インタラクティブシステム論 第11回

梶本裕之

日程

講義番号	講義日	講義内容	pdf	video	レポート締め切り
1	4/9	イントロダクション	[pdf] 2020年版	video	4/16
		Scilab課題	[pdf] 2020年版		↑
		上記資料のPython版	[pdf] 2020年版		↑
2	4/16	フーリエ変換	[pdf] 2020年版	video	4/23
3	4/23	フーリエ変換と線形システム	[pdf] 2020年版	video	4/30
4	4/30	信号処理の基礎	[pdf] 2020年版	video	5/7
5	5/7	信号処理の応用1(相関)	[pdf] 2020年版	video	5/14
6	5/14	信号処理の応用2(画像処理)	[pdf] 2020年版	video	5/21
-	5/21	中間確認テスト準備（自習）	[pdf]2020年版		
-	5/28	中間確認テスト（現在は大学を予定）	[pdf]2020年版		
7	6/4	ラプラス変換	[pdf] 2020年版	video	6/11
8	6/11	古典制御の基礎	[pdf] 2020年版	video	6/18
9	6/18	行列	[pdf] 2020年版	video	6/25
10	6/25	行列と最小二乗法	[pdf] 2020年版	video	7/2
11	7/2	ロボティクス	[pdf] 2020年版	video	7/9
-	7/9	期末テスト準備（自習）	[pdf]2020年版		
-	7/16	期末確認テスト（現在は大学を予定）			

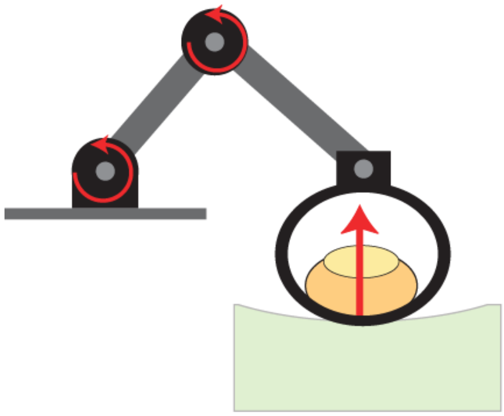
日程およびテストを大学で行うかについては、随時Google Classroomや授業のページを見てください。

ロボティクスの
基礎の基礎：

ロボットの姿勢・力・速度

ロボット

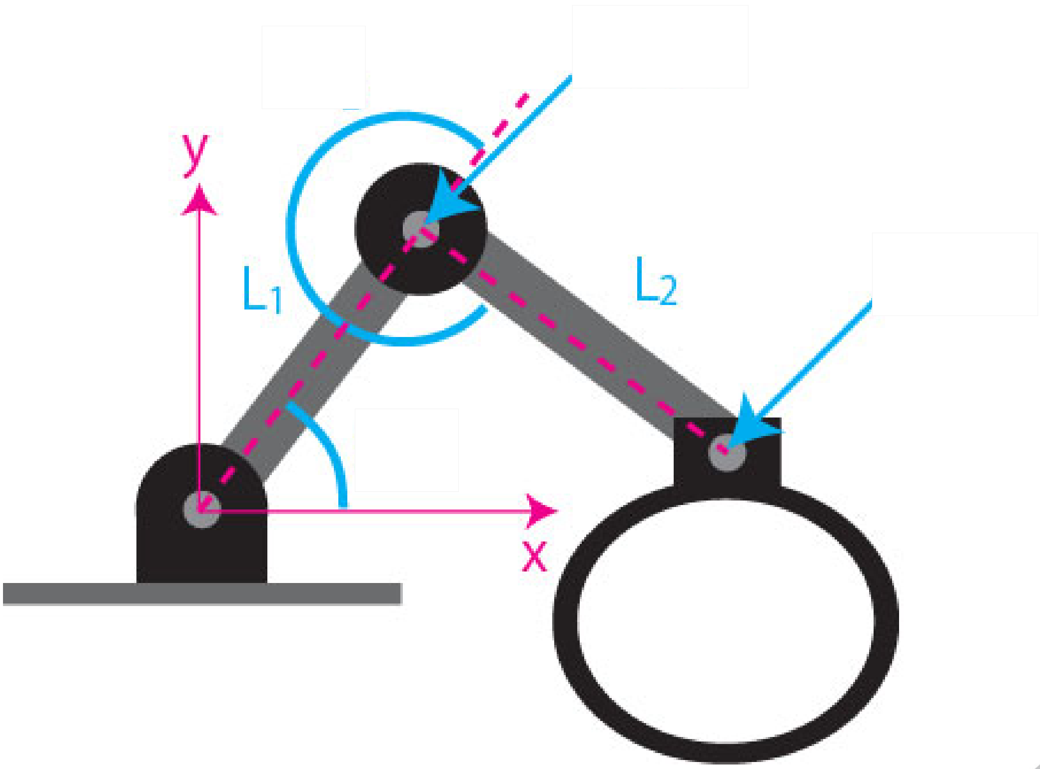
座標変換の必要性



- 関節の角度から，ロボット末端の位置を知りたい．
- ロボット末端の位置から，関節の角度を知りたい．

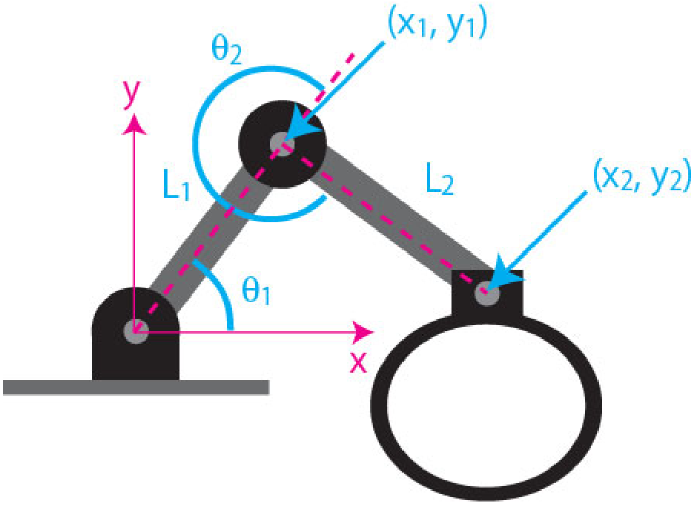
- ロボット末端をある速度で動かすための関節の速度は？
- ロボット末端にある力を出すための，関節のトルクは？

座標の定義



順キネマティクス

関節の角度(θ1,θ2)から，
ロボット末端の位置(x2,y2)を知りたい．



$x_1 =$

$y_1 =$

$x_2 =$

$=$

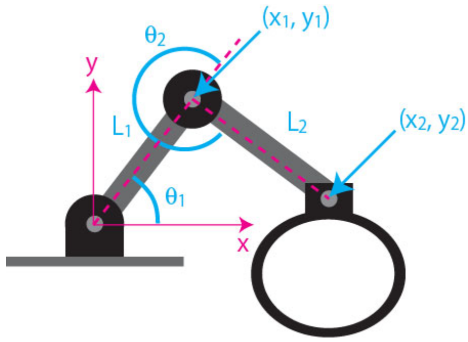
$y_2 =$

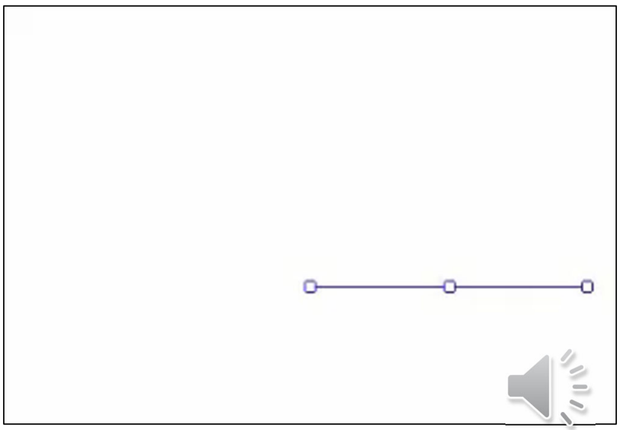
$=$

順キネマティクスのシミュレーション

Scilabコード

```
L1 = 1.0;
L2 = 1.0;

for t=0:0.1:%pi,
  theta1 = t; //関節１の角度
  theta2 = t*2;//関節2の角度
  //関節座標
  x1 = L1 * cos(theta1);
  y1 = L1 * sin(theta1);
  x2 = x1 + L2 * cos(theta1+theta2);
  y2 = y1 + L2 * sin(theta1+theta2);

  armX = [0,x1,x2]; //関節のx座標
  armY = [0,y1,y2]; //関節のy座標

  plot(armX,armY,'O-'); //描画
  sleep(100); //100ms休む
end
```

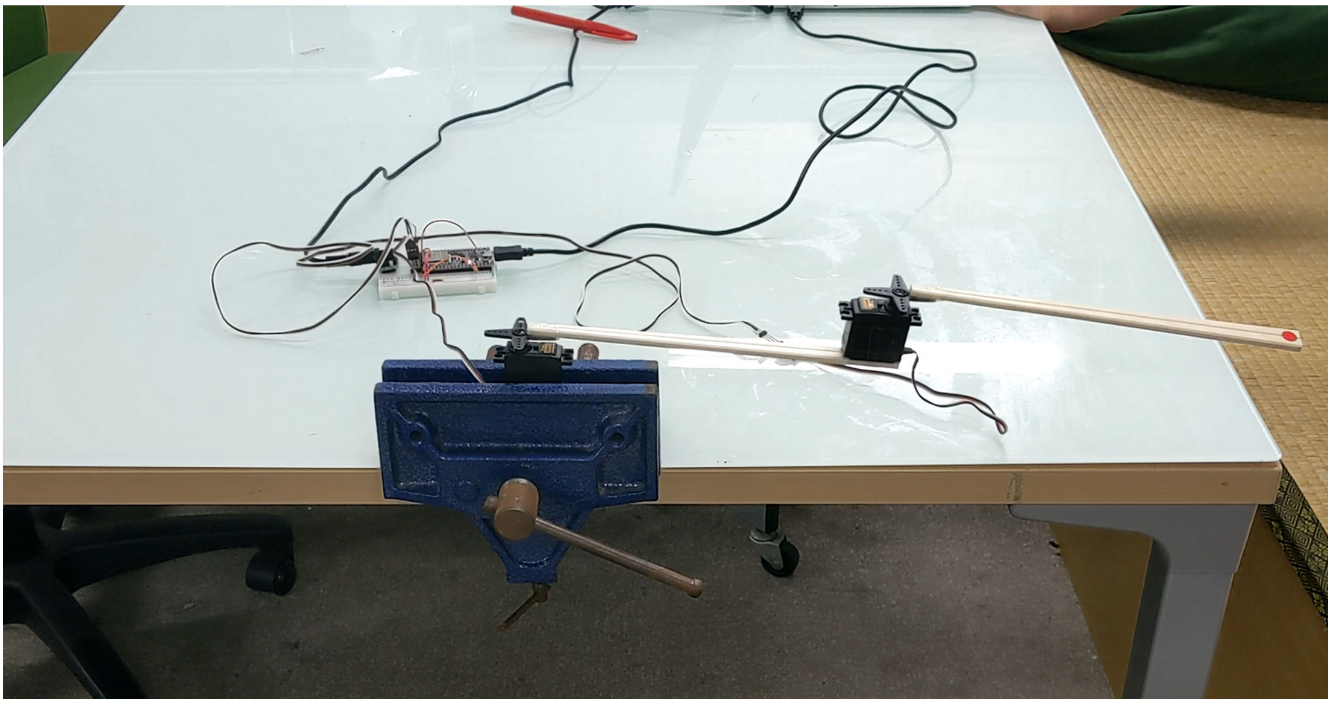




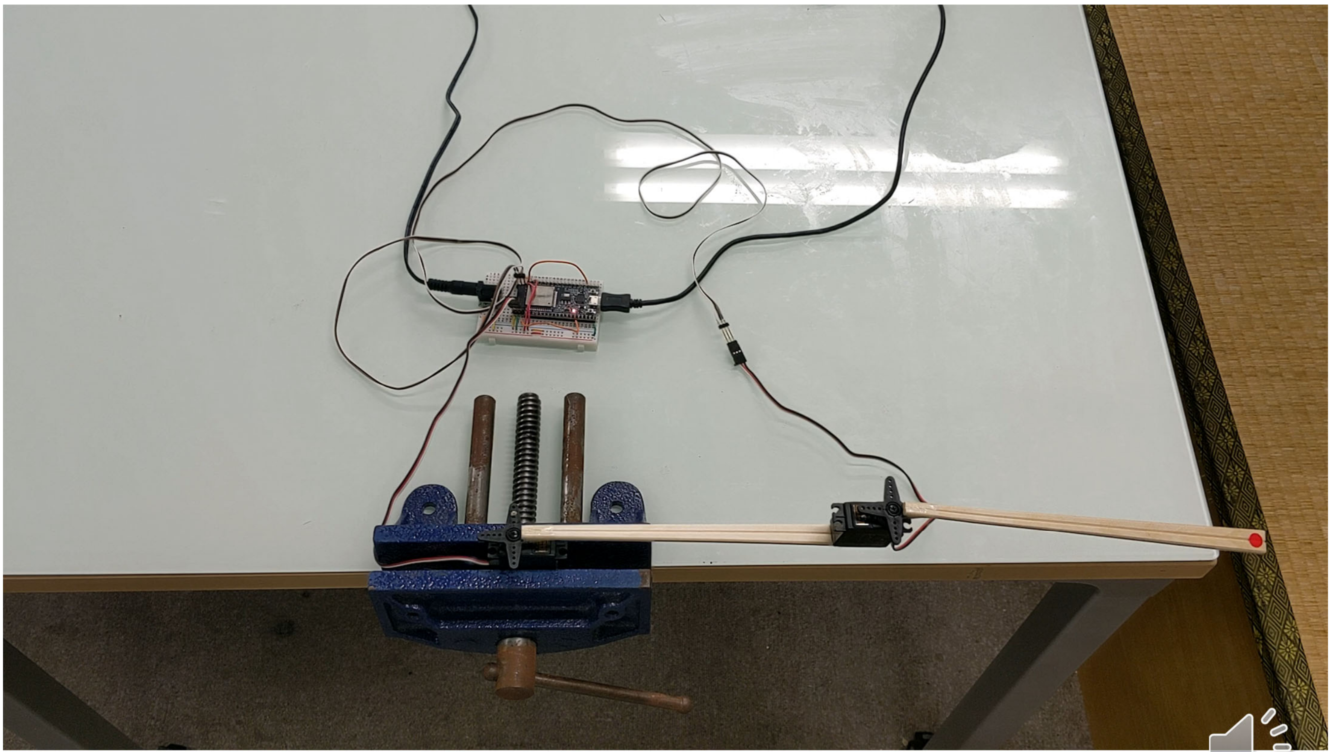
(参考)pythonコード

```
import matplotlib.pyplot as plt
import numpy as np
import matplotlib.animation as animation
fig = plt.figure()
ims = []
L1 = 1.0
L2 = 1.0
time = np.arange(0,np.pi,0.1)
record = list()
for t in time:
    theta1 = t
    theta2 = t * 2
    x1 = L1 * np.cos(theta1)
    y1 = L1 * np.sin(theta1)
    x2 = x1 + L2 * np.cos(theta1 + theta2)
    y2 = y1 + L2 * np.sin(theta1 + theta2)
    armX = [0,x1,x2]
    armY = [0,y1,y2]
    im = plt.plot(armX,armY,marker='o')
    ims.append(im)
ani = animation.ArtistAnimation(fig,ims,interval=100)
plt.axes().set_aspect('equal','datalim')
plt.show()
```

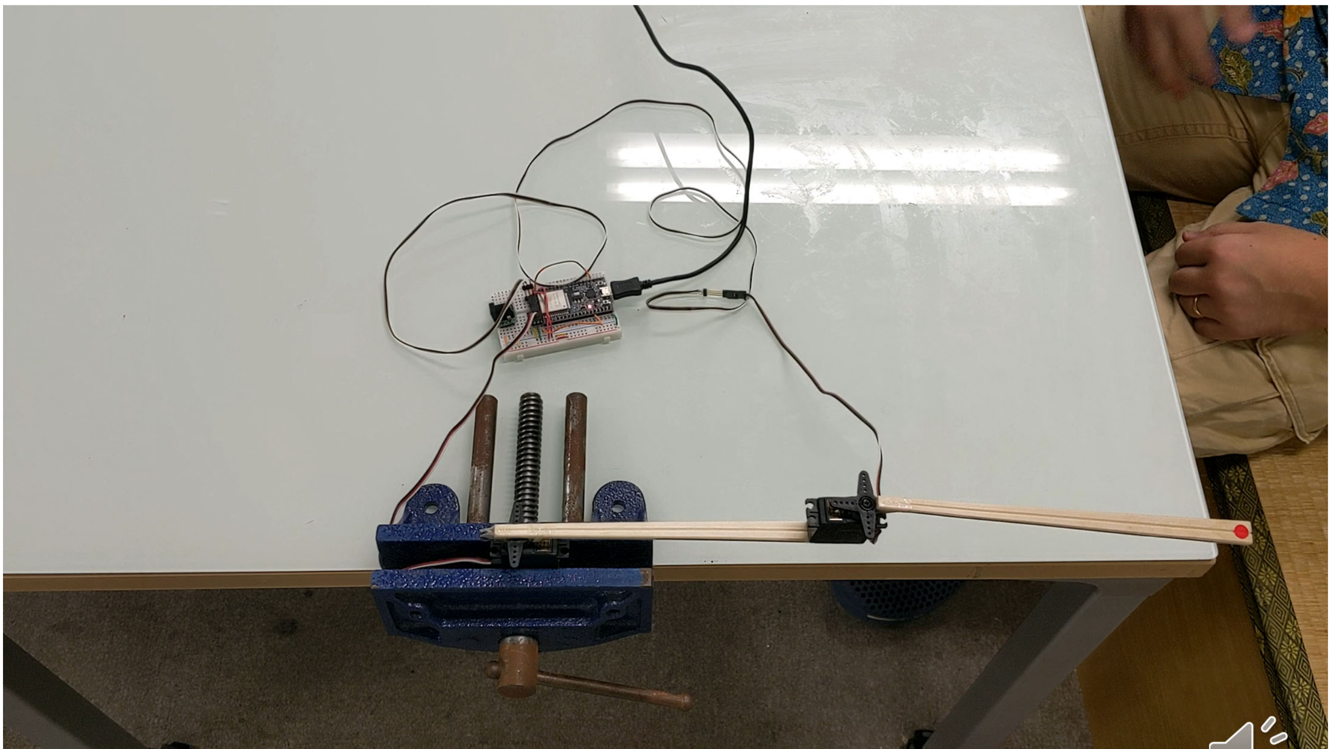
２軸ロボット（ラジコンサーボ+割り箸）



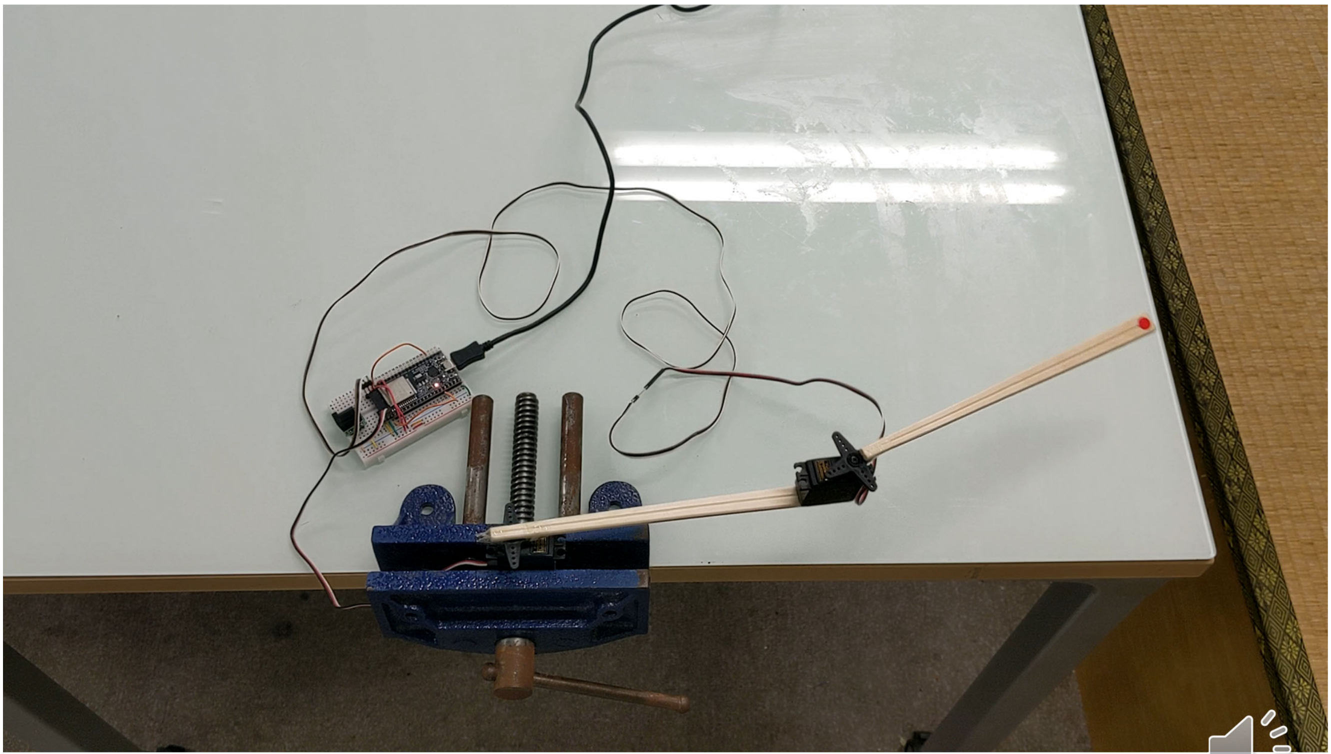
$\theta_1 = \theta_2$



$\omega_1 : \omega_2 = 1 : 1$



$\omega_1 : \omega_2 = 1 : 2$

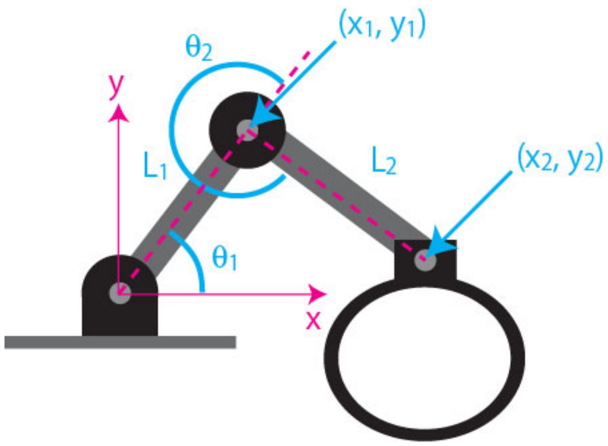


POSER（モデリングソフトの一つ）中の順キネマティクス

ロボティクスの知識はCGアニメーションに必須.（基礎的な知識は共通）

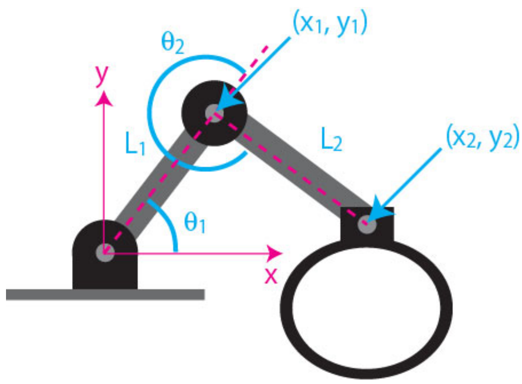
逆キネマティクス

ロボット末端の位置を(x2,y2) に移動したい.
関節の角度(θ1,θ2) は何度回せば良いか？



$$x_2 = L_1 \cos\theta_1 + L_2 \cos(\theta_1 + \theta_2)$$
$$y_2 = L_1 \sin\theta_1 + L_2 \sin(\theta_1 + \theta_2)$$

逆キネマティクス



θ1+θ2をθ12と書いて

$$x_2 = L_1 \cos\theta_1 + L_2 \cos\theta_{12}$$
$$y_2 = L_1 \sin\theta_1 + L_2 \sin\theta_{12}$$

$$x_2{}^2 + y_2{}^2 =$$

$$=$$

$$=$$

余弦定理

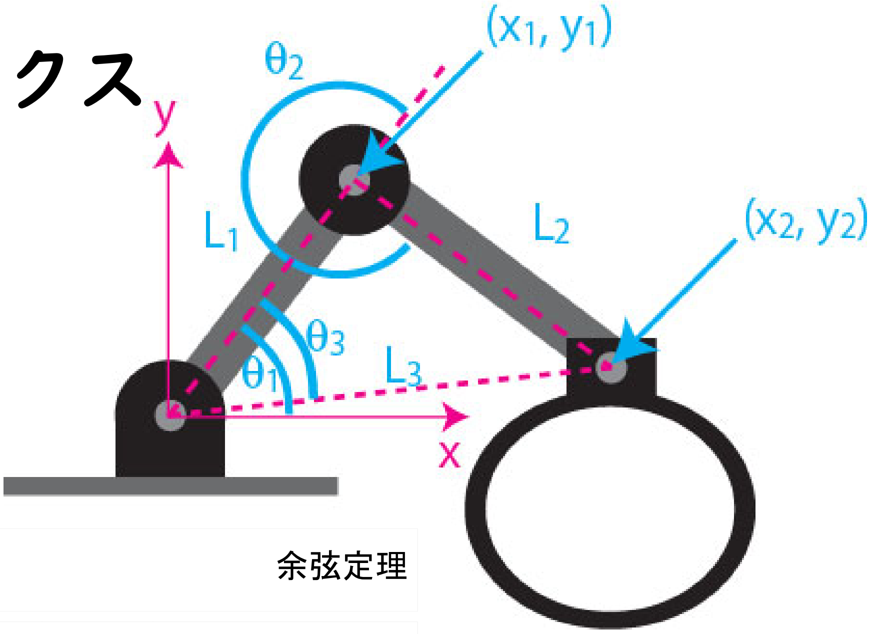
$$\cos(\theta_2) =$$

$$\theta_2 =$$

任意性あり

逆キネマティクス

L3，θ3を定義



$$L_3 = \sqrt{x_2^{\ 2} + y_2^{\ 2}}$$

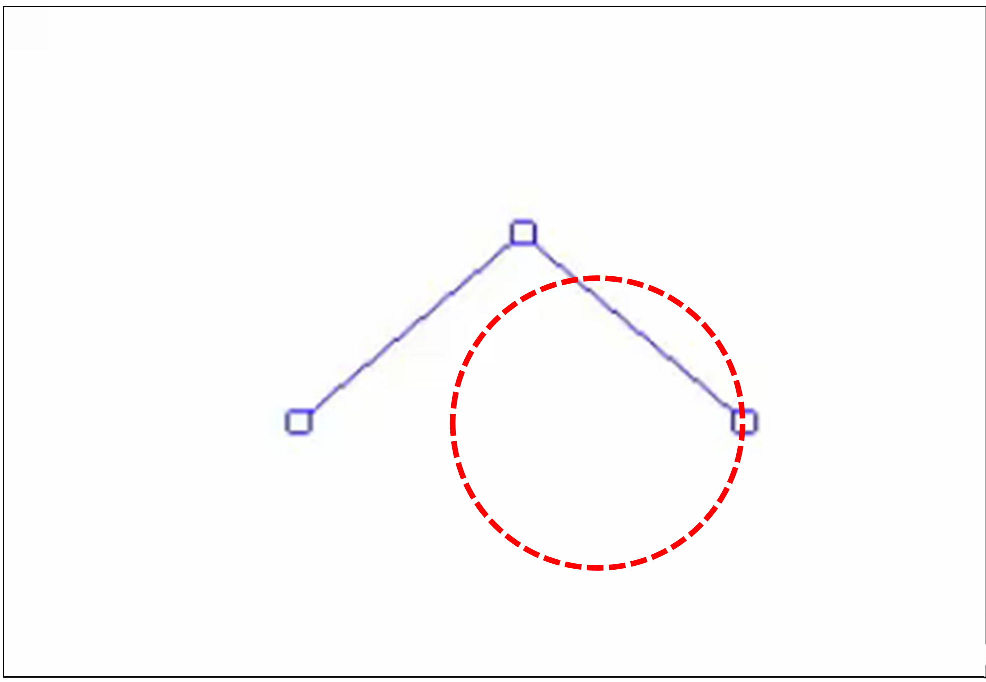
$$L_2^{\ 2} =$$

余弦定理

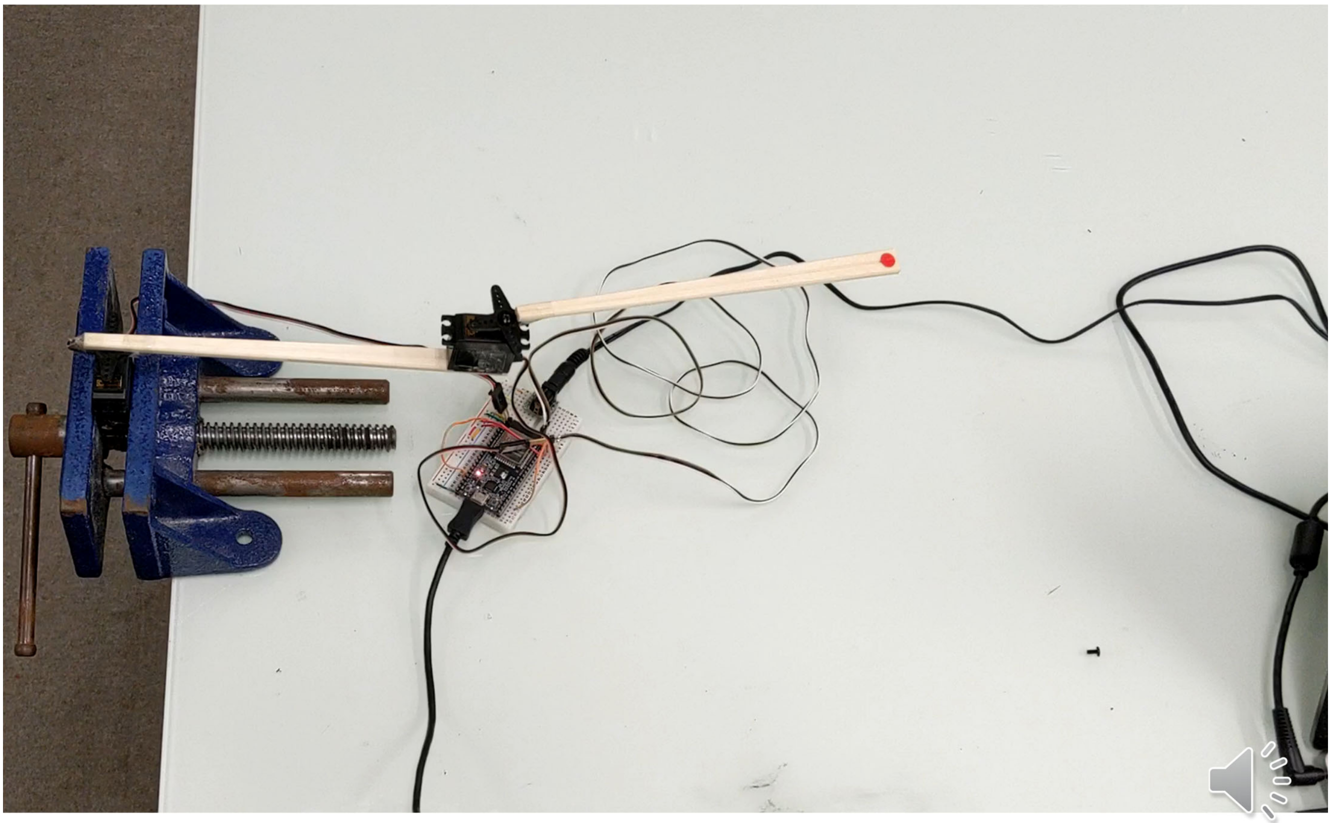
$$\theta_3 =$$

$$\theta_1 =$$

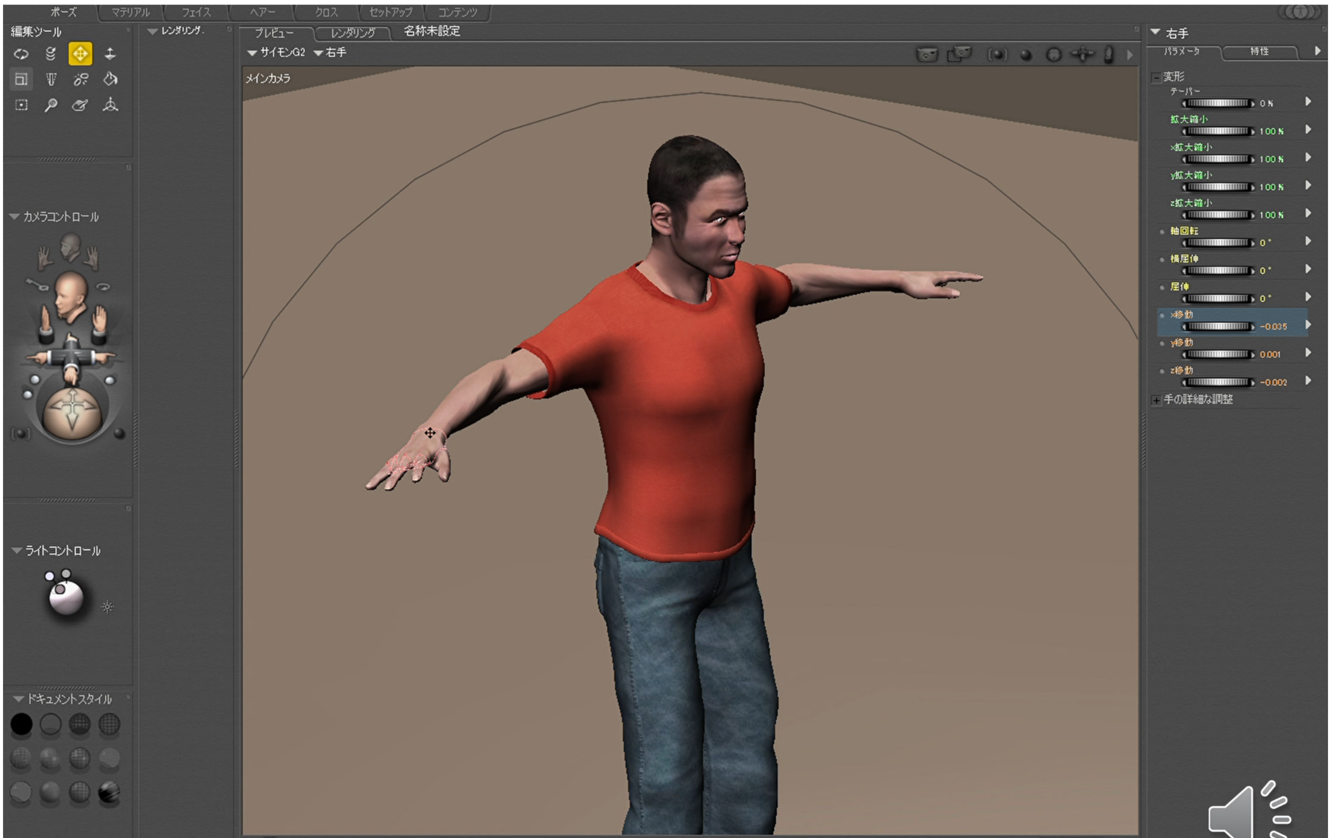
逆キネマティクス:
先端に円を描かせるシミュレーション結果



２軸ロボットでの逆キネマティクス



POSER中の逆キネマティクス



ロボティクスの知識はCGアニメーションに必須.（基礎的な知識は共通）

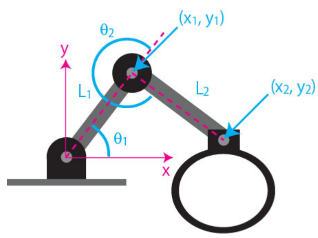
レポート課題

以下は逆キネマティクスの式を用いてロボット先端に円を描かせたプログラムである．完成させよ．
※acosには正負の任意性がある事に注意．上手くいくよう調節

```
L1 = 1.0;
L2 = 1.0;

for t=0:0.1:2*%pi,
 //目標の先端位置．円を描かせる
 x2 = 1+0.5*cos(t);
 y2 = 0.5*sin(t);

 L3 =
 theta2 =
 theta3 =

 theta1 =
```

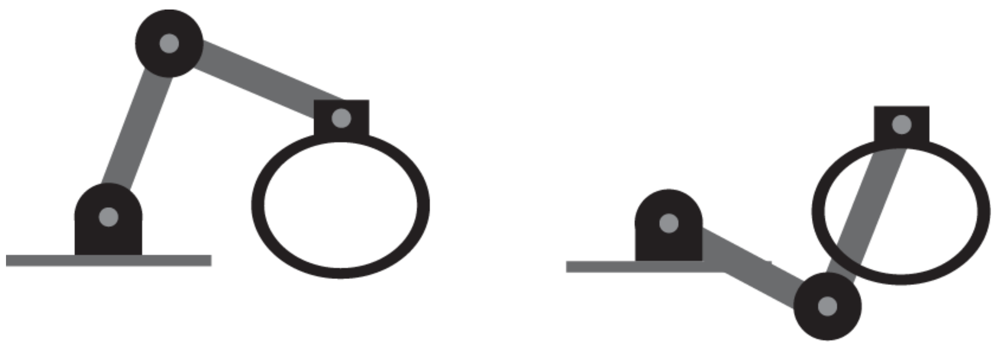
```
 //以下は順キネティクス
 x1 = L1 * cos(theta1);
 y1 = L1 * sin(theta1);
 x2 = x1 + L2 * cos(theta1+theta2);
 y2 = y1 + L2 * sin(theta1+theta2);

 //以下は描画用
 armX = [0,x1,x2]; //関節のx座標
 armY = [0,y1,y2]; //関節のy座標
 square(-2.5,-2.5,2.5,2.5);
 plot(armX,armY,'O-'); //描画
 sleep(100); //100ms休む
end
```

逆キネマティクスまとめ



ロボット末端の位置を(x2,y2) に移動したい．
関節の角度(θ1,θ2) は何度回せば良いか？

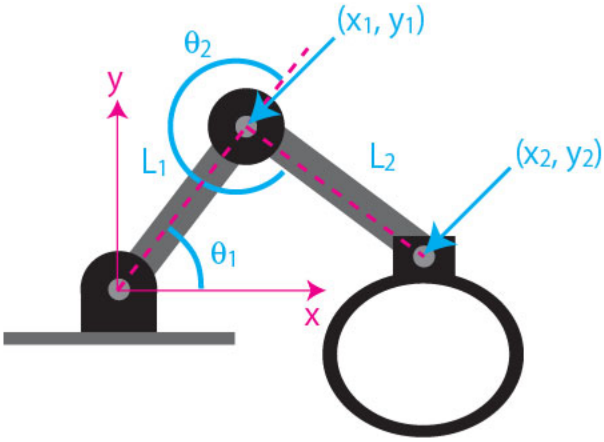
頑張って式変形し，θ1，θ2をx2，y2で表す．
一般的な解法は無い．<u>とても大変</u>．
解が複数個あることも．



先端速度の計算

関節の速度$\dot{\theta}_1, \dot{\theta}_2$からロボット末端の速度を計算.

$$x_2 = L_1 \cos\theta_1 + L_2 \cos(\theta_1 + \theta_2)$$
$$y_2 = L_1 \sin\theta_1 + L_2 \sin(\theta_1 + \theta_2)$$

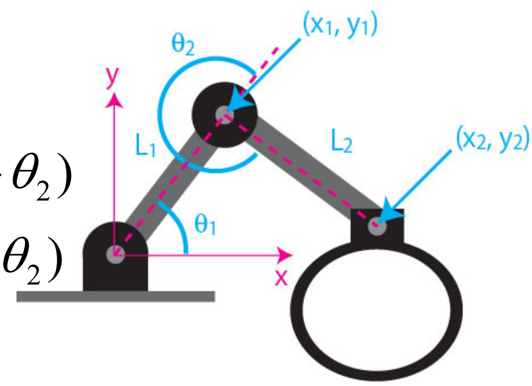


例えば $\dfrac{d\cos\theta_1}{dt} = -\dot{\theta}_1 \sin\theta_1$ から

$$\dot{x}_2 =$$
$$\dot{y}_2 =$$

先端速度の計算

$$\dot{x}_2 = -L_1\dot{\theta}_1 \sin\theta_1 - L_2(\dot{\theta}_1 + \dot{\theta}_2)\sin(\theta_1 + \theta_2)$$
$$\dot{y}_2 = L_1\dot{\theta}_1 \cos\theta_1 + L_2(\dot{\theta}_1 + \dot{\theta}_2)\cos(\theta_1 + \theta_2)$$



$$\begin{bmatrix} \dot{x}_2 \\ \dot{y}_2 \end{bmatrix} = \begin{bmatrix} \quad & \quad \\ \quad & \quad \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix}$$

これを $\begin{bmatrix} \dot{x}_2 \\ \dot{y}_2 \end{bmatrix} = \mathbf{J} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix}$ と書き，J をヤコビアンと呼ぶ.

ヤコビアンは時々刻々と変化する. 毎サイクル計算

ヤコビアン

- 元の式

$$x_2 = L_1 \cos\theta_1 + L_2 \cos(\theta_1 + \theta_2)$$
$$y_2 = L_1 \sin\theta_1 + L_2 \sin(\theta_1 + \theta_2)$$

- 一般的に

$$x_2 = f(\theta_1, \theta_2)$$
$$y_2 = g(\theta_1, \theta_2)$$

- 偏微分で

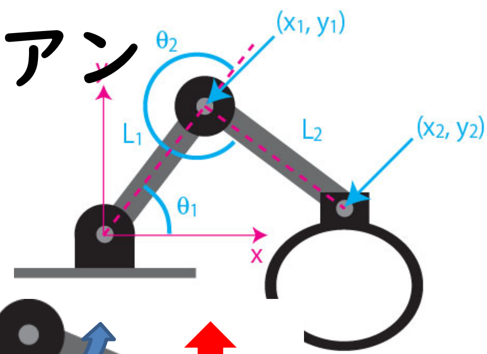
$$\dot{x}_2 = \frac{\partial f}{\partial \theta_1}\dot{\theta}_1 + \frac{\partial f}{\partial \theta_2}\dot{\theta}_2$$
$$\dot{y}_2 = \frac{\partial g}{\partial \theta_1}\dot{\theta}_1 + \frac{\partial g}{\partial \theta_2}\dot{\theta}_2$$

- まとめると

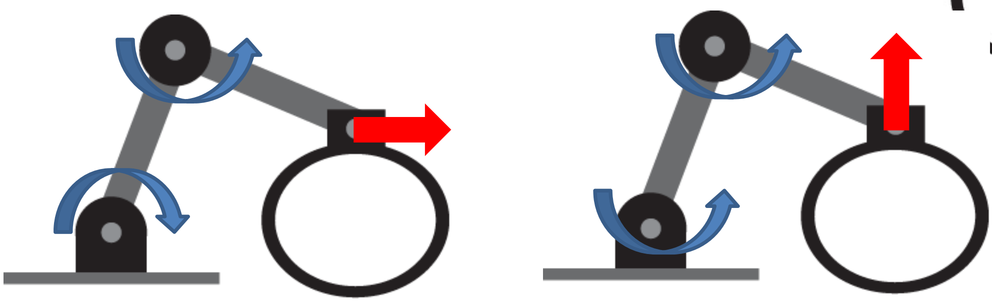
ヤコビアン

$$\begin{bmatrix} \dot{x}_2 \\ \dot{y}_2 \end{bmatrix} = \boxed{\begin{bmatrix} \dfrac{\partial f}{\partial \theta_1} & \dfrac{\partial f}{\partial \theta_2} \\ \dfrac{\partial g}{\partial \theta_1} & \dfrac{\partial g}{\partial \theta_2} \end{bmatrix}} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix}$$

先端速度の実現とヤコビアン



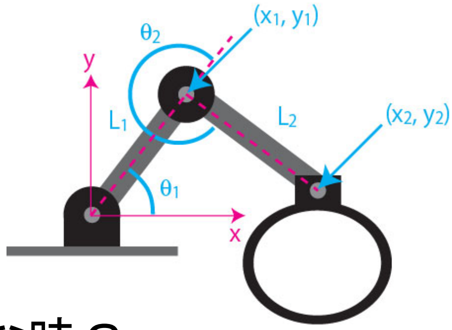
ロボット末端をある速度で動かす
ための関節の角速度は？



$$\begin{bmatrix} \dot{x}_2 \\ \dot{y}_2 \end{bmatrix} = \mathbf{J} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} \quad \Rightarrow \quad \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} = \mathbf{J}^{-1} \begin{bmatrix} \dot{x}_2 \\ \dot{y}_2 \end{bmatrix}$$

- ヤコビアンの逆行列で求めることができる！
- 逆行列がない場合は？

特異点



$$\mathbf{J} = \begin{bmatrix} -L_1 \sin\theta_1 - L_2 \sin(\theta_1+\theta_2) & -L_2 \sin(\theta_1+\theta_2) \\ L_1 \cos\theta_1 + L_2 \cos(\theta_1+\theta_2) & L_2 \cos(\theta_1+\theta_2) \end{bmatrix}$$

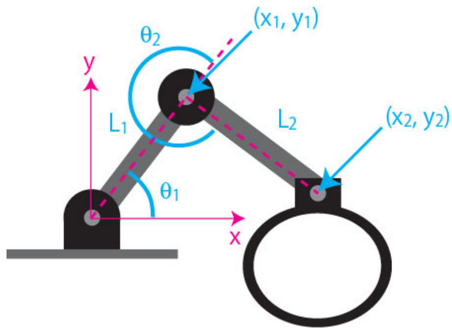
ヤコビアンの逆行列がないのはどんな時？

ad – bc = 0より

$$\left(-L_1 \sin\theta_1 - L_2 \sin\theta_{12}\right) L_2 \cos\theta_{12} + \left(L_1 \cos\theta_1 + L_2 \cos\theta_{12}\right) L_2 \sin\theta_{12} = 0$$

ただし $\theta_{12} = \theta_1 + \theta_2$

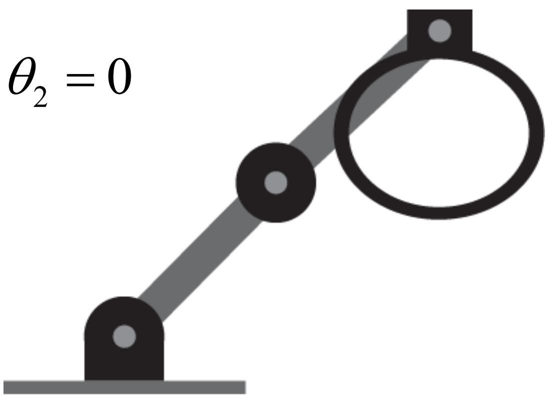
特異点



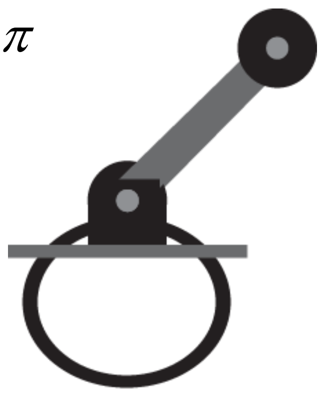
ヤコビアンの逆行列がないのは

$\theta_2 = 0, \pi$　のとき.

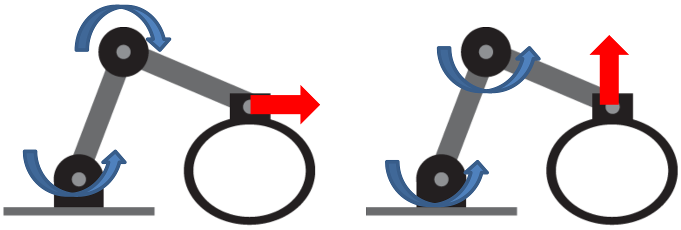
$\theta_2 = 0$

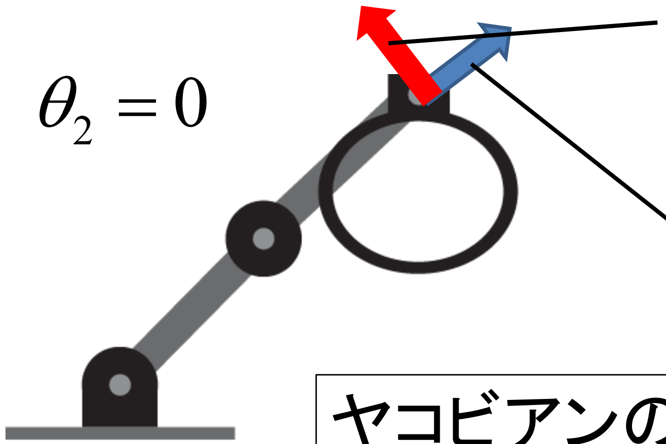


$\theta_2 = \pi$



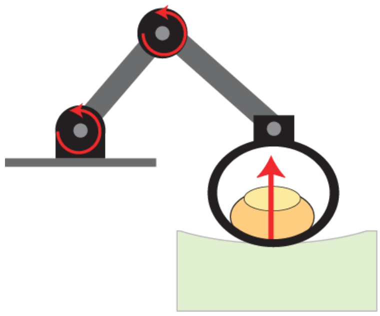
特異点と速度





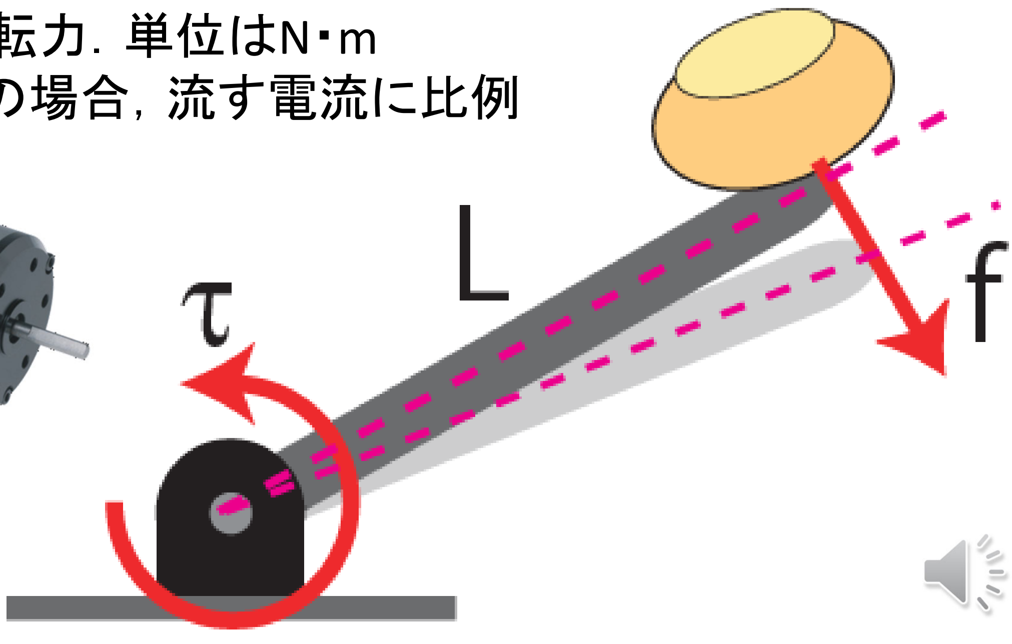
ヤコビアンの逆行列がない
＝動かせない方向あり（特異点）

先端力の実現とヤコビアン

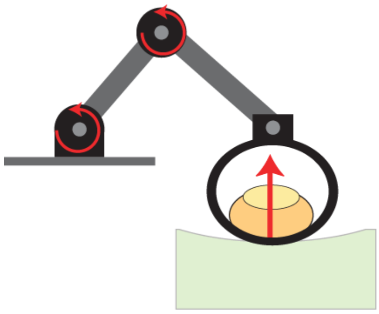


ロボット末端にある力を出すための，
関節のトルクは？

トルク：回転力．単位はN・m
DCモータの場合，流す電流に比例



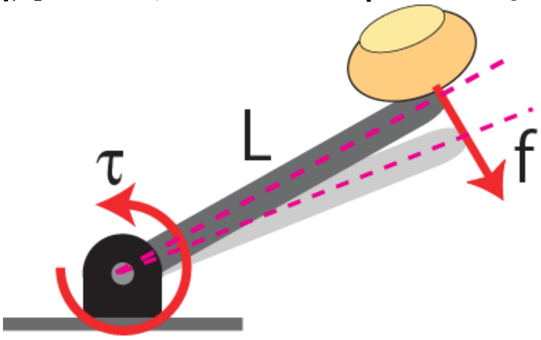
先端力の実現とヤコビアン



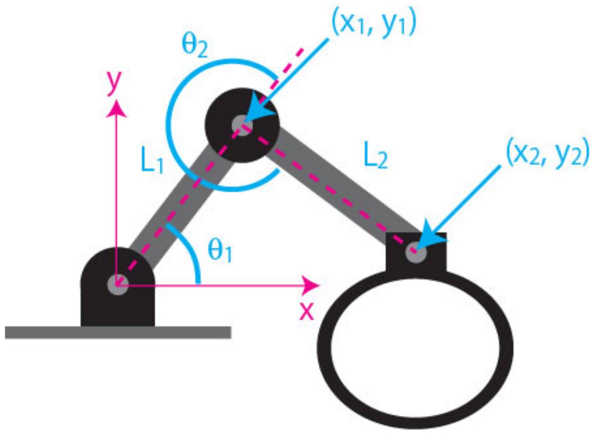
ロボット末端にある力を出すための，
関節のトルクは？

＜仮想仕事の原理＞を用いる
力 f で dx だけ微小変位したとき　仕事 = f･dx
トルクτで dθ だけ微小回転したときの仕事 = τ･dθ

例えばアーム一本のロボットでは，この二つが釣り合うから，



２軸の場合



＜仮想仕事の原理＞を用いる
モータの出すトルクによる仕事：

$$W_{motor} =$$

先端の力による仕事：

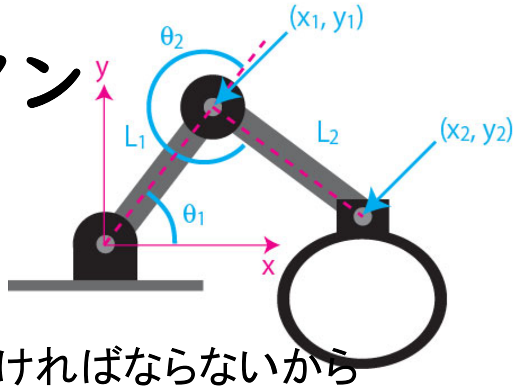
$$W_{hand} =$$

これが釣り合うから，Wmotor=Whandとなり，

$$= \qquad =$$

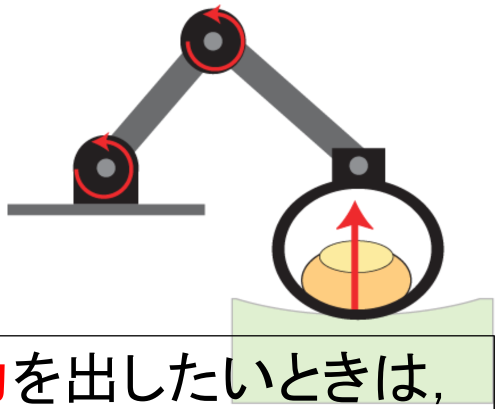
$$\left(\therefore \begin{bmatrix} \dot{x}_2 \\ \dot{y}_2 \end{bmatrix} = \mathbf{J} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} \right)$$

先端力の実現とヤコビアン



$$\begin{bmatrix}\tau_1 & \tau_2\end{bmatrix}\begin{bmatrix}\dot{\theta}_1 \\ \dot{\theta}_2\end{bmatrix} = \begin{bmatrix}f_x & f_y\end{bmatrix}\mathbf{J}\begin{bmatrix}\dot{\theta}_1 \\ \dot{\theta}_2\end{bmatrix}$$

これが任意の角速度 $\dot{\theta}_1, \dot{\theta}_2$ で成立しなければならないから



$\therefore$

つまりロボット先端にある力を出したいときは，
ヤコビアンの転置をかけることにより，
関節に必要なトルクに変換できる．

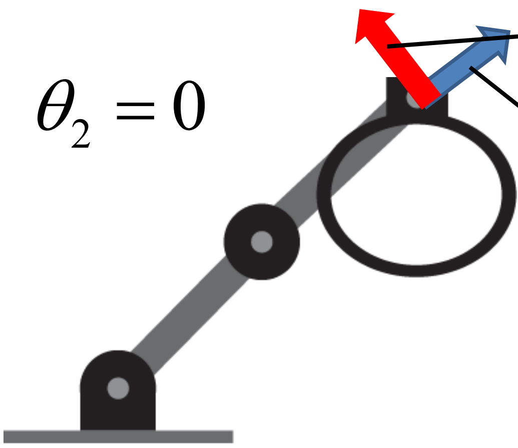
特異点と先端力

モータにあるトルクを加えた時，先端にかかる力を求める．

$$\begin{bmatrix}\tau_1 \\ \tau_2\end{bmatrix} = \mathbf{J}^T\begin{bmatrix}f_x \\ f_y\end{bmatrix}$$

ヤコビアンの逆行列がない＝力の出ない方向あり（特異点）

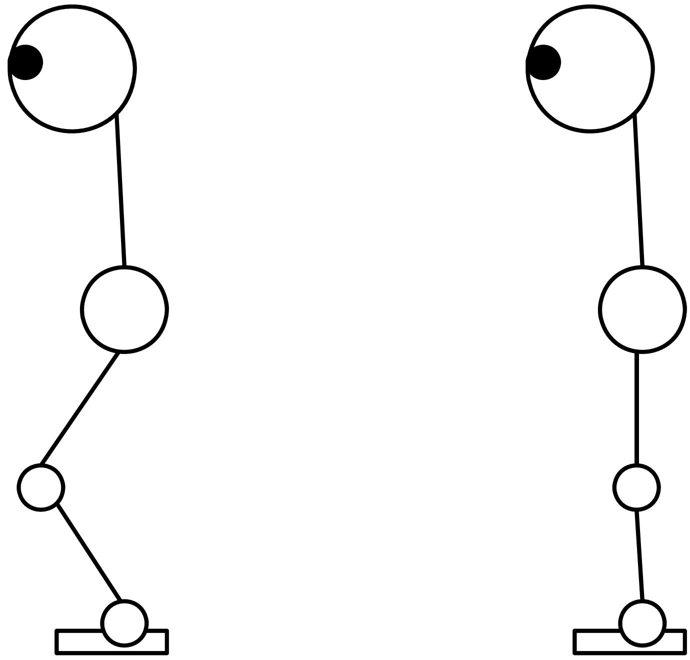
$$\theta_2 = 0$$



この方向には力を出せる

関節をどう動かしても
この方向に力を出せない

特異点と先端力（おまけ）



初期の二足歩行ロボット．　　現在の二足歩行ロボット、人間

ロボティクスの基礎の基礎：まとめ

●ロボット各関節の角度，角速度，トルク

●ロボット先端の位置，速度，力

この二つは相互に変換可能である．
変換には単純な幾何学の知識と，ヤコビアンの知識が必要
これらのロボティクスの知識は，CGの基礎知識でもある．