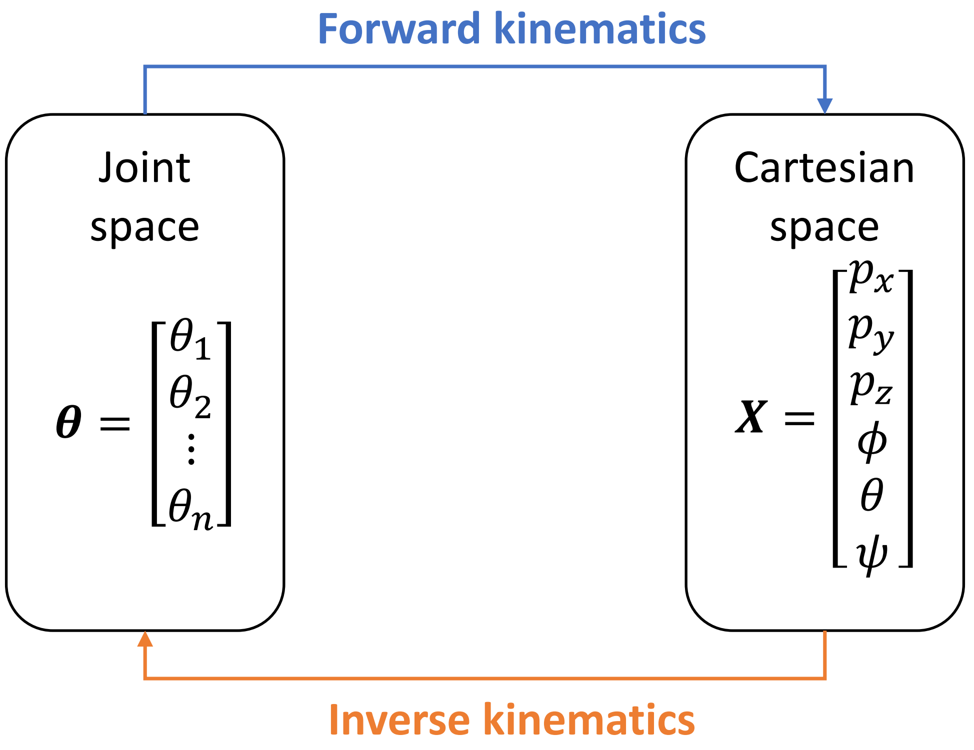
ME729 Advanced Robotics - Inverse Kinematics

2/12/2018

Sangsin Park, Ph.D.

Inverse Kinematics

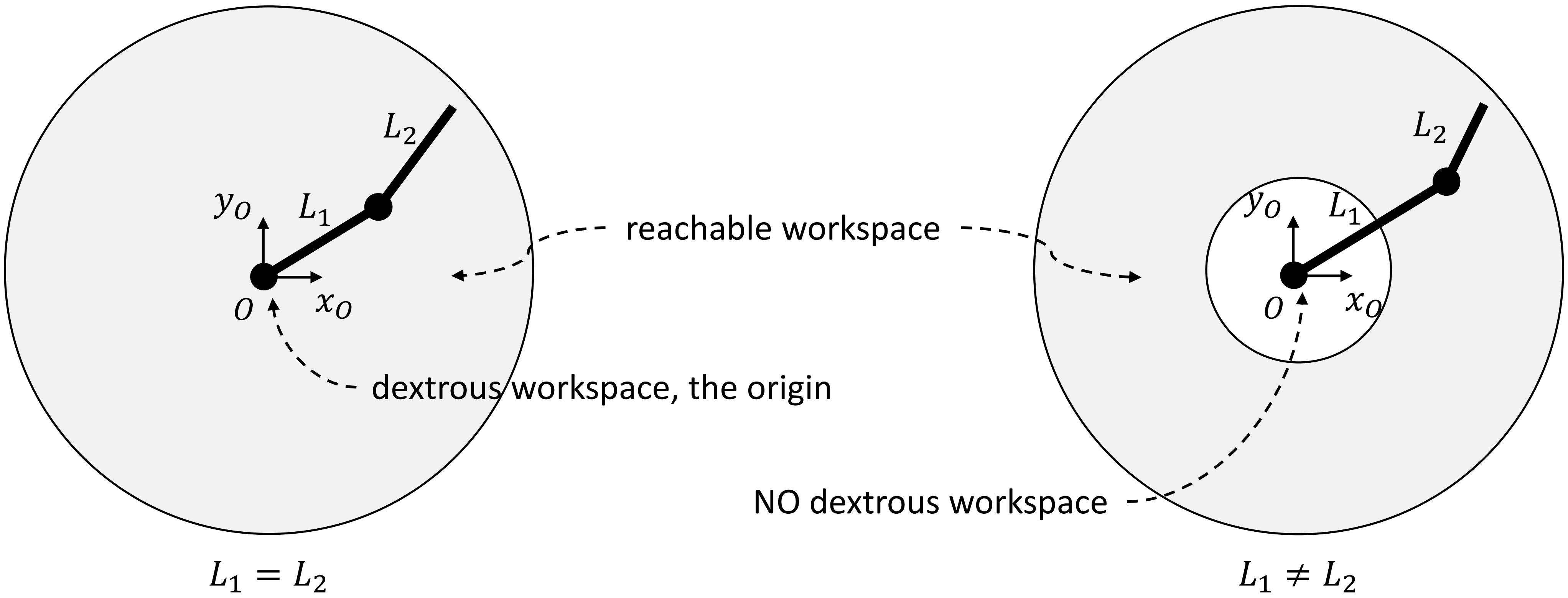
❑ Mappings between kinematic descriptions



- Joint space: A set of n joint variables is referred to as the $n \times 1$ joint vector. The space of all such joint vectors is referred to as joint space.
- Cartesian space: The term is used when position is measured along orthogonal axes, and orientation is measured to any of the conventions said before.

Inverse Kinematics

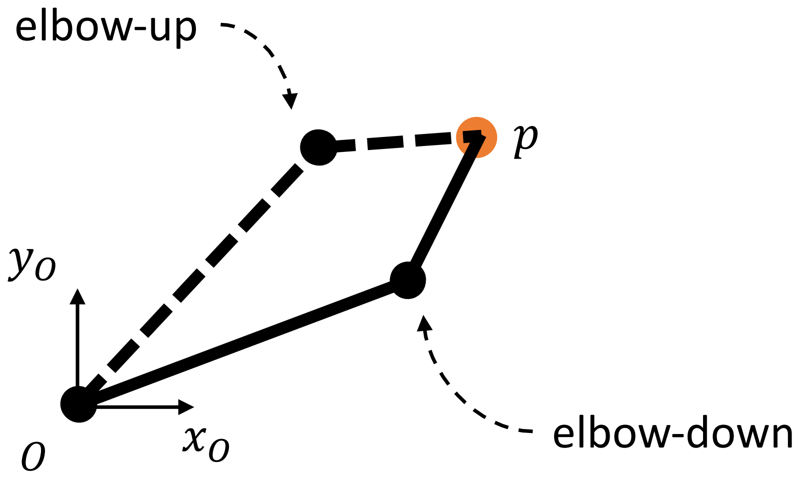
- Existence of solutions
  - Workspace: volume of space which **the end-effector of the manipulator can reach**.
  - Dextrous workspace: volume of space which the robot end-effector can reach **with all orientation**.
  - Reachable workspace: volume of space which the robot can reach **in at least one orientation**.
  - For example, consider the two-link planar manipulator when $\boldsymbol{L_1 = L_2}$ and $\boldsymbol{L_1 \neq L_2}$.

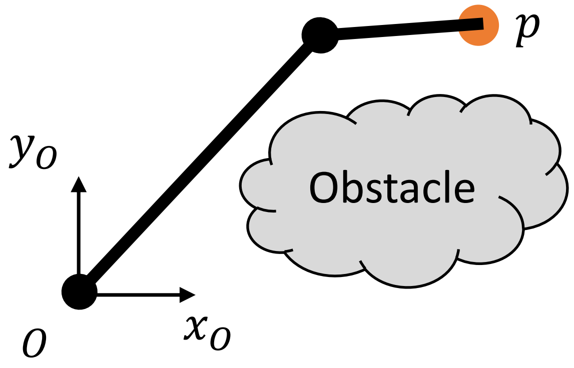


Inverse Kinematics

❑ Multiple solutions

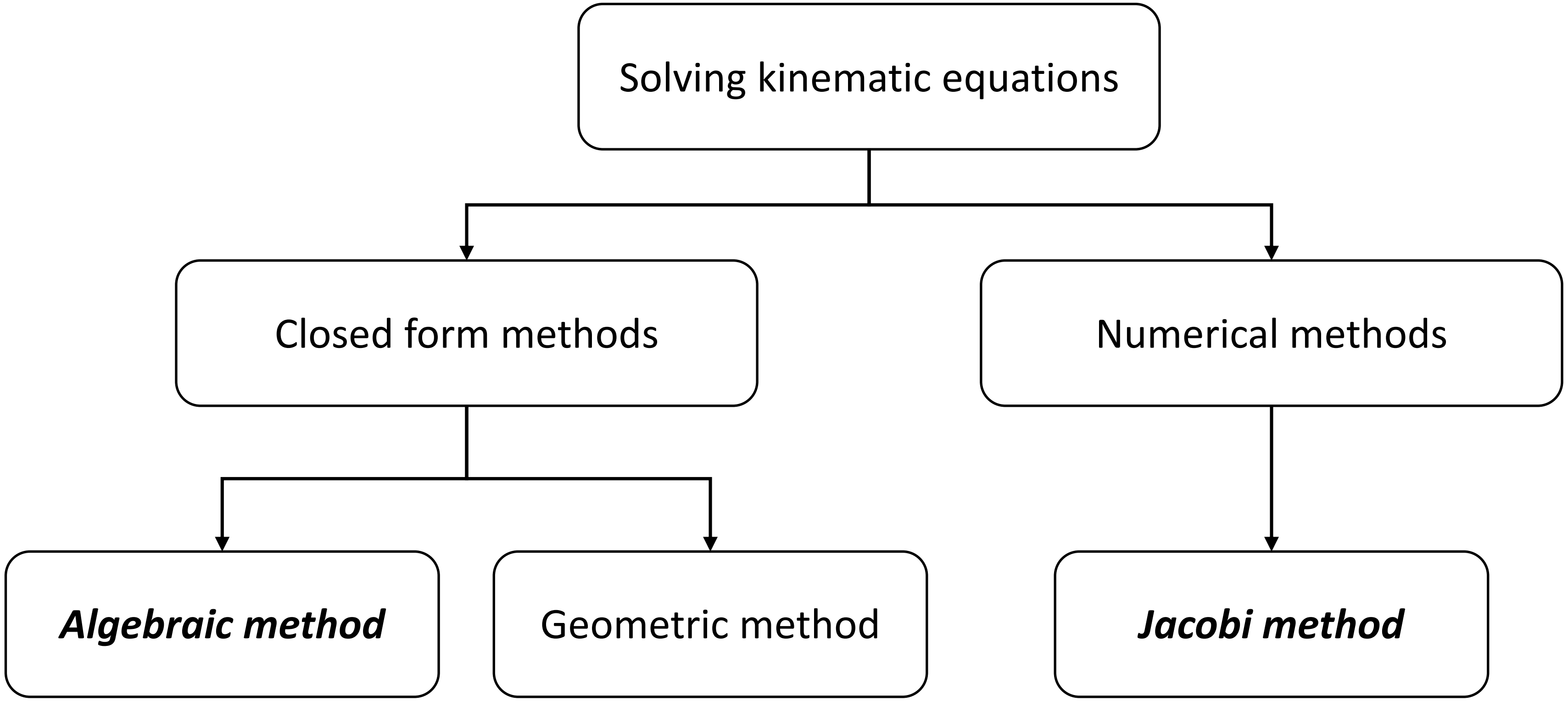
- When solving kinematic equations, we encounter multiple solutions.
- Let's see the two-link planar manipulator.
- To reach a point p, the robot can have two configurations: elbow up and down.
- In the present of the obstacle, the elbow up configuration would be chosen.





Inverse Kinematics

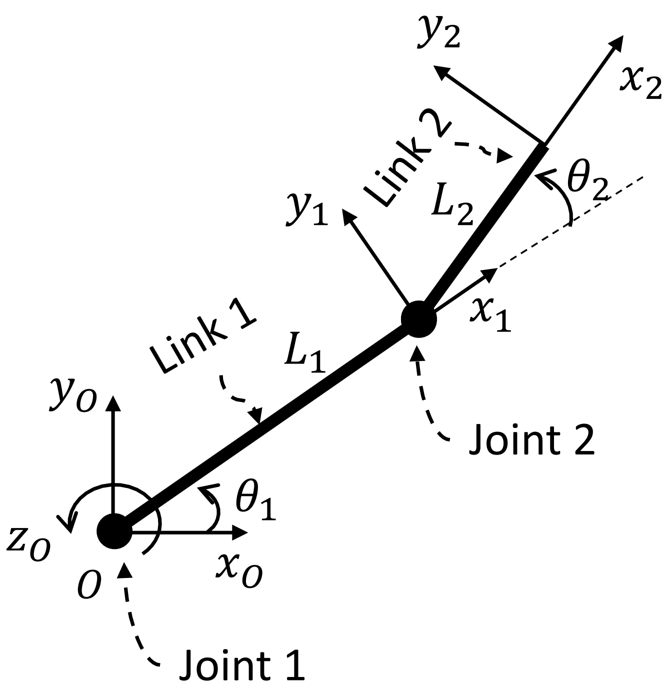
- Method of solution
  - Closed form methods: based on analytic expressions.
  - Numerical methods: based on iterative procedures.
  - We consider two methods: **algebraic** and **Jacobi** method.



Sangsin Park, Ph. D.

Kinematic Equations

- The two-link planar manipulator



1) DH parameters table

i	θ_i	α_i	a_i	d_i	$\cos(\alpha_i)$	$\sin(\alpha_i)$
1	θ_1	0°	L_1	0	1	0
2	θ_2	0°	L_2	0	1	0

2) Link transformation matrices

$$ {}^{O}A_1 = \begin{bmatrix} c_1 & -s_1 & 0 & L_1c_1 \\ s_1 & c_1 & 0 & L_1s_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, {}^{1}A_2 = \begin{bmatrix} c_2 & -s_2 & 0 & L_2c_2 \\ s_2 & c_2 & 0 & L_2s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$*\cos(\theta_i) = c_i$, $\sin(\theta_i) = s_i$, $\cos(\theta_i + \theta_j) = c_{ij}$, $\sin(\theta_i + \theta_j) = s_{ij}$

3) The single transformation matrix that relates frame *2* from frame *O*

$$ {}^{O}T_2 = \begin{bmatrix} c_1c_2 - s_1s_2 & -c_1s_2 - s_1c_2 & 0 & L_1c_1 + L_2c_1c_2 - L_2s_1s_2 \\ s_1c_2 + c_1s_2 & c_1c_2 - s_1s_2 & 0 & L_1s_1 + L_2s_1c_2 + L_2c_1s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c_{12} & -s_{12} & 0 & L_1c_1 + L_2c_{12} \\ s_{12} & c_{12} & 0 & L_1s_1 + L_2s_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

Algebraic Method

- From kinematic equations, ${}^{O}A_1{}^{-1}\,{}^{O}T_2 = {}^{1}A_2$

$$\begin{bmatrix} c_1 & s_1 & 0 & -L_1 \\ -s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c_2 & -s_2 & 0 & L_2c_2 \\ s_2 & c_2 & 0 & L_2s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\begin{cases} p_xc_1 + p_ys_1 = L_2c_2 + L_1 \\ -p_xs_1 + p_yc_1 = L_2s_2 \end{cases}$$

- Square both equations and add them.

$$\begin{cases} p_x^2c_1^2 + p_y^2s_1^2 + 2p_xp_yc_1s_1 = L_2^2c_2^2 + L_1^2 + 2L_1L_2c_2 \\ p_x^2s_1^2 + p_y^2c_1^2 - 2p_xp_yc_1s_1 = L_2^2s_2^2 \end{cases}$$

$$p_x^2 + p_y^2 = L_2^2 + L_1^2 + 2L_1L_2c_2$$

$$\therefore c_2 = \frac{p_x^2 + p_y^2 - L_2^2 - L_1^2}{2L_1L_2}$$

Algebraic Method

- We write an expression for s_2 as

$$s_2 = \pm\sqrt{1 - c_2^2}$$

$$\therefore \theta_2 = atan2(s_2, c_2)$$

  - The choice of s_2 signs corresponds to the multiple solution in which we can choose the 'elbow-down' or the 'elbow-up' solution.

- Remind these equations, $\begin{cases} p_x c_1 + p_y s_1 = L_2 c_2 + L_1 \\ -p_x s_1 + p_y c_1 = L_2 s_2 \end{cases}$

- Expressions for s_1 and c_1 are obtained by solving the equations.

$$\begin{cases} p_x p_y c_1 + p_y^2 s_1 = p_y(L_2 c_2 + L_1) \\ -p_x^2 s_1 + p_x p_y c_1 = p_x L_2 s_2 \end{cases}$$

$$(p_x^2 + p_y^2) s_1 = p_y(L_2 c_2 + L_1) - p_x L_2 s_2$$

$$\therefore s_1 = \frac{p_y(L_2 c_2 + L_1) - p_x L_2 s_2}{p_x^2 + p_y^2}$$

$$\begin{cases} p_x^2 c_1 + p_x p_y s_1 = p_x(L_2 c_2 + L_1) \\ -p_x p_y s_1 + p_y^2 c_1 = p_y L_2 s_2 \end{cases}$$

$$(p_x^2 + p_y^2) c_1 = p_x(L_2 c_2 + L_1) + p_y L_2 s_2$$

$$\therefore c_1 = \frac{p_x(L_2 c_2 + L_1) + p_y L_2 s_2}{p_x^2 + p_y^2}$$

$$\therefore \theta_1 = atan2(s_1, c_1)$$

Jacobi Method

❑ Derive the Jacobian

- From kinematic equations,

$$\begin{cases} x_1 = f_1(\theta_1, \theta_2, \cdots, \theta_n) \\ x_2 = f_2(\theta_1, \theta_2, \cdots, \theta_n) \\ \vdots \\ x_n = f_n(\theta_1, \theta_2, \cdots, \theta_n) \end{cases} \xrightarrow{\text{vector notation}} \boldsymbol{X} = f(\boldsymbol{\theta}),$$

where $\boldsymbol{X}$ is a Cartesian space vector and $\boldsymbol{\theta}$ is a joint space vector.

- Differentiate the kinematic equations.

$$\dot{\boldsymbol{X}} = \frac{\partial f}{\partial \boldsymbol{\theta}} \dot{\boldsymbol{\theta}} = \begin{bmatrix} f_1/\theta_1 & f_1/\theta_2 & \cdots & f_1/\theta_n \\ f_2/\theta_1 & f_2/\theta_2 & \cdots & f_2/\theta_n \\ \vdots & \vdots & \vdots & \vdots \\ f_n/\theta_1 & f_n/\theta_2 & \cdots & f_n/\theta_n \end{bmatrix} \dot{\boldsymbol{\theta}}$$

- Define the matrix of partial derivatives as the Jacobian, $J(\boldsymbol{\theta})$.

$$\therefore \dot{\boldsymbol{X}} = J(\boldsymbol{\theta})\dot{\boldsymbol{\theta}}$$

- If $J(\boldsymbol{\theta})$ is invertible, we can calculate joint velocities given Cartesian velocities.

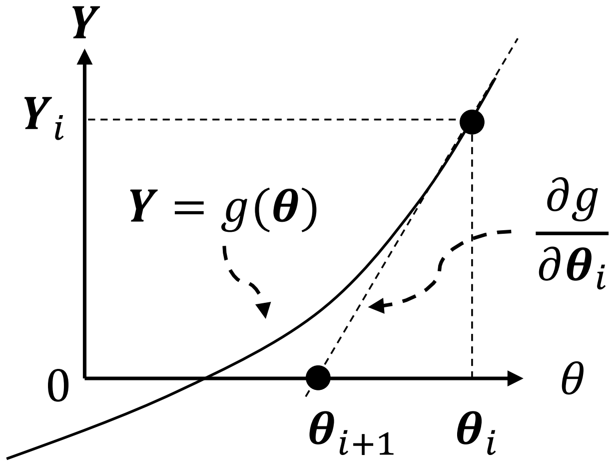
$$\boxed{\dot{\boldsymbol{\theta}} = J(\boldsymbol{\theta})^{-1}\dot{\boldsymbol{X}}}$$

Jacobi Method

- A way to find a solution of inverse kinematics
  - When the desired Cartesian space, $\boldsymbol{X}_d$, the kinematic equation are rewritten as

$$\boldsymbol{X} - \boldsymbol{X}_d = f(\boldsymbol{\theta}) - \boldsymbol{X}_d$$

  - Let $\boldsymbol{X} - \boldsymbol{X}_d = \boldsymbol{Y}$ and $f(\boldsymbol{\theta}) - \boldsymbol{X}_d = g(\boldsymbol{\theta})$. Then $\boldsymbol{Y} = g(\boldsymbol{\theta})$.
  - By ***Newton-Rhapson method***, we can find the solution (i.e. the joint angles) satisfied with $\boldsymbol{X}_d$.



$$\frac{-\boldsymbol{Y}_i}{\boldsymbol{\theta}_{i+1} - \boldsymbol{\theta}_i} = \frac{\partial g}{\partial \boldsymbol{\theta}_i} = \frac{\partial f}{\partial \boldsymbol{\theta}_i}$$

$$\boldsymbol{Y}_i = \frac{\partial f}{\partial \boldsymbol{\theta}_i}(\boldsymbol{\theta}_i - \boldsymbol{\theta}_{i+1})$$

$$\boldsymbol{\theta}_{i+1} = -\frac{\partial f}{\partial \boldsymbol{\theta}_i}^{-1} \boldsymbol{Y}_i + \boldsymbol{\theta}_i = \frac{\partial f}{\partial \boldsymbol{\theta}_i}^{-1} (\boldsymbol{X}_d - \boldsymbol{X}_i) + \boldsymbol{\theta}_i$$

$$\therefore \boldsymbol{\theta}_{i+1} = J(\boldsymbol{\theta}_i)^{-1}(\boldsymbol{X}_d - \boldsymbol{X}_i) + \boldsymbol{\theta}_i$$

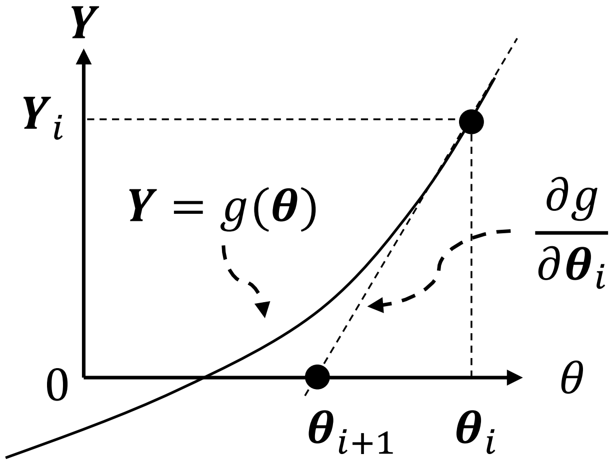
Jacobi Method

❑ A way to find a solution of inverse kinematics

- When the desired Cartesian space, $\boldsymbol{X}_d$, the kinematic equation are rewritten as

$$\boldsymbol{X} - \boldsymbol{X}_d = f(\boldsymbol{\theta}) - \boldsymbol{X}_d$$

- Let $\boldsymbol{X} - \boldsymbol{X}_d = \boldsymbol{Y}$ and $f(\boldsymbol{\theta}) - \boldsymbol{X}_d = g(\boldsymbol{\theta})$. Then $\boldsymbol{Y} = g(\boldsymbol{\theta})$.
- By ***Newton-Rhapson method***, we can find the solution (i.e. the joint angles) satisfied with $\boldsymbol{X}_d$.



- To remember easily, we rewrite the joint rates equation, $\dot{\boldsymbol{\theta}} = J(\boldsymbol{\theta})^{-\mathbf{1}}\dot{\boldsymbol{X}}$ as

$$\frac{\Delta\boldsymbol{\theta}}{\Delta t} = J(\boldsymbol{\theta})^{-1}\frac{\Delta\boldsymbol{\theta}}{\Delta t} \longrightarrow \Delta\boldsymbol{\theta} = J(\boldsymbol{\theta})^{-1}\Delta\boldsymbol{X}$$

$$\boldsymbol{\theta}_{\boldsymbol{i+1}} - \boldsymbol{\theta}_{\boldsymbol{i}} = J(\boldsymbol{\theta}_{\boldsymbol{i}})^{-1}(\boldsymbol{X}_{\boldsymbol{d}} - \boldsymbol{X}_{\boldsymbol{i}})$$

$$\therefore \boldsymbol{\theta}_{i+1} = J(\boldsymbol{\theta}_i)^{-1}(\boldsymbol{X}_d - \boldsymbol{X}_i) + \boldsymbol{\theta}_i$$

Jacobi Method

❑ Iteration flow

- Set initial guess $\boldsymbol{\theta}_0$ and $i = 0$.
- Compute $\boldsymbol{X}_0$ and $\boldsymbol{J}_0^{-1}$.
- Check the error, i.e. $\boldsymbol{X}_d - \boldsymbol{X}_0$ is less than a tolerance.
- If the error is less than the tolerance, terminate the iteration.
- If it isn't, compute $\boldsymbol{\theta}_1 = J_0^{-1}(\boldsymbol{X}_d - \boldsymbol{X}_0) + \boldsymbol{\theta}_0$.
- Increase $i = 1$, and update $\boldsymbol{\theta}_i = \theta_1$.
- Keep iteration until the condition is satisfied.