

ロボット工学プログラミング課題の工夫点

学籍番号：20T1126N 氏名：YANG GUANGZE

まずは、プログラムを理解するために、サンプルプログラムを見ながら「GLUT による OpenGL 入門」を 6 時間に全部学習した。まとめてサンプルプログラムを以下のように解釈する。

- glshapes.c:- ロボットアーム (描画) - 軌道 (変動しなく)
- invK3d_l:- 順運動学 (座標) - ヤコビ行列- 逆運動学- 行列計算 library
- invK3d.c:- 初期設定- データ入力: 目標座標- main 文
- glwindows.c:- keyboard 機能- windows 設定
- armIK.dat- リンクの長さや角度
- glshapes.h- リンクの半径

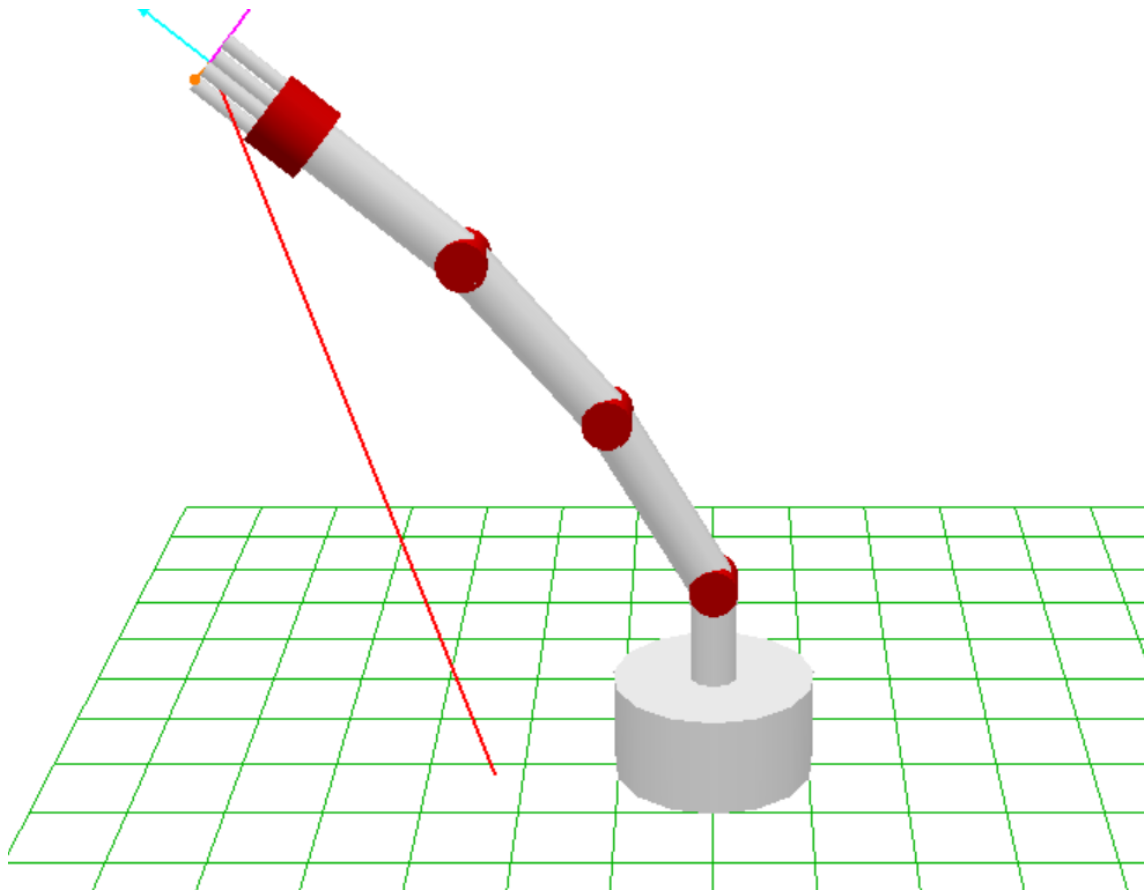
これらのプログラムを理解した上で、ロボットを作ってきた。3 次元 3 軸を作るために、プログラム内の各リンクとジョイント用の変数を加えたり、for 文の条件にあるジョイント「J2」とリンク「L2」などを「Jnum」と「Lnum」を書き換えたりした。また、ほかの初期設定をした。

次に 3 軸を作る時に、先生の 3 軸 3 軸マニピュレータの追従に従い、サンプルプログラムを修正した。アームの動きが軌道と逆になった。順運動学とヤコビ行列を何回チェックし、間違っていないが、動きが逆である。いろいろな場合を考え、理由を分かった。この理由は、描画する時、ジョイントの回転によって、角度の方向が逆になった。そして、描画する時に、 θ_2 と θ_3 の角度をマイナスに変えたら、アームは正しく動くようになった。

そして、3 次元 4 軸マニピュレータの順運動学とヤコビ行列を求め、疑似行列を用いる。3 次元 4 軸マニピュレータができた。

また、描画について、以下の図に示すような色と形状でロボットアームを作ってきた。Keyboard 機能については、ノートパソコンが「Home」key がないので、実行 key を「F1」を変えた。なお、「カメラ移動」の左右は、「A」「D」key に変換した。「カメラ移動」の上下は、「Q」「E」key に変換した。「カメラ移動」の前と後は、「W」「S」key に変換した。「カメラ角度」の上下左右は、「↑」「↓」「←」「→」key に変換した。

以上、ロボット工学のプログラミング課題を完成した。



3次元4軸マニピュレータについての計算：

3次元4軸順運動学：

$$\begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} = \begin{bmatrix} p_{3x} + l_4 \cdot C_{234} \cdot C_1 \\ p_{3y} + l_4 \cdot C_{234} \cdot S_1 \\ p_{3z} + l_4 \cdot S_{234} \end{bmatrix}$$

$$= \begin{bmatrix} (l_2 C_2 + l_3 C_{23} + l_4 C_{234}) C_1 \\ (l_2 C_2 + l_3 C_{23} + l_4 C_{234}) S_1 \\ l_0 + l_1 + l_2 S_2 + l_3 S_{23} + l_4 S_{234} \end{bmatrix}$$

3次元4軸ヤコビ行列：

$$\dot{\mathbf{p}} = \mathbf{J} \dot{\boldsymbol{\theta}}$$

よって、

$$\begin{bmatrix} \dot{p}_x \\ \dot{p}_y \\ \dot{p}_z \end{bmatrix} = \begin{bmatrix} -(l_2 C_2 + l_3 C_{23} + l_4 C_{234}) S_1 & -(l_2 C_2 + l_3 C_{23} + l_4 C_{234}) C_1 & -(l_3 C_{23} + l_4 C_{234}) C_1 & -l_4 C_{234} C_1 \\ (l_2 C_2 + l_3 C_{23} + l_4 C_{234}) C_1 & -(l_2 C_2 + l_3 C_{23} + l_4 C_{234}) S_1 & -(l_3 C_{23} + l_4 C_{234}) S_1 & -l_4 C_{234} S_1 \\ 0 & l_2 C_2 + l_3 C_{23} + l_4 C_{234} & l_3 C_{23} + l_4 C_{234} & l_4 C_{234} \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \\ \dot{\theta}_4 \end{bmatrix}$$

ただし、プログラムでは、ハンドの長さ l_5 を加えると、 $l_4 = l_5 + l_4$ となる。