Springhead2 の IK 実装について

三武 裕玄

2009 年 6 月 26 日

1　はじめに

Springhead2 は，剛体・関節モデルについて，ヤコビアンを用いた IK を搭載している．そのアルゴリズムと実装について解説する．

2 節ではヤコビアンを用いた IK の一般概念を軽く解説する．3 節では実際に使用するヤコビアンの算出方法について述べる．4 節ではヤコビアンと目標位置の微少変化量から関節角度の微小変化量を求める手法 (Gauss-Seidel 法) の計算式について述べる．5 節で IK で計算された関節角に向けて実際に関節を動作させる際の手法とそれによる IK の機能拡張について述べる．

2　IK:一般

IK (Inverse Kinematics, 逆運動学) は，リンク構造において，エンドエフェクタの特定の位置・姿勢を実現するように関節角を求める手法である．全てのエンドエフェクタの目標位置・姿勢を並べたベクトルを $\boldsymbol{x}$，全ての関節の角度を並べたベクトルを $\boldsymbol{\theta}$ とすると，与えられた $\boldsymbol{x}$ から $\boldsymbol{\theta}$ を算出する事である．

本稿では，目標 $\boldsymbol{x}$ を与えるものを**制御点**，制御点が目標を達成するために使用できる動作要素を**関節**と呼称する．制御点 (PHIKControlPoint) は，剛体の位置の指定・剛体の姿勢の指定などである．関節 (PHIKNode) は，1 軸関節 (HingeJoint)，3 軸関節 (BallJoint)，剛体の並進移動などである[*1]．

$\boldsymbol{x}, \boldsymbol{\theta}$ の微少変化量 $\boldsymbol{v}, \boldsymbol{\omega}$ を用いると，$\boldsymbol{v}$ と $\boldsymbol{\omega}$ の関係はヤコビアン J を用いて行列の式で記述できる．

$$J\boldsymbol{\omega} = \boldsymbol{v}$$

J は関節角の微少変化 $\boldsymbol{\omega}$ がもたらす制御点の微少位置変化 $\boldsymbol{v}$ を記述する行列であり，$\boldsymbol{\theta}$ に依存する．J が求まると，擬似逆行列 $J^\sharp$ を用いて 目標の $\boldsymbol{v}$ を達成する $\boldsymbol{\omega}$ を求めることができる．

$$\boldsymbol{\omega} = J^\sharp \boldsymbol{v}$$

ヤコビアンを用いた IK では，以下の処理を繰り返す事で最終的に制御点が目標位置 $\boldsymbol{x}$ を達成するような目標関節角 $\boldsymbol{\theta}$ を得る．

1. ヤコビアン J の計算
2. 制御点を目標位置 $\boldsymbol{x}$ に至らせる微小変化 $\boldsymbol{v}$ の計算
3. 擬似逆行列解 $\boldsymbol{\omega}$ の計算
4. 関節角 $\boldsymbol{\theta}$ を $\boldsymbol{\omega}$ だけ変化

3　ヤコビアン

制御点・関節がそれぞれ複数個ある場合，ヤコビアンはブロック構造となる．

$$\begin{pmatrix} J_{c_0}^{n_0} & J_{c_0}^{n_1} & \dots & J_{c_0}^{n_x} \\ J_{c_1}^{n_0} & J_{c_1}^{n_1} & \dots & J_{c_1}^{n_x} \\ \vdots & \vdots & \ddots & \vdots \\ J_{c_y}^{n_0} & J_{c_y}^{n_1} & \dots & J_{c_y}^{n_x} \end{pmatrix} \tag{1}$$

$J_{c_y}^{n_x}$ は，関節 n_x のみを動かした場合の制御点 c_y の位置の微小変化を記述したヤコビアンである．以下に具体的な計算法を示す．なお，関節 n_x のみを動かした場合に制御点 c_y の位置が変化しない場合 (関節 n_x の先に制御点 c_y を有するリンクがつながっていない場合) には $J_{c_y}^{n_x} = \boldsymbol{O}$ とする必要がある．

■1 軸関節 (HingeJoint) と位置制御点の場合

関節の回転軸方向を表すベクトル (長さ 1) を $\boldsymbol{e_{nx}}$，関節の回転中心からエンドエフェクタの位置を指すベクトルを $\boldsymbol{r}_{(\boldsymbol{c_y},\boldsymbol{n_x})}$ とすると，$J_{c_y}^{n_x}$ は 3×1 行列 (2) となる．

$$J_{c_y}^{n_x} = \boldsymbol{e_{nx}} \times \boldsymbol{r}_{(\boldsymbol{c_y},\boldsymbol{n_x})} \tag{2}$$

■3 軸関節 (BallJoint) と位置制御点の場合

3 軸関節については特殊な取り扱いを要する．

関節の回転中心からエンドエフェクタの位置を指すベクトルを $\boldsymbol{r}_{(\boldsymbol{c_y},\boldsymbol{n_x})}$ とすると，単純に考えた場合 $J_{c_y}^{n_x}$ は 3×3 行列 (3) となる．

$$J_{c_y}^{n_x} = \left[\boldsymbol{r}_{(\boldsymbol{c_y},\boldsymbol{n_x})}\times\right] \tag{3}$$

[*1] 剛体の並進移動は 2009 年 6 月 26 日時点でまだ実装されていない

ここで，

$$[\boldsymbol{r}\times] = \begin{pmatrix} 0 & -r_z & r_y \\ r_z & 0 & -r_x \\ -r_y & r_x & 0 \end{pmatrix} \tag{4}$$

である (この行列は 3 次元ベクトルのクロス積を与える).

しかし，連立方程式 $J\boldsymbol{\omega} = \boldsymbol{v}$ を解くために Gauss-Seidel 法 (詳細は「ソルバ」の節で記述) を用いるため，繰り返し計算の収束のためにはヤコビアン J に対して J^tJ が正定値対称である必要がある．一方で，上記の定義による $J^{n_x}_{c_y}$ を用いると $J^{n_x\,t}_{c_y} J^{n_x}_{c_y}$ は固有値を 2 つしか持たず，正定値性を満たさない[*2].

$$J^tJ = \begin{pmatrix} r_z^2 + r_y^2 & -r_xr_y & -r_xr_z \\ -r_xr_y & r_z^2 + r_x^2 & -r_zr_y \\ -r_xr_z & -r_yr_z & r_y^2 + r_x^2 \end{pmatrix} \tag{5}$$

これは，関節と制御点を結ぶ軸まわりの回転は制御点の位置を変えない，ということに起因する．そこで，関節と制御点を結ぶ軸に垂直な 2 軸を可動回転軸とする．制御点の位置を $\boldsymbol{r_{cy}}$、関節の回転中心位置を $\boldsymbol{r_{nx}}$，ワールド座標系の x, y 軸を $\boldsymbol{e_x}, \boldsymbol{e_y}$ として，求める回転軸は以下の通りである．

$$\text{非回転軸}: \boldsymbol{e_0} = \frac{(\boldsymbol{r_{cy}} - \boldsymbol{r_{nx}})}{\|\boldsymbol{r_{cy}} - \boldsymbol{r_{nx}}\|} \tag{6}$$

$$\boldsymbol{e''_1} = \begin{cases} \boldsymbol{e_x}(\boldsymbol{e_0} \not\approx \boldsymbol{e_x}) \\ \boldsymbol{e_y}(else) \end{cases} \tag{7}$$

$$\text{直交化}: \boldsymbol{e'_1} = \boldsymbol{e''_1} - (\boldsymbol{e''_1} \cdot \boldsymbol{e_0})\boldsymbol{e_0} \tag{8}$$

$$\text{回転軸 1}: \boldsymbol{e_1} = \frac{\boldsymbol{e'_1}}{\|\boldsymbol{e'_1}\|} \tag{9}$$

$$\text{回転軸 2}: \boldsymbol{e_2} = \boldsymbol{e_1} \times \boldsymbol{e_0} \tag{10}$$

これらの回転軸 1,2 を用い，3×2 行列 (11) をヤコビアンとして用いる．

ヤコビアン：

$$J^{n_x}_{c_y} = \left(\left[\boldsymbol{e_1} \times (\boldsymbol{r_{cy}} - \boldsymbol{r_{nx}}) \right] \quad \left[\boldsymbol{e_2} \times (\boldsymbol{r_{cy}} - \boldsymbol{r_{nx}}) \right] \right) \tag{11}$$

このヤコビアンを用いる場合，関節角の表現も 2 次元ベクトル $(\omega_1, \omega_2)^t$ とする必要がある．

なお，同じ関節に姿勢制御点が接続された場合はヤコビアンとして 3×3 行列 (3) を使用し，関節角の表現も 3 次元ベクトル $(\omega_1, \omega_2, \omega_3)^t$ を使用する必要がある．

■3 軸関節 (BallJoint) と姿勢制御点の場合

$J^{n_x}_{c_y}$ は 3×3 の単位行列 $\boldsymbol{E}$ である．

[*2] この点についてはきちんと調べたわけではないので確実ではないが，その傾向がある事は言えそうである

4 ソルバ

ソルバは与えられたヤコビアン J および制御点の目標速度 $\boldsymbol{v}$ に基づいて式 (12) の解 $\boldsymbol{\omega}$ を計算する．

$$J\boldsymbol{\omega} = \boldsymbol{v} \tag{12}$$

4.1 非正方の係数行列を持つ連立方程式の Gauss-Seidel 法

求解には Gauss-Seidel 法を用いる．Gauss-Seidel 法では係数行列が正方行列である必要があるが，ヤコビアン J は一般に非正方行列である．そこで式 (12) の両辺に J^t を掛け，式 (13) に Gauss-Seidel 法を適用する．

$$\underbrace{(J^tJ)}_{=A}\boldsymbol{\omega} = \underbrace{(J^t\boldsymbol{v})}_{=\boldsymbol{b}} \tag{13}$$

Gauss-Seidel 法の更新式を式 (14) に示す.

Gauss-Seidel 法:

$$\omega_i = \frac{1}{A_{ii}} \left(b_i - \sum_{j=1(j\neq i)}^{A\,\text{行数}} A_{ij}\omega_j \right) \tag{14}$$

$A, \boldsymbol{b}$ の定義から

$$A_{ij} = \sum_{k=1}^{J\,\text{行数}} J_{ki}J_{kj} \tag{15}$$

$$b_i := \sum_{k=1}^{J\,\text{行数}} J_{ki}v_k \tag{16}$$

これらを式 (14) に代入すると式 (17) を得る．

非正方行列の Gauss-Seidel 法:

$$\omega_i = \underbrace{\frac{1}{\sum_{k=1}^{J\,\text{行数}} {J_{ki}}^2}}_{\alpha} \left(\underbrace{\sum_{k=1}^{J\,\text{行数}} (J_{ki}v_k)}_{\beta} - \sum_{j=1(j\neq i)}^{J\,\text{列数}} \underbrace{\sum_{k=1}^{J\,\text{行数}} J_{ki}J_{kj}}_{\gamma}\ \omega_j \right) \tag{17}$$

式 (17) において下線部 α,β,γ は $\boldsymbol{\omega}$ に依存せず計算できるため Gauss-Seidel 法の繰り返し計算を行う前にあらかじめ計算しておくことができる．

4.2 ブロック構造の係数行列を持つ連立方程式の Gauss-Seidel 法

3 節で述べたように，実際の J は単一の巨大な行列ではなく，関節・制御点の数に従ってブロック構造をなす．

IK で用いる制御点の集合を $\boldsymbol{Cp} := \{c_0, c_1, \dots\}$，関節の集合を $\boldsymbol{Nd} := \{n_0, n_1, \dots\}$ と定義する．また，関節 n_x と制御点 c_z の間のヤコビアンを $\left[J^{n_x}_{c_z}\right]$ と表記する．

これらを用いると，前節で述べた事前計算値 α,β,γ を，各関節ごとに式 (18)〜(20) のように計算できる．ただし $1 \leq i \leq NDOF(n_x)$，$1 \leq j \leq NDOF(n_y)$ である．

$$[\alpha_{n_x}]_i = \sum_{c_z \in \boldsymbol{Cp}} \left\{ \sum_{k=0}^{NDOF(c_z)} \left(\left[J_{c_z}^{n_x} \right]_{ki} \right)^2 \right\} \tag{18}$$

$$[\beta_{n_x}]_i = \sum_{c_z \in \boldsymbol{Cp}} \left\{ \sum_{k=0}^{NDOF(c_z)} \left(\left[J_{c_z}^{n_x} \right]_{ki} \left[v_{c_z} \right]_k \right) \right\} \tag{19}$$

$$\left[\gamma_{n_x}^{n_y}\right]_{ij} = \sum_{c_z \in \boldsymbol{Cp}} \left\{ \sum_{k=0}^{NDOF(c_z)} \left(\left[J_{c_z}^{n_x} \right]_{ki} \left[J_{c_z}^{n_y} \right]_{kj} \right) \right\} \tag{20}$$

これらの値を用い，繰り返し計算を進行する．

$$[\delta_{n_x}]_i = \sum_{n_y \in \boldsymbol{Nd}} \left\{ \sum_{k=0}^{NDOF(n_y)} \left(\left[\gamma_{n_x}^{n_y} \right]_{ik} \left[\omega_{ny} \right]_k \right) \right\} \tag{21}$$

$$[\epsilon_{n_x}]_i = \sum_{k=0(k\neq i)}^{NDOF(n_x)} \left(\left[\gamma_{n_x}^{n_x} \right]_{ik} [\omega_{nx}]_k \right) \tag{22}$$

$\boldsymbol{\omega}$ の更新式は式 (23) のようになる．

$$[\omega_{n_x}]_i = \frac{1}{[\alpha_{nx}]_i} \left\{ [\beta_{nx}]_i - ([\delta_{nx}]_i + [\epsilon_{nx}]_i) \right\} \tag{23}$$

ただしここで，ヤコビアン $\left[J_{c_z}^{n_x}\right]$ について，関節 n_x を動かしても制御点 c_z を動かす事ができない場合 $\left[J_{c_z}^{n_x}\right] = \boldsymbol{O}$ である．ゆえに，α,β,γ を計算する上で，$c_z \in \boldsymbol{Cp}$ のうち関節 n_x によって動かす事のできない制御点については計算する必要が無い．そこで，関節 n_x によって動かす事のできる制御点の集合を $\boldsymbol{Cp_{n_x}}$ と定義し，実際の $[\alpha_{n_x}], [\beta_{n_x}], \left[\gamma_{n_x}^{n_y}\right]$ の計算には $\boldsymbol{Cp}$ のかわりに $\boldsymbol{Cp_{n_x}}$ を用いる．

同様に，γ を計算する上で，$\left[J_{c_z}^{n_x}\right]_{ki} \left[J_{c_z}^{n_x}\right]_{kj}$ が常に 0 となるような n_x, n_y の組については計算する必要が無い．$\left[J_{c_z}^{n_x}\right]_{ki} \left[J_{c_z}^{n_x}\right]_{kj}$ が常に 0 となるのは，関節 n_x, n_y の両方によって動かすことのできる制御点が一つもない場合である (逆に，そのような制御点が一つでも存在すれば $\left[J_{c_z}^{n_x}\right]_{ki} \left[J_{c_z}^{n_x}\right]_{kj}$ は常に 0 とはならない)．そこで，ある関節 n_x について以下のように集合 $\boldsymbol{Nd_{n_x}}$ を定義し，$\left[\gamma_{n_x}^{n_y}\right]$ の計算には $\boldsymbol{Nd}$ の代わりに $\boldsymbol{Nd_{n_x}}$ を用いる．

$$\boldsymbol{Nd_{n_x}} := \left\{ n_y \mid \exists c_z : c_z \in \boldsymbol{Cp_{n_x}} \text{かつ } c_z \in \boldsymbol{Cp_{n_y}} \right\}$$

なお，Springhead2 における実装では，関節 n_x が集合 $\boldsymbol{Cp_{n_x}}$ と $\boldsymbol{Nd_{n_x}}$ を保持するほか，制御点 c_y も，自身を動かすために使用できる関節の集合 $\boldsymbol{Nd_{c_y}}$ を保持している．関節 n_x を動かすために制御点 c_y を使用したい場合，ユーザは関節 n_x に制御点 c_y を「登録」する．その際に，各集合が更新される．

5　関節角の更新

Springhead2 において，関節の動作は関節に組み込まれたバネダンパの基準角度を変更する事によって行われる．その際にいくつか処理を行うことで，通常の IK には無い機能を実現している．

5.1　基準位置からの変位を少なくする IK

各関節には IK によって基準角度を変化させる前に「本来の基準角度」がある．そこで，IK によって求めた関節角を適用する際に,「本来の基準角度へ向けようとするバネダンパ」と「IK で求めた角度へ向けようとするバネダンパ」の両者が関節を引き合う構造とすることで，本来の基準角度からの変位をなるべく少なく保ちつつ目標位置を達成する IK を実現できる．

本来関節が持つ基準角度を θ_{orig}，バネダンパを p_{orig}, d_{orig} とし，IK で求めた角度を θ_{ik}，そのためのバネダンパを p_{ik}, d_{ik} とすると，IK 適用後の関節角度・バネダンパ値は以下のようになる．

$$\theta_{new} = \frac{p_{orig}\theta_{orig} + p_{ik}\theta_{ik}}{p_{orig} + p_{ik}} \tag{24}$$

$$p_{new} = p_{orig} + p_{ik} \tag{25}$$

$$d_{new} = d_{orig} + d_{ik} \tag{26}$$

注意点として，p_{ik} に対する p_{orig} の値が大きすぎると，IK を作動させても制御点が目標位置に到達しなくなる．

人型モデルにおいて人間の標準姿勢を「本来の基準角度」としておくことで不自然な姿勢をとりづらくする，といった事に効果がある．

5.2　制御点における力の発生

IK のために使用したヤコビアンの転置を用いると，制御点において特定の力を発生するために必要な関節トルクを計算できる．この関節トルクを発生するよう，バネダンパの基準位置を IK で求めたものからずらす事によって，制御点において力を発生することができる．

この機能は 2009 年 6 月 26 日時点で実装が完了していないため，完了次第ドキュメント化する．