document

Actat

1　ロボット製作の動機

このロボットの製作に取り組むのは，歩行ロボット製作を経験するためである．最終的には二足歩行ロボットの製作を目指しているが，6 自由度の脚の設計は難易度が高いと判断し，今回は 3 自由度の脚を 4 つ備えた四足歩行ロボットを製作することにした．

このロボットの製作では以下の内容を経験することを期待している．

- 自宅でのロボット製作
- サーボモータを用いた機構の設計
- 最低限の電気回路の取扱い
- ROS を用いたプログラミング
- 歩行ロボットの制御

2　機械設計・製作

設計・製作したロボットを図 1 に示す．脚は前後左右に鏡像になっている同一の機構とした．サーボモータは研究室で使用しているモータと合わせて近藤科学株式会社の B3M-SC-1170-A を用いた．CIT Brains が公開している B3M モータを用いた二足歩行ロボットを参考に，関節間の距離は 100 mm にした．

胴体中央下部に株式会社アールティの USB 出力 9 軸 IMU センサモジュールを搭載した．センサの位置をロボットのベースリンクと一致させて計算を簡単にする意図がある．胴体上部前方に取り付けられた板にサーボモータと PC の間の通信のための基板を固定した．

3　電装

ロボット全体で 12 個のモータが使われている．モータとパソコンは RS485USB/シリアル変換アダプターを介して通信する．直列に接続できるモータの数は限られているので XH コネクター用ハブ typeA によって各脚ごとの系統に分けて接続した．配線の様子を図 2 に示す．

XH コネクター用ハブには 5 つのコネクタがあり，すべて並列に接続される．このハブはロ

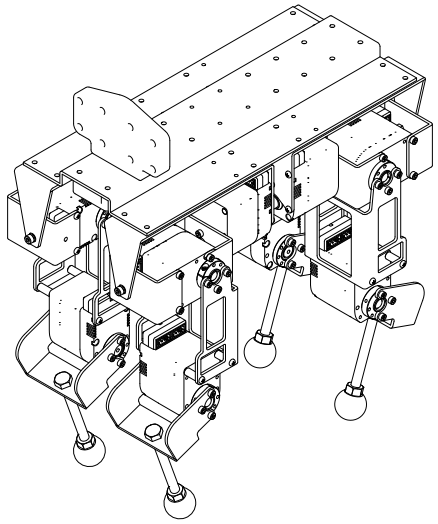

図 1　設計・製作したロボット

ボット前方に横向きに取り付けられており，中央のコネクタを RS485USB/シリアル変換アダプターに接続し，左右のコネクタをそれぞれ脚のモータと接続した．脚の 3 つのモータは Hip flexion/extension, Hip ab/adduction, knee の順に接続された．Hip のモータは機械的な接続とは逆の順序である．

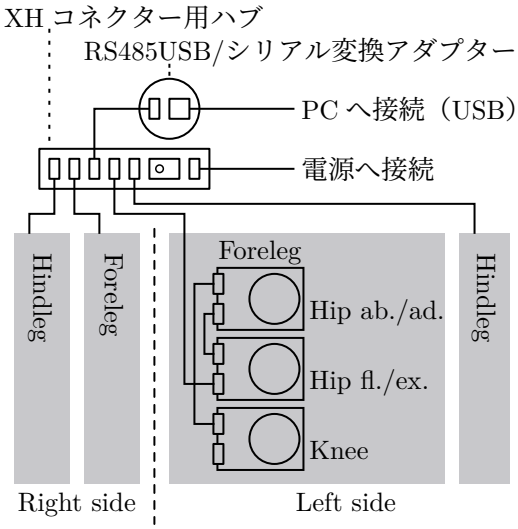

図 2　配線の様子

サーボモータは配線に先立って ID の書き込みを行った．書き込んだ ID は表 1 に示すとおりである．

表 1　ID と関節の対応

id	脚	関節
0	左前	Hip ab/adduction
1	左前	Hip flexion/extension
2	左前	knee
3	右前	Hip ab/adduction
4	右前	Hip flexion/extension
5	右前	knee
6	左後	Hip ab/adduction
7	左後	Hip flexion/extension
8	左後	knee
9	右後	Hip ab/adduction
10	右後	Hip flexion/extension
11	右後	knee

4　Package の作成

別のディレクトリで`$ ros2 pkg create --build-type ament_cmake --node-name quadruped_takahashi quadruped_takahashi` を実行して package.xml, CMakeLists.txt, quadruped_takahashi.cpp を作成した．その後，作成したファイルをこのリポジトリ内へ移動した．

5　モータとの通信

モータとの通信には Actat/kondo_b3m_ros2 を用いる．remake ブランチ[1)]に合わせて launch ファイルを用意し，通信ができることを確認する．以下に launch ファイルのうち，kondo_b3m_ros2 のノードの設定に関する部分を抜粋する．軸の方向を反転しているのは，後述する座標系に合わせるためである．

```
kondo_b3m_ros2_node = Node(
  package='kondo_b3m_ros2',
  executable='kondo_b3m',
  remappings=[('b3m_joint_state', 'joint_states')],
  parameters=[{'motor_list': [
    "{'id': 0, 'name': 'lf0', 'direction': False}",
    "{'id': 1, 'name': 'lf1', 'direction': False}",
```

1)　このプロジェクトでうまく動作することを確認したら master にする予定である．

```
        "{'id': 2, 'name': 'lf2', 'direction': False}",
        "{'id': 3, 'name': 'rf0', 'direction': False}",
        "{'id': 4, 'name': 'rf1'}",
        "{'id': 5, 'name': 'rf2'}",
        "{'id': 6, 'name': 'lh0'}",
        "{'id': 7, 'name': 'lh1', 'direction': False}",
        "{'id': 8, 'name': 'lh2', 'direction': False}",
        "{'id': 9, 'name': 'rh0'}",
        "{'id': 10, 'name': 'rh1'}",
        "{'id': 11, 'name': 'rh2'}"
      ]}],
    )
```

あるウインドウで`$ ros2 launch quadruped_takahashi launch.py` を実行して起動し，別のウインドウで`$ ros2 topic echo /joint_states` することで各関節の角度・角速度が出力されていることを確認した．

6 URDF ファイルの作成

URDF ファイルの作成にあたってロボットの関節に関する座標系を設定する．初めに左前脚を考える（図 3）．2 つの Hip の関節は軸が交わるように作られており，この交点を脚の付け根とする．脚の付け根を原点としてロボットの前方に x 軸，左方向に y 軸，鉛直上方向に z 軸をとった座標系を Σ_{lf0} とする．Σ_{lf0} の x 軸は Hip ab/adduction の関節軸と一致しており，この軸周りに θ_{lf0} 回転した座標系を Σ_{lf1} とする．θ_{lf0} は x 軸の方向を正とする．Σ_{lf1} の y 軸は Hip flextion/extension の関節軸と一致しており，この軸周りに Σ_{lf1} 回転した座標系を Σ_{lf2} とする．Σ_{lf1} は y 軸の方向を正とする．この座標系 Σ_{lf2} は腿のリンクと対応しており，腿は $-z$ の方向にのびている．膝関節は Σ_{lf2} において $[0, 0, -0.1]\,\mathrm{m}$ の位置に存在する．この位置を原点とし，y 軸周りに θ_{lf2} 回転した座標系を Σ_{lf3} とする．この座標系は脛のリンクと対応しており，$[0, 0, -0.1]\,\mathrm{m}$ の位置が足の球の中心になる．

他の脚にも同様に 3 つの関節角と 4 つの座標系を定義する．すべて座標軸は直立したときに，x 軸が前方，y 軸は左方を向くように設定する．関節角の向きは x, y 軸の正方向によって定める．

ロボットの中心となる base_link の位置は，4 つの hip joint の中心に定める．座標系の向きは x 軸が前方，y 軸は左方，z 軸が鉛直上方である．

URDF ファイルを作成して，CMakeLists.txt に追記した．rviz の config ファイルも同様に CMakeLists.txt に追記した．launch.py を編集し，rviz を起動して/joint_states の関節角をモデルの表示に反映させる．base_link を固定して表示させることができた（図 4）．

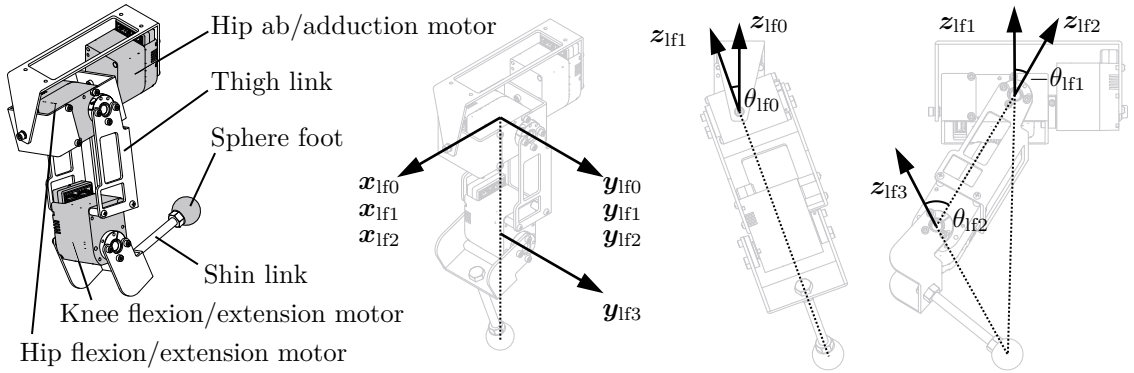

図 3　左前脚の座標系の設定．脚は 3 つの関節軸を持っておりそれぞれが独立に回転する．脚を垂直に伸ばした状態で 2 つの尻関節の交点と足の球面の中心を結ぶ線分上に各座標系の原点を設定する．それぞれの関節の回転角を $\theta_{\mathrm{lf0}}, \theta_{\mathrm{lf1}}, \theta_{\mathrm{lf2}}$ とする．

7　IMU のデータを読み取る

胴体中央下部に株式会社アールティの USB 出力 9 軸 IMU センサモジュールが搭載されている．rt-net/rt_usb_9axisimu_driver によって ros の topic（/imu/data_raw と/imu/mag）にデータをパブリッシュする．パブリッシュされたデータは CCNYRoboticsLab/imu_tools の imu_complementary_filter で姿勢の情報にする．使用するパッケージを package.xml に追記し，launch ファイルにノードを起動する記述を追加した．

`odom` から `base_link` への `tf` をパブリッシュする `quadruped_takahashi_odometry_node` というノードを作る．`/imu/data` をサブスクライブし，更新が入ったらコールバックで足の情報を参照して `tf` に情報を出す．ひとまず xy 平面内での位置はゼロに固定し，高さは最も低い位置にある足が接地しているものとする．

8　脚の逆運動学

`base_link` における足の位置を指令して各モータを追従させるために逆運動学計算を実装する．初めに，各関節がとれる値の範囲を表 2 のとおりに定める．

初めに順運動学を考え，`base_link` からみた各足の位置を表すベクトル $\boldsymbol{r}_{\mathrm{xx4}}^{\mathrm{base}}$ を関節角 $\theta_{\mathrm{xx0}}, \theta_{\mathrm{xx1}}, \theta_{\mathrm{xx2}}$ を用いて表す．ここでは xx は左前脚・右前脚・左後脚・右後脚に合わせて lf, rf, lh, rh が入るものとする．`base_link` からみた脚の付け根の位置を表すベクトルを $\boldsymbol{r}_{\mathrm{xx0}}^{\mathrm{base}}$，腿と脛の長さを $l_{\mathrm{t}}, l_{\mathrm{s}}$ とすると以下のようになる．

$$\boldsymbol{r}_{\mathrm{xx4}}^{\mathrm{base}} = \boldsymbol{r}_{\mathrm{xx0}}^{\mathrm{base}} + \begin{bmatrix} -l_{\mathrm{t}} \sin\theta_{\mathrm{xx1}} - l_{\mathrm{s}} \sin\left(\theta_{\mathrm{xx1}} + \theta_{\mathrm{xx2}}\right) \\ \sin\theta_{\mathrm{xx0}} \left(l_{\mathrm{t}} \cos\theta_{\mathrm{xx1}} + l_{\mathrm{s}} \cos\left(\theta_{\mathrm{xx1}} + \theta_{\mathrm{xx2}}\right)\right) \\ -\cos\theta_{\mathrm{xx0}} \left(l_{\mathrm{t}} \cos\theta_{\mathrm{xx1}} + l_{\mathrm{s}} \cos\left(\theta_{\mathrm{xx1}} + \theta_{\mathrm{xx2}}\right)\right) \end{bmatrix}$$

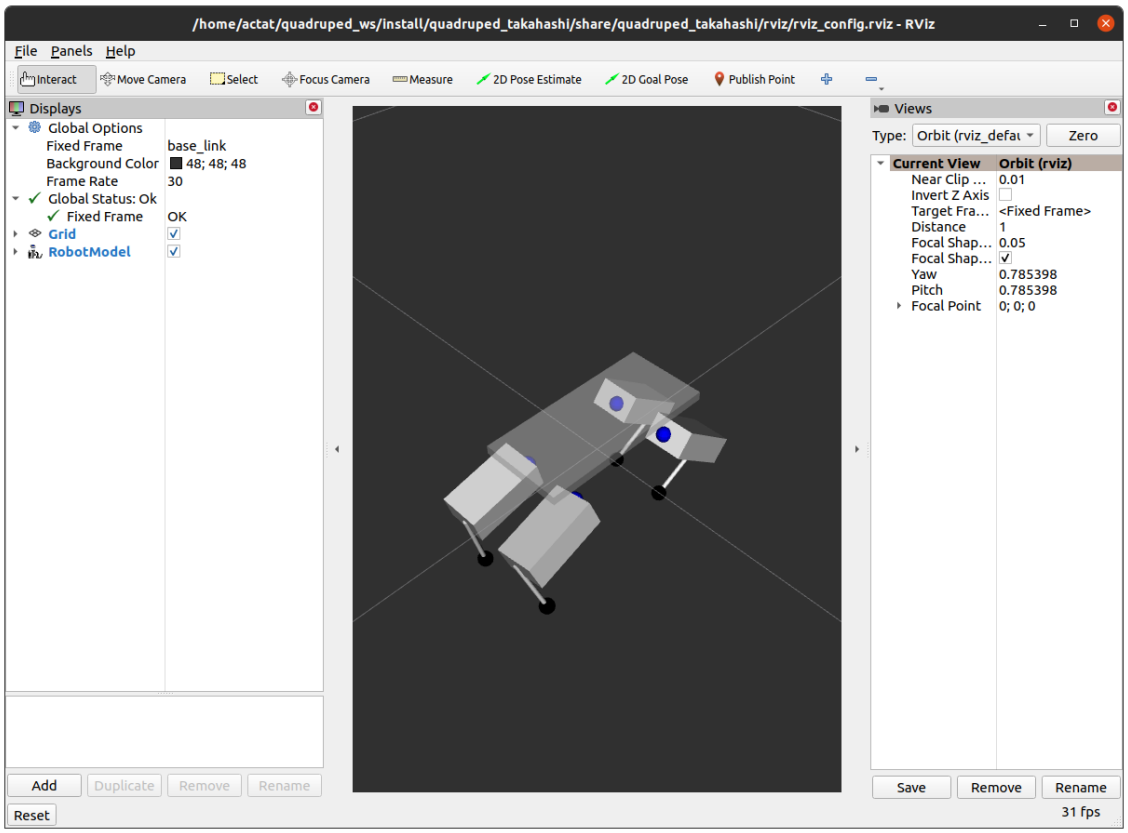

図 4　URDF ファイルを読み込んで表示する rviz の画面

ここからは逆運動学を考える．脚の付け根の位置ベクトル $\boldsymbol{r}_{\mathrm{xx0}}^{\mathrm{base}}$ は定数なので最初にこれを引いて $\boldsymbol{r}_{\mathrm{xx4}}^{\mathrm{base}} - \boldsymbol{r}_{\mathrm{xx0}}^{\mathrm{base}} = [x\, y\, z]^T$ とする．

初めに y, z のみで定まる θ_{xx0} を求める．$z \neq 0$ の場合は以下の通り，arctan で求められる．

$$
\begin{cases}
y = \sin\theta_{\mathrm{xx0}} \left(l_{\mathrm{t}} \cos\theta_{\mathrm{xx1}} + l_{\mathrm{s}} \cos\left(\theta_{\mathrm{xx1}} + \theta_{\mathrm{xx2}}\right)\right) \\
z = -\cos\theta_{\mathrm{xx0}} \left(l_{\mathrm{t}} \cos\theta_{\mathrm{xx1}} + l_{\mathrm{s}} \cos\left(\theta_{\mathrm{xx1}} + \theta_{\mathrm{xx2}}\right)\right)
\end{cases}
$$

$$
-\frac{y}{z} = \tan\theta_{\mathrm{xx0}}
$$

$$
\theta_{\mathrm{xx0}} = \arctan\left(-\frac{y}{z}\right)
$$

$z = 0$ の場合，図 5 に示すように θ_{xx0} は任意の値をとれる．便宜上，この場合は $\theta_{\mathrm{xx0}} = 0$ に定めることにする．

$$
\theta_{\mathrm{xx0}} = \begin{cases}
0 & z = 0\text{のとき} \\
\arctan\left(-\frac{y}{z}\right) & z \neq 0\text{のとき}
\end{cases}
$$

表 2　関節角の範囲

id	脚	関節	最小値		最大値	
			deg	rad	deg	rad
0	左前	Hip ab/adduction	-10	-0.174532925	45	0.785398163
1	左前	Hip flexion/extension	-90	-1.570796326	0	0
2	左前	knee	0	0	160.53	2.801777048
3	右前	Hip ab/adduction	-45	-0.785398163	10	0.174532925
4	右前	Hip flexion/extension	-90	-1.570796326	0	0
5	右前	knee	0	0	160.53	2.801777048
6	左後	Hip ab/adduction	-10	-0.174532925	45	0.785398163
7	左後	Hip flexion/extension	0	0	90	1.570796326
8	左後	knee	-160.53	-2.801777048	0	0
9	右後	Hip ab/adduction	-45	-0.785398163	10	0.174532925
10	右後	Hip flexion/extension	0	0	90	1.570796326
11	右後	knee	-160.53	-2.801777048	0	0

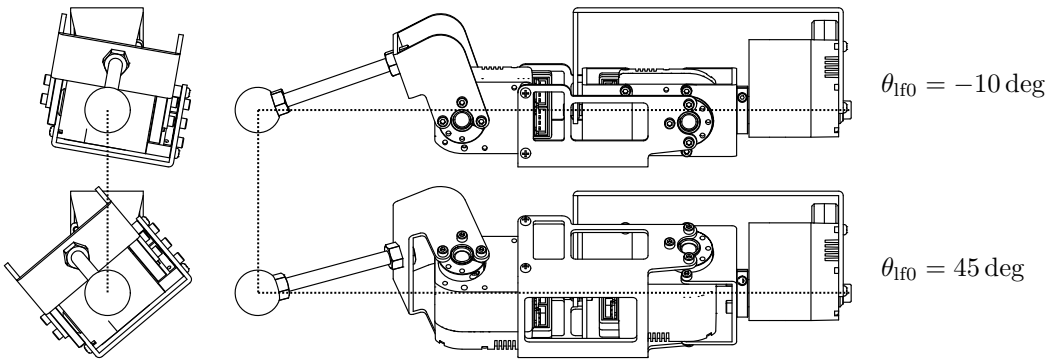

図 5　$z = 0$ の場合，θ_{xx0} は任意の値をとれる．

次に脚の長さに注目して膝関節の角度 θ_{xx2} を求める．

$$
\begin{aligned}
x^2 + y^2 + z^2 &= \left| \boldsymbol{r}_{\mathrm{xx4}}^{\mathrm{base}} - \boldsymbol{r}_{\mathrm{xx0}}^{\mathrm{base}} \right|^2 \\
&= \left(-l_{\mathrm{t}} \sin\theta_{\mathrm{xx1}} - l_{\mathrm{s}} \sin\left(\theta_{\mathrm{xx1}} + \theta_{\mathrm{xx2}}\right)\right)^2 \\
&\quad + \left(\sin\theta_{\mathrm{xx0}} \left(l_{\mathrm{t}} \cos\theta_{\mathrm{xx1}} + l_{\mathrm{s}} \cos\left(\theta_{\mathrm{xx1}} + \theta_{\mathrm{xx2}}\right)\right)\right)^2 \\
&\quad + \left(-\cos\theta_{\mathrm{xx0}} \left(l_{\mathrm{t}} \cos\theta_{\mathrm{xx1}} + l_{\mathrm{s}} \cos\left(\theta_{\mathrm{xx1}} + \theta_{\mathrm{xx2}}\right)\right)\right)^2 \\
&= \left(-l_{\mathrm{t}} \sin\theta_{\mathrm{xx1}} - l_{\mathrm{s}} \sin\left(\theta_{\mathrm{xx1}} + \theta_{\mathrm{xx2}}\right)\right)^2 \\
&\quad + \left(l_{\mathrm{t}} \cos\theta_{\mathrm{xx1}} + l_{\mathrm{s}} \cos\left(\theta_{\mathrm{xx1}} + \theta_{\mathrm{xx2}}\right)\right)^2 \\
&= l_{\mathrm{t}}^2 + 2 l_{\mathrm{t}} l_{\mathrm{s}} \left(\sin\theta_{\mathrm{xx1}} \sin\left(\theta_{\mathrm{xx1}} + \theta_{\mathrm{xx2}}\right) + \cos\theta_{\mathrm{xx1}} \cos\left(\theta_{\mathrm{xx1}} + \theta_{\mathrm{xx2}}\right)\right) + l_{\mathrm{s}}^2 \\
&= l_{\mathrm{t}}^2 + 2 l_{\mathrm{t}} l_{\mathrm{s}} \cos\theta_{\mathrm{xx2}} + l_{\mathrm{s}}^2 \\
\cos\theta_{\mathrm{xx2}} &= \frac{x^2 + y^2 + z^2 - l_{\mathrm{t}}^2 - l_{\mathrm{s}}^2}{2 l_{\mathrm{t}} l_{\mathrm{s}}}
\end{aligned}
$$

膝関節の値域は前脚の場合 $0 \leq \theta_{\mathrm{xf2}} \leq 2.801777048$，後脚の場合 $-2.801777048 \leq \theta_{\mathrm{xh2}} \leq 0$ であることに注意する．

$$\theta_{\mathrm{xx2}} = \begin{cases} \arccos \frac{x^2+y^2+z^2-l_{\mathrm{t}}^2-l_{\mathrm{s}}^2}{2l_{\mathrm{t}}l_{\mathrm{s}}} & \text{前脚の場合} \\ -\arccos \frac{x^2+y^2+z^2-l_{\mathrm{t}}^2-l_{\mathrm{s}}^2}{2l_{\mathrm{t}}l_{\mathrm{s}}} & \text{後脚の場合} \end{cases}$$

最後に θ_{xx1} を求める．x と y^2+z^2 の式変形により，$\sin\theta_{\mathrm{xx1}}, \cos\theta_{\mathrm{xx1}}$ の関係式を得る．

$$\begin{aligned} x &= -l_{\mathrm{t}}\sin\theta_{\mathrm{xx1}} - l_{\mathrm{s}}\sin\left(\theta_{\mathrm{xx1}}+\theta_{\mathrm{xx2}}\right) \\ &= -l_{\mathrm{t}}\sin\theta_{\mathrm{xx1}} - l_{\mathrm{s}}\left(\sin\theta_{\mathrm{xx1}}\cos\theta_{\mathrm{xx2}} + \cos\theta_{\mathrm{xx1}}\sin\theta_{\mathrm{xx2}}\right) \\ &= \left(-l_{\mathrm{s}}\sin\theta_{\mathrm{xx2}}\right)\cos\theta_{\mathrm{xx1}} + \left(-l_{\mathrm{t}} - l_{\mathrm{s}}\cos\theta_{\mathrm{xx2}}\right)\sin\theta_{\mathrm{xx1}} \end{aligned}$$

$$\begin{aligned} y^2+z^2 &= \left(l_{\mathrm{t}}\cos\theta_{\mathrm{xx1}} + l_{\mathrm{s}}\cos\left(\theta_{\mathrm{xx1}}+\theta_{\mathrm{xx2}}\right)\right)^2 \\ &= \left(l_{\mathrm{t}}\cos\theta_{\mathrm{xx1}} + l_{\mathrm{s}}\left(\cos\theta_{\mathrm{xx1}}\cos\theta_{\mathrm{xx2}} - \sin\theta_{\mathrm{xx1}}\sin\theta_{\mathrm{xx2}}\right)\right)^2 \\ &= \left(\left(l_{\mathrm{t}} + l_{\mathrm{s}}\cos\theta_{\mathrm{xx2}}\right)\cos\theta_{\mathrm{xx1}} + \left(-l_{\mathrm{s}}\sin\theta_{\mathrm{xx2}}\right)\sin\theta_{\mathrm{xx1}}\right)^2 \end{aligned}$$

値の範囲を確認する．前脚の場合 $-1.570796326 \leq \theta_{\mathrm{xf1}} \leq 0, 0 \leq \theta_{\mathrm{xf2}} \leq 2.801777048$ なので，$0 \leq \cos\theta_{\mathrm{xf1}} \leq 1, -1 \leq \sin\theta_{\mathrm{xf1}} \leq 0, -1 < \cos\theta_{\mathrm{xf2}} \leq 1, 0 \leq \sin\theta_{\mathrm{xf2}} \leq 1$ である．今，$l_{\mathrm{t}} = l_{\mathrm{s}}$ なので $\left(l_{\mathrm{t}} + l_{\mathrm{s}}\cos\theta_{\mathrm{xf2}}\right)\cos\theta_{\mathrm{xf1}} \geq 0, \left(-l_{\mathrm{s}}\sin\theta_{\mathrm{xf2}}\right)\sin\theta_{\mathrm{xf1}} \geq 0$ だから

$$\sqrt{y^2+z^2} = \left(l_{\mathrm{t}} + l_{\mathrm{s}}\cos\theta_{\mathrm{xf2}}\right)\cos\theta_{\mathrm{xf1}} + \left(-l_{\mathrm{s}}\sin\theta_{\mathrm{xf2}}\right)\sin\theta_{\mathrm{xf1}}$$

である．後脚の場合 $0 \leq \theta_{\mathrm{xh1}} \leq 1.570796326, -2.801777048 \leq \theta_{\mathrm{xh2}} \leq 0$ なので，$0 \leq \cos\theta_{\mathrm{xh1}} \leq 1, 0 \leq \sin\theta_{\mathrm{xh1}} \leq 1, -1 < \cos\theta_{\mathrm{xh2}} \leq 1, -1 \leq \sin\theta_{\mathrm{xh2}} \leq 0$ である．$l_{\mathrm{t}} = l_{\mathrm{s}}$ だから $\left(l_{\mathrm{t}} + l_{\mathrm{s}}\cos\theta_{\mathrm{xh2}}\right)\cos\theta_{\mathrm{xh1}} \geq 0, \left(-l_{\mathrm{s}}\sin\theta_{\mathrm{xh2}}\right)\sin\theta_{\mathrm{xh1}} \geq 0$ なので

$$\sqrt{y^2+z^2} = \left(l_{\mathrm{t}} + l_{\mathrm{s}}\cos\theta_{\mathrm{xh2}}\right)\cos\theta_{\mathrm{xh1}} + \left(-l_{\mathrm{s}}\sin\theta_{\mathrm{xh2}}\right)\sin\theta_{\mathrm{xh1}}$$

である．結局前脚でも後脚でも

$$\sqrt{y^2+z^2} = \left(l_{\mathrm{t}} + l_{\mathrm{s}}\cos\theta_{\mathrm{xx2}}\right)\cos\theta_{\mathrm{xx1}} + \left(-l_{\mathrm{s}}\sin\theta_{\mathrm{xx2}}\right)\sin\theta_{\mathrm{xx1}}$$

である．これで $\sin\theta_{\mathrm{xx1}}, \cos\theta_{\mathrm{xx1}}$ の 1 次式を 2 つ得られたので，連立して値を求められる．

$$\begin{bmatrix} x \\ \sqrt{y^2+z^2} \end{bmatrix} = \begin{bmatrix} -l_{\mathrm{s}}\sin\theta_{\mathrm{xx2}} & -l_{\mathrm{t}} - l_{\mathrm{s}}\cos\theta_{\mathrm{xx2}} \\ l_{\mathrm{t}} + l_{\mathrm{s}}\cos\theta_{\mathrm{xx2}} & -l_{\mathrm{s}}\sin\theta_{\mathrm{xx2}} \end{bmatrix} \begin{bmatrix} \cos\theta_{\mathrm{xx1}} \\ \sin\theta_{\mathrm{xx1}} \end{bmatrix}$$

$$\begin{bmatrix} \cos\theta_{\mathrm{xx1}} \\ \sin\theta_{\mathrm{xx1}} \end{bmatrix} = \frac{1}{l_{\mathrm{t}}^2 + 2l_{\mathrm{t}}l_{\mathrm{s}}\cos\theta_{\mathrm{xx2}} + l_{\mathrm{s}}^2} \begin{bmatrix} -l_{\mathrm{s}}\sin\theta_{\mathrm{xx2}} & l_{\mathrm{t}} + l_{\mathrm{s}}\cos\theta_{\mathrm{xx2}} \\ -l_{\mathrm{t}} - l_{\mathrm{s}}\cos\theta_{\mathrm{xx2}} & -l_{\mathrm{s}}\sin\theta_{\mathrm{xx2}} \end{bmatrix} \begin{bmatrix} x \\ \sqrt{y^2+z^2} \end{bmatrix}$$

$$\begin{aligned} \tan\theta_{\mathrm{xx1}} &= \frac{\sin\theta_{\mathrm{xx1}}}{\cos\theta_{\mathrm{xx1}}} \\ &= \frac{x\left(-l_{\mathrm{t}} - l_{\mathrm{s}}\cos\theta_{\mathrm{xx2}}\right) + \sqrt{y^2+z^2}\left(-l_{\mathrm{s}}\sin\theta_{\mathrm{xx2}}\right)}{x\left(-l_{\mathrm{s}}\sin\theta_{\mathrm{xx2}}\right) + \sqrt{y^2+z^2}\left(l_{\mathrm{t}} + l_{\mathrm{s}}\cos\theta_{\mathrm{xx2}}\right)} \\ \theta_{\mathrm{xx1}} &= \arctan\left(\frac{x\left(-l_{\mathrm{t}} - l_{\mathrm{s}}\cos\theta_{\mathrm{xx2}}\right) + \sqrt{y^2+z^2}\left(-l_{\mathrm{s}}\sin\theta_{\mathrm{xx2}}\right)}{x\left(-l_{\mathrm{s}}\sin\theta_{\mathrm{xx2}}\right) + \sqrt{y^2+z^2}\left(l_{\mathrm{t}} + l_{\mathrm{s}}\cos\theta_{\mathrm{xx2}}\right)}\right) \end{aligned}$$

以上まとめると `base_link` からみた各足の位置を表すベクトル $\boldsymbol{r}_{\mathrm{xx4}}^{\mathrm{base}}$ から関節角を以下の手順で求められる．

- 脚の付け根の位置ベクトル $\boldsymbol{r}_{\mathrm{xx0}}^{\mathrm{base}}$ を引いて $[x\,y\,z]^T = \boldsymbol{r}_{\mathrm{xx4}}^{\mathrm{base}} - \boldsymbol{r}_{\mathrm{xx0}}^{\mathrm{base}}$ とする．
- $\theta_{\mathrm{xx0}} = \mathrm{atan2}\,(y, -z)$
- $\theta_{\mathrm{xx2}} = \begin{cases} \arccos\frac{x^2+y^2+z^2-l_{\mathrm{t}}^2-l_{\mathrm{s}}^2}{2l_{\mathrm{t}}l_{\mathrm{s}}} & \text{前脚の場合} \\ -\arccos\frac{x^2+y^2+z^2-l_{\mathrm{t}}^2-l_{\mathrm{s}}^2}{2l_{\mathrm{t}}l_{\mathrm{s}}} & \text{後脚の場合} \end{cases}$
- $\theta_{\mathrm{xx1}} = \mathrm{atan2}\left(\frac{x(-l_{\mathrm{t}}-l_{\mathrm{s}}\cos\theta_{\mathrm{xx2}})+\sqrt{y^2+z^2}(-l_{\mathrm{s}}\sin\theta_{\mathrm{xx2}})}{l_{\mathrm{t}}^2+2l_{\mathrm{t}}l_{\mathrm{s}}\cos\theta_{\mathrm{xx2}}+l_{\mathrm{s}}^2}, \frac{x(-l_{\mathrm{s}}\sin\theta_{\mathrm{xx2}})+\sqrt{y^2+z^2}(l_{\mathrm{t}}+l_{\mathrm{s}}\cos\theta_{\mathrm{xx2}})}{l_{\mathrm{t}}^2+2l_{\mathrm{t}}l_{\mathrm{s}}\cos\theta_{\mathrm{xx2}}+l_{\mathrm{s}}^2}\right)$

逆運動学計算を実装する．脚の付け根の関節角 $\theta_{\mathrm{xx0}}, \theta_{\mathrm{xx1}}$ は atan2 であり，引数両方がゼロの時にエラーになる可能性がある．θ_{xx0} で引数両方がゼロになるのは $y = z = 0$ のときで，この時は $\theta_{\mathrm{xx0}} = 0$ とすることに決めていた．θ_{xx1} は $\theta_{\mathrm{xx1}} = \pi$ になって脚の長さがゼロにならない限り問題は起こらない．θ_{xx2} は $\boldsymbol{r}_{\mathrm{xx4}}^{\mathrm{base}} - \boldsymbol{r}_{\mathrm{xx0}}^{\mathrm{base}}$ の長さによって定義域を外れる．脚の長さが長すぎるときは最大限に伸展した状態になるように例外処理を入れる．

9 ロボットに指令を送る方法

ロボットの制御に関連する処理は `quadruped_takahashi_control_node` の処理として実装する．ロボットが行うべき操作（モータの制御モードを切り替える・姿勢を変える・歩行する）を外部から指令する仕組みが必要である．そこで入力が文字列で真理値と文字列を返すサービス `quadruped_takahashi_control_node/mode` がある．入力文字列に応じて処理を行い，問題なければ `true` を返す．問題がある場合は `false` と問題点を文字列で返す．

サービスのコールバックには`void quadruped_takahashi_control_node::callback_mode_(...)` が設定されている．この関数は

- 前のモードを解除

- 入力の文字列に対応する関数の呼び出し

を行う．呼び出される関数は `void quadruped_takahashi_control_node::mode_{`**文字列**`}_(...)` としてある．サービスはノードの起動時に立ち上がるようにしてある．

launch ファイルによってノードを起動したら別のウインドウを開いてサービスを呼べる．例えば，モータを free するときは `ros2 service call /quadruped_takahashi_control_node/mode quadruped_takahashi/srv/Mode "{'data': 'motor_free'}"`とする．

10 ロボットをまっすぐ立たせる（stand モード）

stand モードでの処理は`void quadruped_takahashi_control_node::timer_callback_stand_()` に記述されている．このモードでの制御の目標は `base_link` の x, y 軸が水平になるようにすることであり，そのために脚の長さを変化させる．tf を参照して `map` に対する `base_link` の位置と姿勢を調べ，目標とする位置・姿勢になるようにフィードバック制御を行う．フィードバック制御は単純な PID 制御を実装した．

PID 制御は不完全微分を用いた方法で実装した．目標値との差分 $X(s)$ から制御入力 $Y(s)$ への伝達関数は以下のようになる．

$$\begin{aligned} Y(s) &= \left(K_p + \frac{K_i}{s} + K_d \frac{s}{1+\tau s} \right) X(s) \\ &= \left(K_p + \frac{K_p}{T_i s} + K_p \frac{T_d s}{1+\eta T_d s} \right) X(s) \end{aligned}$$

積分時間を T_i，微分時間を T_d，微分ゲインを $1/\eta$ とした．I 成分と D 成分は後退差分を用いて離散化する．I 成分は

$$\begin{aligned} Y_i[i] &= \frac{K_p}{T_i \frac{1-z^{-1}}{T}} X[i] \\ \frac{T_i}{T}\left(1-z^{-1}\right) Y_i[i] &= K_p X[i] \\ Y_i[i] &= Y_i[i-1] + \frac{T}{T_i} K_p X[i] \end{aligned}$$

D 成分は

$$\begin{aligned} Y_d[i] &= K_p \frac{T_d \frac{1-z^{-1}}{T}}{1+\eta T_d \frac{1-z^{-1}}{T}} X[i] \\ &= K_p \frac{T_d - T_d z^{-1}}{(T+\eta T_d) - \eta T_d z^{-1}} X[i] \\ (T+\eta T_d)\, Y_d[i] &= \eta T_d Y_d[i-1] + K_p T_d X[i] - K_p T_d X[i-1] \\ Y_d[i] &= \frac{\eta T_d}{T+\eta T_d} Y_d[i-1] + \frac{K_p T_d}{T+\eta T_d} X[i] - \frac{K_p T_d}{T+\eta T_d} X[i-1] \end{aligned}$$

となる．

全成分を計算するためには今回の入力 $X[i]$ に加えて前回の入出力 $X[i-1], Y_i[i-1], Y_d[i-1]$ の情報も必要になる．そこで，static 変数を用いて前回の情報を保持するよう，以下のように実装した．

```
pid_control(x) {
  const Kp  = 0.7;
  const Ti  = 100000;
  const Td  = 0.0;
  const T   = control_period;
  const eta = 0.1;

  static x_last = 0.0;
  static yi     = 0.0;
  static yd     = 0.0;

  yi      = yi + Kp * T / Ti * x;
  yd      = (eta * Td) / (T + eta * Td) * yd
          + (Kp * Td) / (T + eta * Td) * x
          - (Kp * Td) / (T + eta * Td) * x_last;
  auto yp = Kp * x;
  auto y  = yp + yi + yd;
  x_last  = x;

  return y;
}
```

ゲインの調整は資料[2] を参考にした．微分ゲインは 10～8（$\eta = 0.1 \sim 0.125$）とされており，今回は 0.1 に設定した．最初は積分時間 T_i は大きな値，微分時間 T_d はゼロに設定して比例制御のみが行われるようにした状態で比例ゲイン K_p を設定してから T_i, T_d を定めた．

試行錯誤したが微分制御は発散するので微分時間はゼロのままになっている．現状の実装では指令値が変化したときに，瞬間的に大きな出力が出ることが知られている．この問題を回避するために，微分先行型と呼ばれる PI-D 制御や I-PD 制御が存在している．これらの手法を適用することでこの問題は解消されるのかもしれないが，現状では理解が不足している．

現状での動作状態の動画を撮影し Twitter にアップロードした．動画はこのツイート[3]で確認できる．

2） 広井和男，PID 制御のお話 第 10 回 実用形態に向けての工夫（その 2）．エムエスツデー，2004.
html: https://www.m-system.co.jp/mstoday1/MSTback/data/2004/11/PID_T.htm
pdf: https://www.m-system.co.jp/rensai/pdf/r0411.pdf

3） https://twitter.com/Actat85/status/1633119309986775045?s=20