translation-axis : :x :y :z for constraint along the x, y, z axis. :xy :yz :zx for plane. Default is t. rotation-axis : Use nil for position only IK. :x, :y, :z for the constraint around axis with plus direction. When the target-coords is list, this should be list too. Default is t.



Solve inverse kinematics problem with `sqp` optimization. `target-coords-list` : The coordinate of the target that returns coordinates. Use a list of targets to solve the IK relative to multiple end links simultaneously. Function is not available to `target-coords`. `move-target-list` : Specify end-effector coordinate. When the `target-coords` is list, this should be list too. `stop` : Maximum number for IK iteration. Default is 50. `debug-view` : Set `t` to show debug message and visualization. Use `:no-message` to just show the `irtview` image. Default is `nil`. `revert-if-fail` : Set `nil` to keep the angle posture of IK solve iteration. Default is `t`, which return to original position when IK fails. `translation-axis-list` : `:x` `:y` `:z` for constraint along the x, y, z axis. `:xy` `:yz` `:zx` for plane. Default is `t`. `rotation-axis-list` : Use `nil` for position only IK. `:x`, `:y`, `:z` for the constraint around axis with plus direction. When the `target-coords` is list, this should be list too. Default is `t`. `thre` : Threshold for position error to terminate IK iteration. Default is 1 [mm]. `rthre` : Threshold for rotation error to terminate IK iteration. Default is 0.017453 [rad] (1 deg).

## robot-model

[class]

```

:super      cascaded-link
:slots      (rot)
              (pos)
              (parent)
              (descendants)
              (worldcoords)
              (manager)
              (changed)
              (links)
              (joint-list)
              (bodies)
              (collision-avoidance-links)
              (end-coords-list)
              (larm-end-coords)
              (rarm-end-coords)
              (lleg-end-coords)
              (rleg-end-coords)
              (head-end-coords)
              (torso-end-coords)
              (larm-root-link)
              (rarm-root-link)
              (lleg-root-link)
              (rleg-root-link)
              (head-root-link)
              (torso-root-link)
              (larm-collision-avoidance-links)
              (rarm-collision-avoidance-links)
              (larm)
              (rarm)
              (lleg)
              (rleg)
              (torso)
              (head)

```

(force-sensors)  
 (imu-sensors)  
 (cameras)  
 (support-polygons)

**:limb** *limb method ℰrest args* [method]  
 Extend to support to call :inverse-kinematics-optmotiongen.

**contact-ik-arg** [class]  
 :super **cascaded-link**  
 :slots (*\_contact-coords*  $T_{cnt}$ )

inverse-kinematics-optmotiongen の *target-coords*, *translation-axis*, *rotation-axis*, *transform-coords* 5 | 数に対応する接触座標を表す仮想の関節・リンク構造 .

**:init** *ℰkey (target-coords)* [method]  
           (*translation-axis*)  
           (*rotation-axis*)  
           (*transform-coords target-coords*)  
           (*name (send target-coords :name)*)  
 Initialize instance

**:contact-coords** *ℰrest args* [method]  
 return  $T_{cnt} := \{\mathbf{p}_{cnt}, \mathbf{R}_{cnt}\}$

**ik-arg-axis->axis-list** *ik-arg-axis* [function]  
 Convert translation-axis / rotatoin-axis to axis list.

**generate-contact-ik-arg-from-rect-face** *ℰkey (rect-face)* [function]  
                                   (*name (send rect-face :name)*)  
                                   (*margin (or (send rect-face :get :margin) 0)*)  
 Generate contact-ik-arg instance from rectangle face.

**generate-contact-ik-arg-from-line-segment** *ℰkey (line-seg)* [function]  
                                   (*name (send line-seg :name)*)  
                                   (*margin (or (send line-seg :get :margin) 0)*)  
 Generate contact-ik-arg instance from line segment.

**axis->index** *axis* [function]  
**axis->sgn** *axis* [function]

## 5.4 関節トルク勾配の計算

**get-link-jacobian-for-contact-torque** *ℰkey (robot)* [function]

(drive-joint-list)  
 (contact-coords)  
 (contact-parent-link)

*contact-coords* に対応する接触部位の番号を  $m$  とする．*contact-coords* の位置姿勢を  $\mathbf{r}_m \in \mathbb{R}^6$  , *drive-joint-list* の関節角度ベクトルを  $\boldsymbol{\psi} \in \mathbb{R}^{N_{\text{drive-joint}}}$  として，次式を満たすヤコビ行列  $\mathbf{J}_m$  を返す．

$$\dot{\mathbf{r}}_m = \mathbf{J}_m \dot{\boldsymbol{\psi}} \quad (5.1)$$

$$= \sum_{i=1}^{N_{\text{drive-joint}}} \mathbf{j}_m^{(i)} \dot{\psi}_i \quad (5.2)$$

$$\mathbf{j}_m^{(i)} = \begin{pmatrix} \mathbf{a}_{\psi_i} \times (\mathbf{p}_m - \mathbf{p}_{\psi_i}) \\ \mathbf{a}_{\psi_i} \end{pmatrix} \quad (5.3)$$

$$\mathbf{J}_m = \begin{pmatrix} \mathbf{j}_m^{(1)} & \mathbf{j}_m^{(2)} & \cdots & \mathbf{j}_m^{(N_{\text{drive-joint}})} \end{pmatrix} \quad (5.4)$$

$\mathbf{a}_{\psi_i}, \mathbf{p}_{\psi_i} \in \mathbb{R}^3$  は  $i$  番目の関節の回転軸ベクトルと位置である．

return  $\mathbf{J}_m \in \mathbb{R}^{6 \times N_{\text{drive-joint}}}$

**get-contact-torque** *ℰkey (robot)* [function]  
 (drive-joint-list)  
 (wrench-list)  
 (contact-target-coords-list)  
 (contact-attention-coords-list)

ロボットの接触部位に加わる接触レンチによって生じる関節トルク  $\boldsymbol{\tau}^{cnt}$  は，以下で得られる．

$$\boldsymbol{\tau}^{cnt} = \sum_{m=1}^{N_{cnt}} \mathbf{J}_m^T \mathbf{w}_m \quad (5.5)$$

$\mathbf{w}_m$  は  $m$  番目の接触部位で受ける接触レンチである．

return  $\boldsymbol{\tau}^{cnt} \in \mathbb{R}^{N_{\text{drive-joint}}}$

**get-contact-torque-jacobian** *ℰkey (robot)* [function]  
 (joint-list)  
 (drive-joint-list)  
 (wrench-list)  
 (contact-target-coords-list)  
 (contact-attention-coords-list)

式 (5.4) の  $\mathbf{J}_m$  を以下のように分解して利用する．

$$\mathbf{J}_m = \begin{pmatrix} \mathbf{j}_{x,m}^T \\ \mathbf{j}_{y,m}^T \\ \mathbf{j}_{z,m}^T \\ \mathbf{j}_{R,m}^T \\ \mathbf{j}_{P,m}^T \\ \mathbf{j}_{Y,m}^T \end{pmatrix} = \begin{pmatrix} \left( \frac{\partial \mathbf{r}_{x,m}}{\partial \boldsymbol{\psi}} \right)^T \\ \left( \frac{\partial \mathbf{r}_{y,m}}{\partial \boldsymbol{\psi}} \right)^T \\ \left( \frac{\partial \mathbf{r}_{z,m}}{\partial \boldsymbol{\psi}} \right)^T \\ \left( \frac{\partial \mathbf{r}_{R,m}}{\partial \boldsymbol{\psi}} \right)^T \\ \left( \frac{\partial \mathbf{r}_{P,m}}{\partial \boldsymbol{\psi}} \right)^T \\ \left( \frac{\partial \mathbf{r}_{Y,m}}{\partial \boldsymbol{\psi}} \right)^T \end{pmatrix} \quad (5.6)$$





つまり

$$\frac{\partial \boldsymbol{\tau}^{grav}}{\partial \boldsymbol{\theta}} = \left[ \sum_{k=1}^{N_{gravity-link}} m_k g \frac{\partial^2 p_{cog,z,k}}{\partial \psi_i \partial \theta_j} \right]_{i=1, \dots, N_{drive-joint}, j=1, \dots, N_{joint}} \quad (5.27)$$

return  $\frac{\partial \boldsymbol{\tau}^{grav}}{\partial \boldsymbol{\theta}} \in \mathbb{R}^{N_{drive-joint} \times N_{joint}}$