

ヒューマノイドロボット勉強会

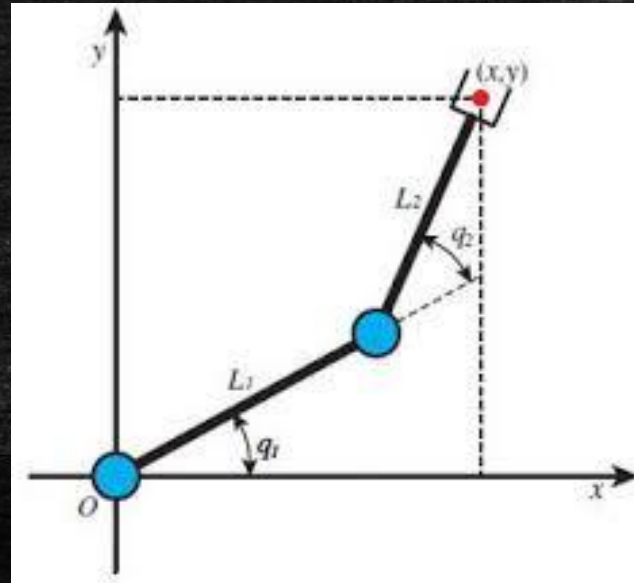
順運動学

前回

- 三角関数の知識を使って2リンクの順運動学を計算するプログラムを作った
- エクセル上で作ったものとCで作ったものを比較してあっているかを検証した

順運動学

- 各関節の角度が分かっている場合にエンドエフェクタ(手先)の位置・姿勢を計算する
- 逆運動学計算時に必要



計算方法

- リンク数の極端に少ないものは三角関数の知識で解析的に計算が可
- 逆運動学を計算するときは順運動学で得られた式の逆関数を計算すればよい



リンク数が増えらるとつらい(3軸以上だと式が複雑化)

逆運動学を求めるときは一般的に解析では解が一意にもとまらない

今回

- 位置と姿勢を考慮するために同次変換行列とチェーンルールから全リンクの位置・姿勢を計算する
- リンクが増えても前後のリンクの位置・姿勢，親子関係の情報を記述すれば簡単に全関節の位置・姿勢の計算が可能

同次変換行列

- 位置と姿勢を表す 4×4 の行列
- 位置を表すベクトル p , 姿勢を表す回転行列 R からなる
- 各関節はローカル座標で位置・姿勢の座標を持つ

$$T = \begin{pmatrix} R & p \\ 0 & 1 \end{pmatrix}$$

$$R = R_x(\varphi)R_y(\theta)R_z(\gamma) = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix}$$

$$p = (x \quad y \quad z)$$

チェーンルール

- 順運動学ではワールド座標で手先の位置を知りたい
- 各関節の位置・姿勢情報は親リンクに基づく記述
- ワールド座標に変換したい場合は原点から各リンクの同次変換行列を目標リンクまで掛け合わせることで原点における手先の絶対位置がわかる

$$T_N = T_1^1 T_2^2 T_3^4 \dots T_N$$

順運動学

- 各リンクの親リンクとの相対位置を予め与えておくことで、同次変換行列とチェーンルールにより手先リンクの位置・姿勢を計算することが可能になる

実際にプログラムを試してみる

- ソースコードの雛形は
HumanoidSystem/HumanoidRobotLibrary/Kinematisを参照
- 今回も前回と同じく、2リンクマニピュレータの順運動学を計算する