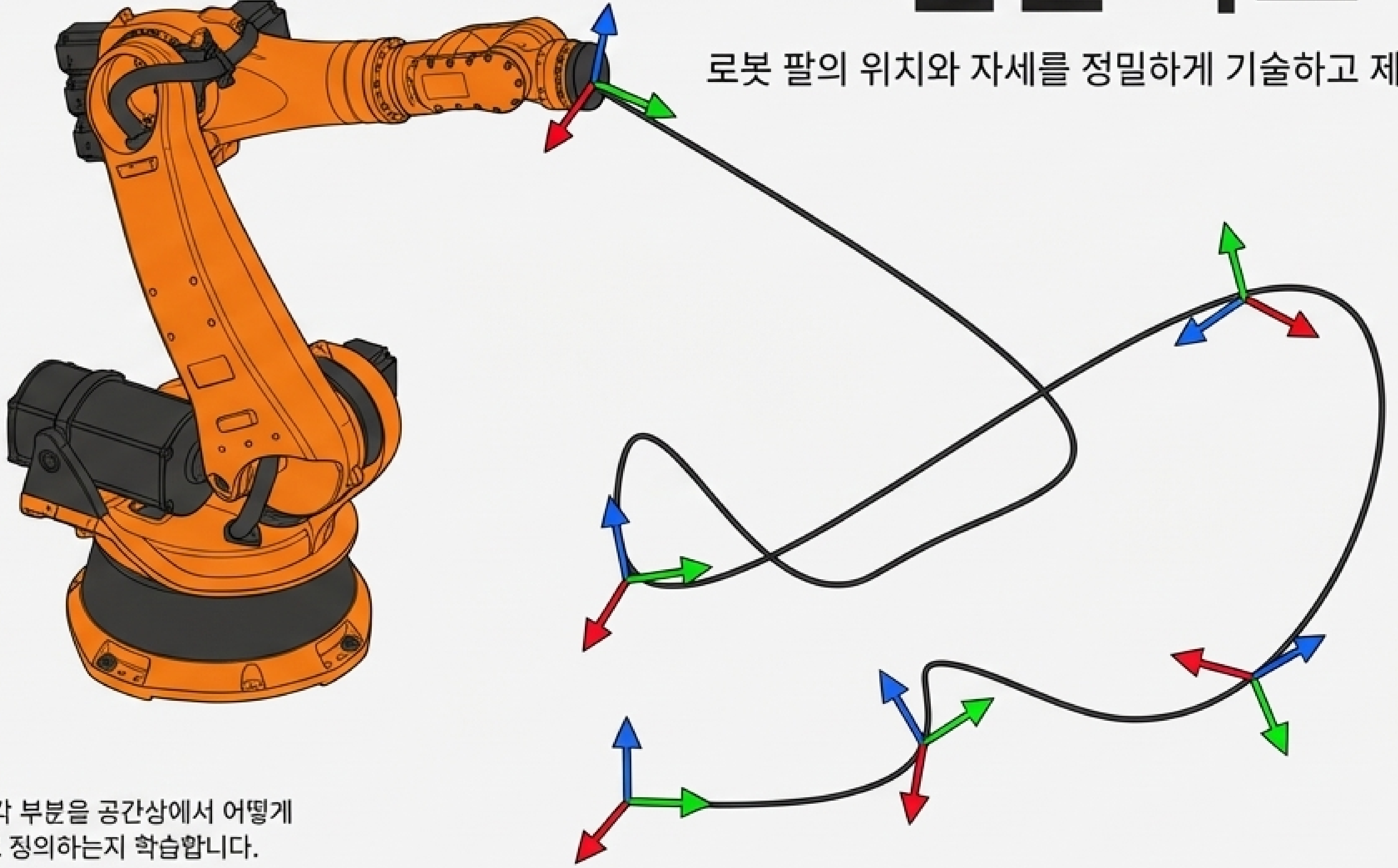
로봇의 언어: 좌표계와 변환 마스터하기

로봇 팔의 위치와 자세를 정밀하게 기술하고 제어하는 수학적 토대



Key Takeaway: 이 장에서는 로봇의 각 부분을 공간상에서 어떻게 표현하고, 그 관계를 어떻게 수학적으로 정의하는지 학습합니다.

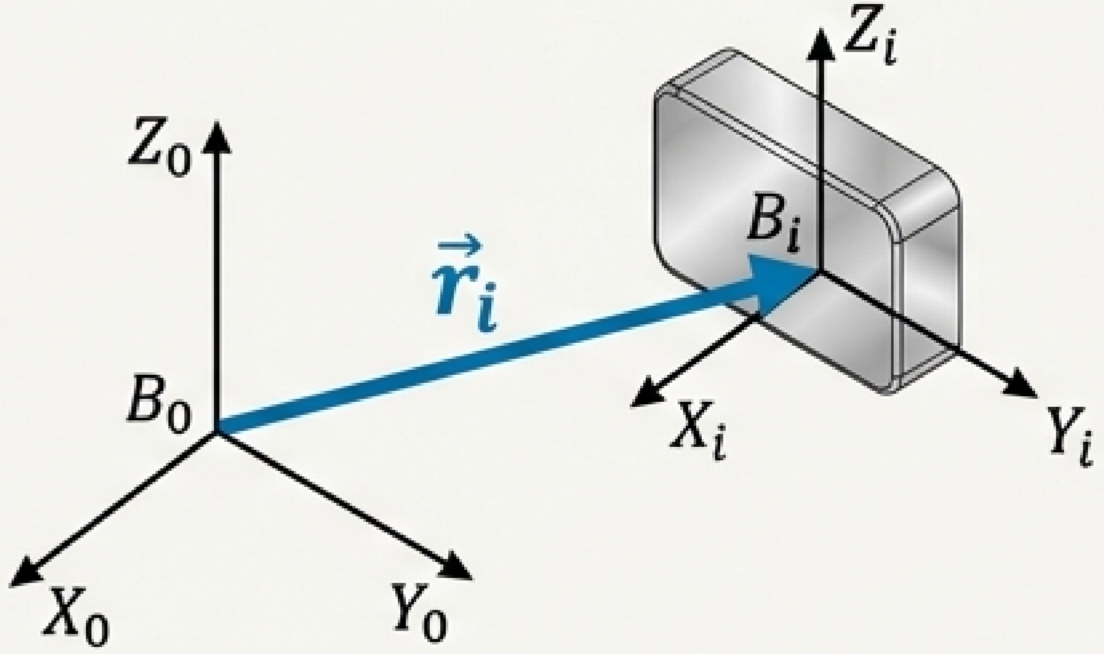
엔지니어의 첫 번째 과제: 공간상 강체(Rigid Body)의 상태 표현하기

로봇의 한 링크(a single link)의 위치(position)와 자세(orientation)를 어떻게 유일하게(uniquely) 정의할 수 있을까?

위치 (Position)

기준 좌표계(B_0)에서 물체의 특정 지점(B_i의 원점)까지의 위치 벡터 $\boldsymbol{r_i}$로 표현합니다.

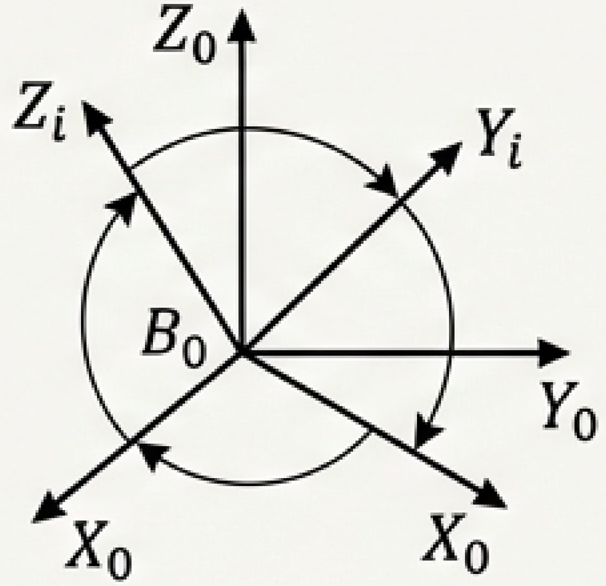
3개의 파라미터가 필요합니다 (예: x, y, z 좌표).



자세 (Orientation)

기준 좌표계(B_0)에 대한 물체 고정 좌표계(B_i)의 상대적인 방향을 회전 행렬(Rotation Matrix) ${}^{i}\boldsymbol{A_0}$로 표현합니다.

3개의 파라미터가 필요합니다.



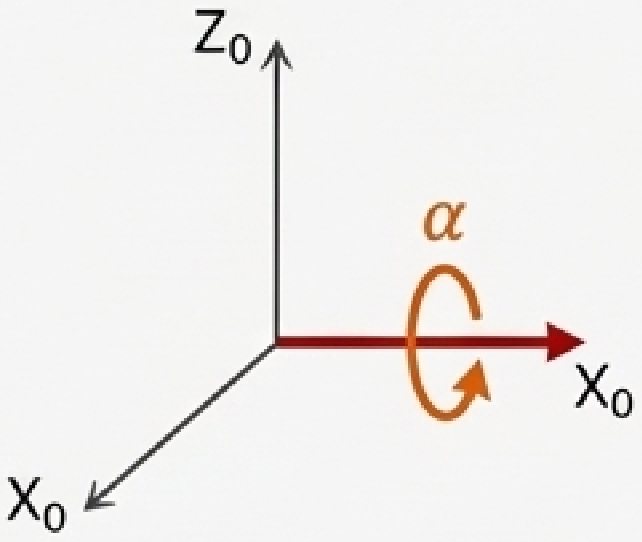
$$
{}^{i}\boldsymbol{A}_0 = \begin{bmatrix} \cos(\theta_1) & -\sin(\theta_1) & 0 \\ \sin(\theta_1) & \cos(\theta_1) & 0 \\ 0 & 0 & 1 \end{bmatrix}
$$

하나의 강체 상태를 완벽하게 기술하기 위해서는 총 6개의 파라미터 (위치 3개, 자세 3개)가 필요합니다.

자세를 표현하는 언어: 회전 행렬

임의의 3차원 회전은 세 번의 기본적인 축(x, y, z) 회전의 조합으로 표현할 수 있습니다. 이를 ‘기본 회전(Elementary)” 이라 합니다. Rotations)'

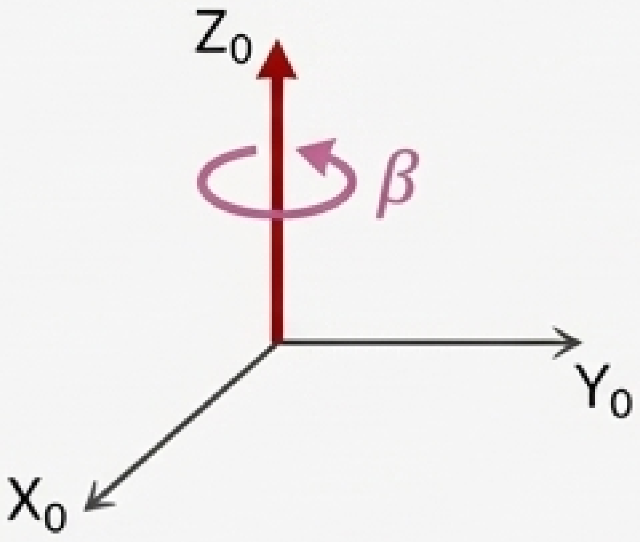
x축 회전 (Rotation about x-axis)



$$A_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix}$$

x축을 중심으로 각도 α만큼 회전시킵니다.

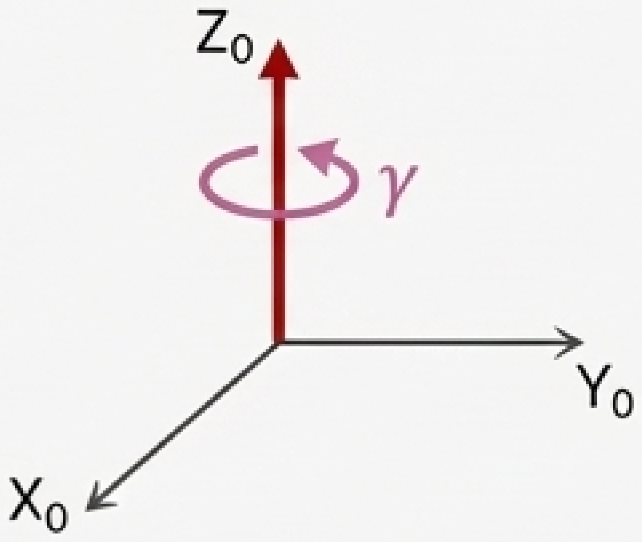
y축 회전 (Rotation about y-axis)



$$A_y(\beta) = \begin{bmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{bmatrix}$$

y축을 중심으로 각도 β만큼 회전시킵니다.

z축 회전 (Rotation about z-axis)



$$A_2(\gamma) = \begin{bmatrix} \cos(\gamma) & -\sin(\gamma) & 0 \\ \sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

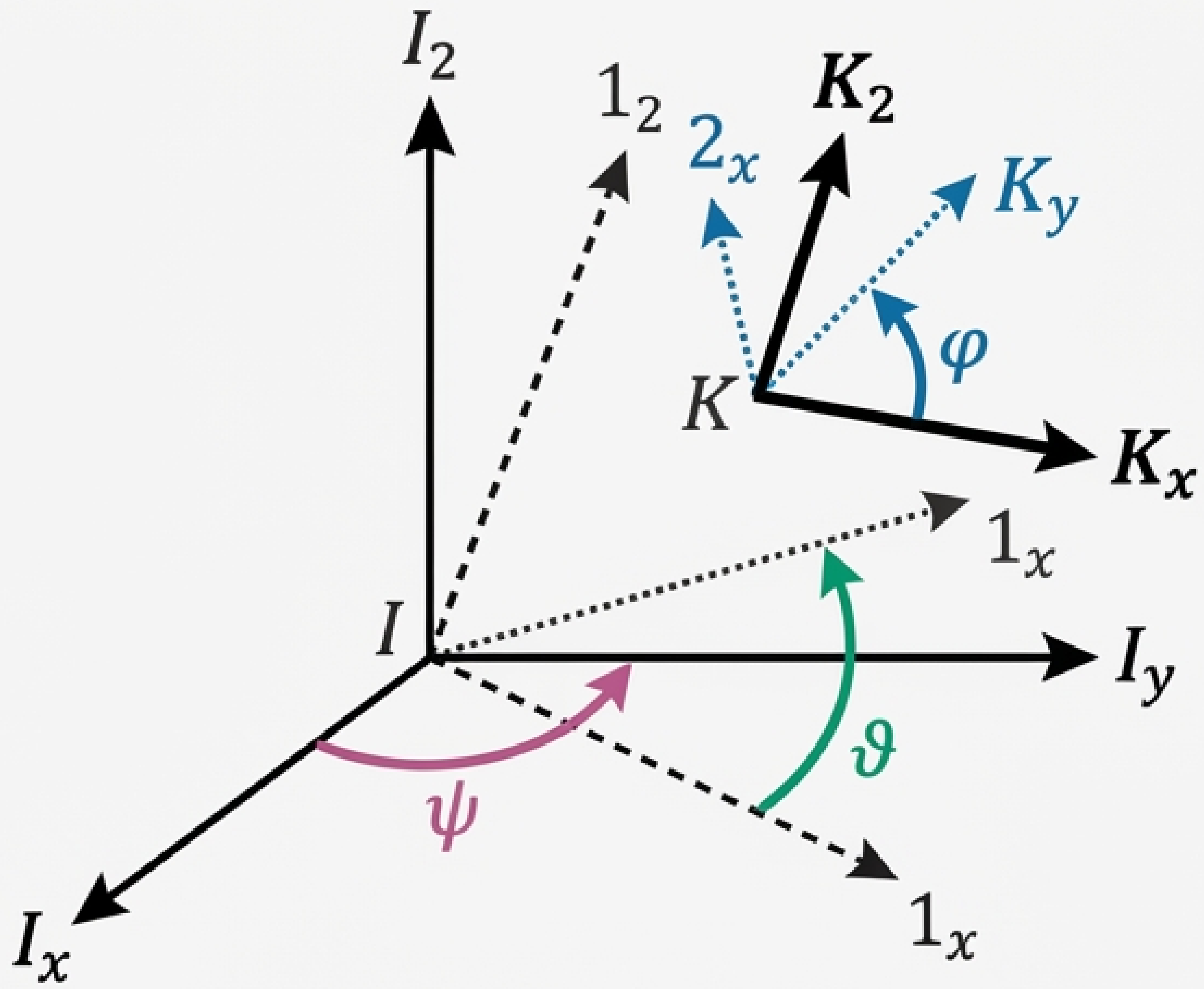
z축을 중심으로 각도 γ만큼 회전시킵니다.

핵심 속성 (Key Property)

회전 행렬 A는 직교 정규 행렬(orthonormal matrix)입니다. 따라서 역행렬은 전치 행렬과 같습니다: $A^{-1} = A^T$. 이 성질은 로봇 기구학 계산에서 매우 유용하게 사용됩니다.

단 3개의 숫자로 모든 3차원 자세 표현하기: 오일러 각

오일러 각은 세 번의 연속적인 기본 회전을 통해 임의의 공간적 자세를 표현하는 방법입니다. 가장 일반적인 규약 중 하나는 Z-X-Z 순서입니다.



Rotation Sequence Explained

1. **1단계:** I_2 축을 중심으로 ψ 만큼 회전.
2. **2단계:** 회전된 1_x 축을 중심으로 ϑ 만큼 회전.
3. **3단계:** 다시 회전된 K_2 축을 중심으로 φ 만큼 회전.

Resulting Rotation Matrix

$$^{K}A_i = A_2(\varphi)A_x(\vartheta)A_2(\psi)$$

$$\begin{bmatrix} c\psi c\varphi - s\psi c\vartheta s\varphi & s\psi c\varphi + c\psi c\vartheta s\varphi & s\vartheta s\varphi \\ -c\psi s\varphi - s\psi c\vartheta c\varphi & -s\psi s\varphi + c\psi c\vartheta c\varphi & s\vartheta c\varphi \\ s\psi s\vartheta & -c\psi s\vartheta & c\vartheta \end{bmatrix}$$

엔지니어의 함정: 특이점(Singularity)을 피하는 법 ⚠️

오일러 각 표현법은 특정 자세에서 자유도를 잃는 '특이점' 문제를 가집니다. $\vartheta = 0$ 일 때, 첫 번째 회전과 세 번째 회전의 축이 정렬되어 두 각도(ψ, φ)를 구분할 수 없게 됩니다.

특이점은 각속도(ω)와 오일러 각의 시간 미분$(\dot{\varphi}, \dot{\vartheta}, \dot{\psi})$ 사이의 관계에서 명확히 드러납니다.

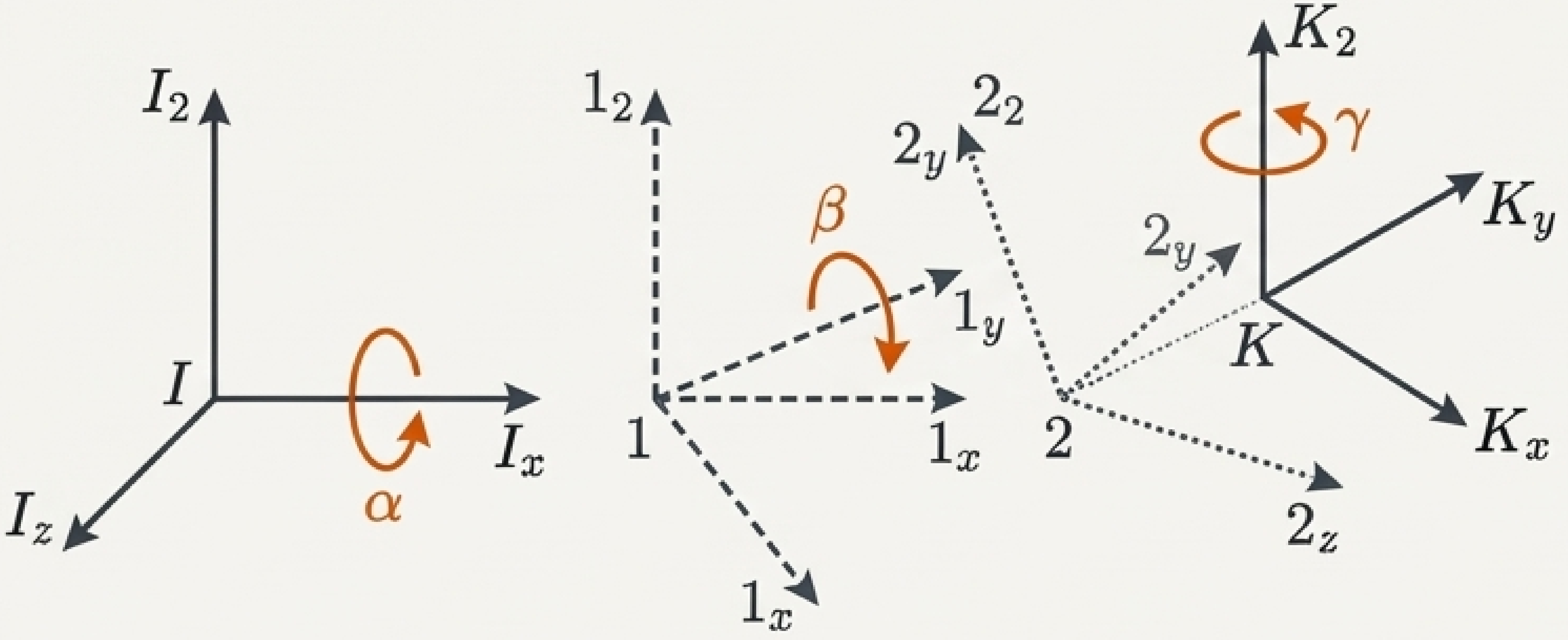
$$\begin{bmatrix} \dot{\varphi} \\ \dot{\vartheta} \\ \dot{\psi} \end{bmatrix} = \frac{1}{s\vartheta} \begin{bmatrix} -s\varphi c\vartheta & -c\varphi c\vartheta & s\vartheta \\ s\vartheta c\varphi & -s\vartheta s\varphi & 0 \\ s\varphi & c\varphi & 0 \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}$$

ϑ가 0에 가까워지면 $\sin(\vartheta)$도 0에 가까워져 각속도 변화율이 무한대로 발산합니다. 이는 물리적으로 불가능하며, 제어 시스템에 심각한 문제를 야기합니다.

실무 적용 (Practical Takeaway)
로봇 제어 시, 특이점 자세는 반드시 피해야 합니다. 이는 다른 자세 표현법을 사용하거나, 작업 경로 계획 시 특이점 영역을 회피함으로써 해결할 수 있습니다.

또 다른 해법: 카르단 각

오일러 각의 특이점 문제를 피하기 위해 사용되는 또 다른 일반적인 방법은 **카르단 각**(혹은 Tait-Bryan 각)입니다. 이는 X-Y-Z와 같이 서로 다른 세 축을 순서대로 회전 시키는 방식입니다.



Rotation Sequence Explained

1. **1단계**: I_x 축을 중심으로 α 만큼 회전.
2. **2단계**: 회전된 1_y 축을 중심으로 β 만큼 회전.
3. **3단계**: 다시 회전된 K_2 축을 중심으로 γ 만큼 회전.

Resulting Rotation Matrix

$$^{K}A_i = A_2(\gamma)A_y(\beta)A_x(\alpha)$$

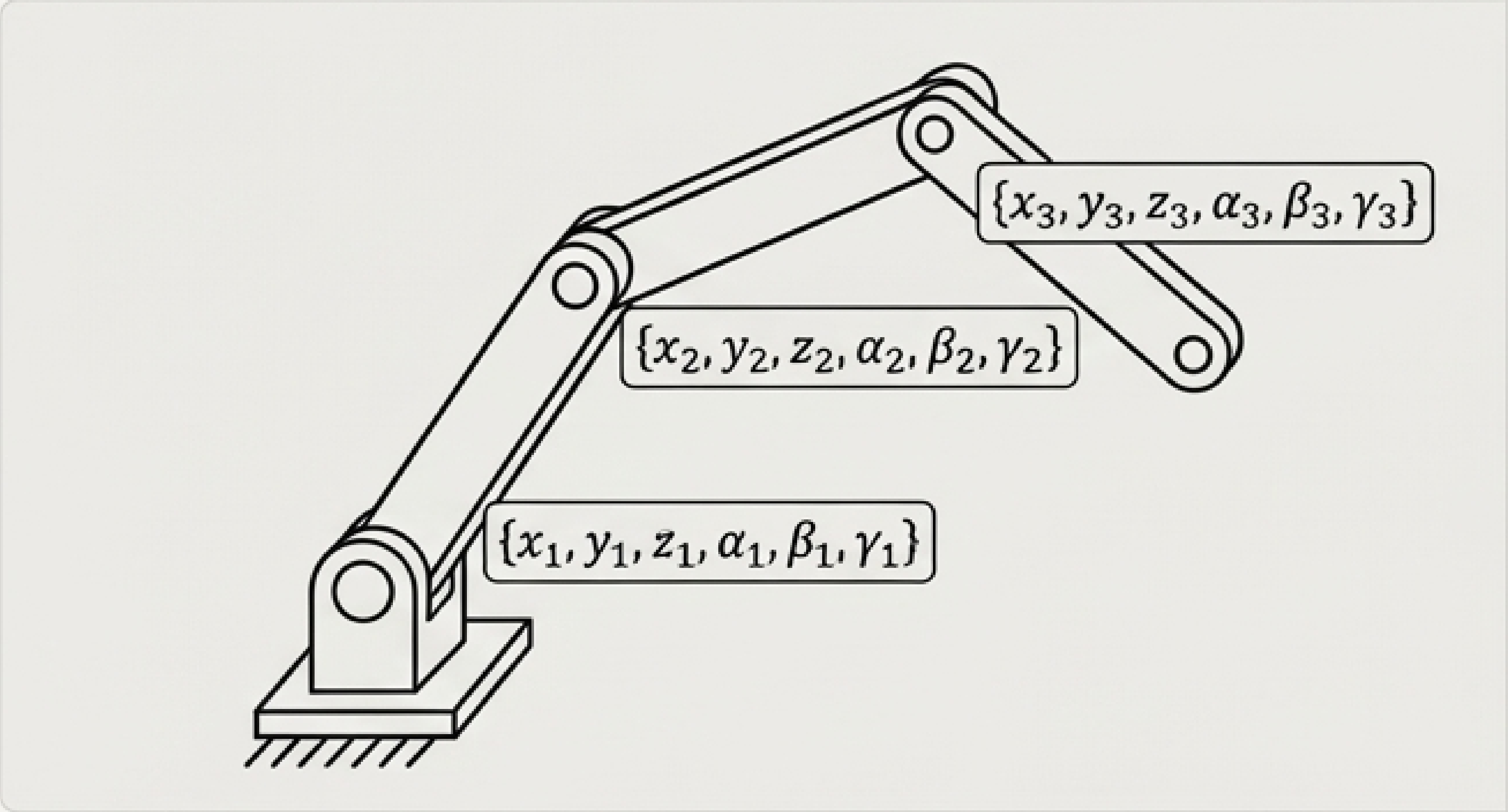
핵심 통찰 (Key Insight)

어떤 자세 표현법을 선택하는가는 어플리케이션과 로봇의 작업 영역에 따라 달라집니다. 각 방법의 장점과 특이점 위치를 이해하는 것이 중요합니다.

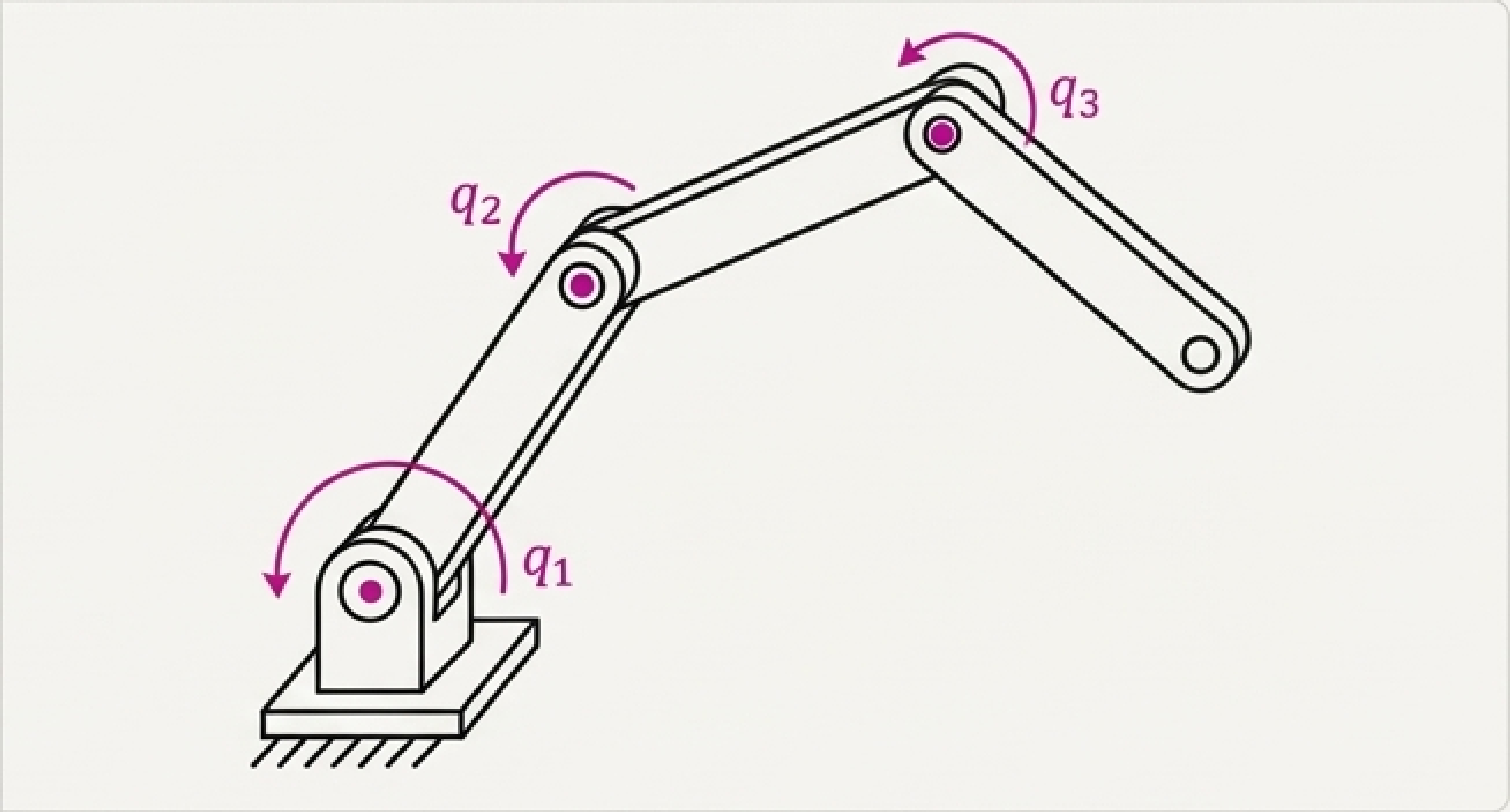
단일 링크에서 전체 시스템으로: 최소 좌표계의 도입

The Scalability Problem

로봇에 N개의 링크가 있다면, 각 링크마다 6개의 파라미터, 즉 총 6N개의 파라미터를 사용해야 할까요? 이 방식은 매우 비효율적이고 중복됩니다. 링크들은 관절(joint)로 연결되어 있어 서로 독립적으로 움직일 수 없기 때문입니다.



Redundant & Complex (6N = 18 parameters)



Efficient & Practical (N = 3 minimal coordinates)

The Solution: 최소 좌표계 (Minimal Coordinates)

- **Definition 2.2.1** 시스템의 기구학적 형상(configuration)을 유일하게(uniquely) 결정하는 데 필요한 최소한의 독립적인 파라미터 집합입니다.
- **In Robotics** 일반적으로 로봇의 관절 각도(revolute joints)나 관절 변위(prismatic joints)가 최소 좌표계 q가 됩니다.

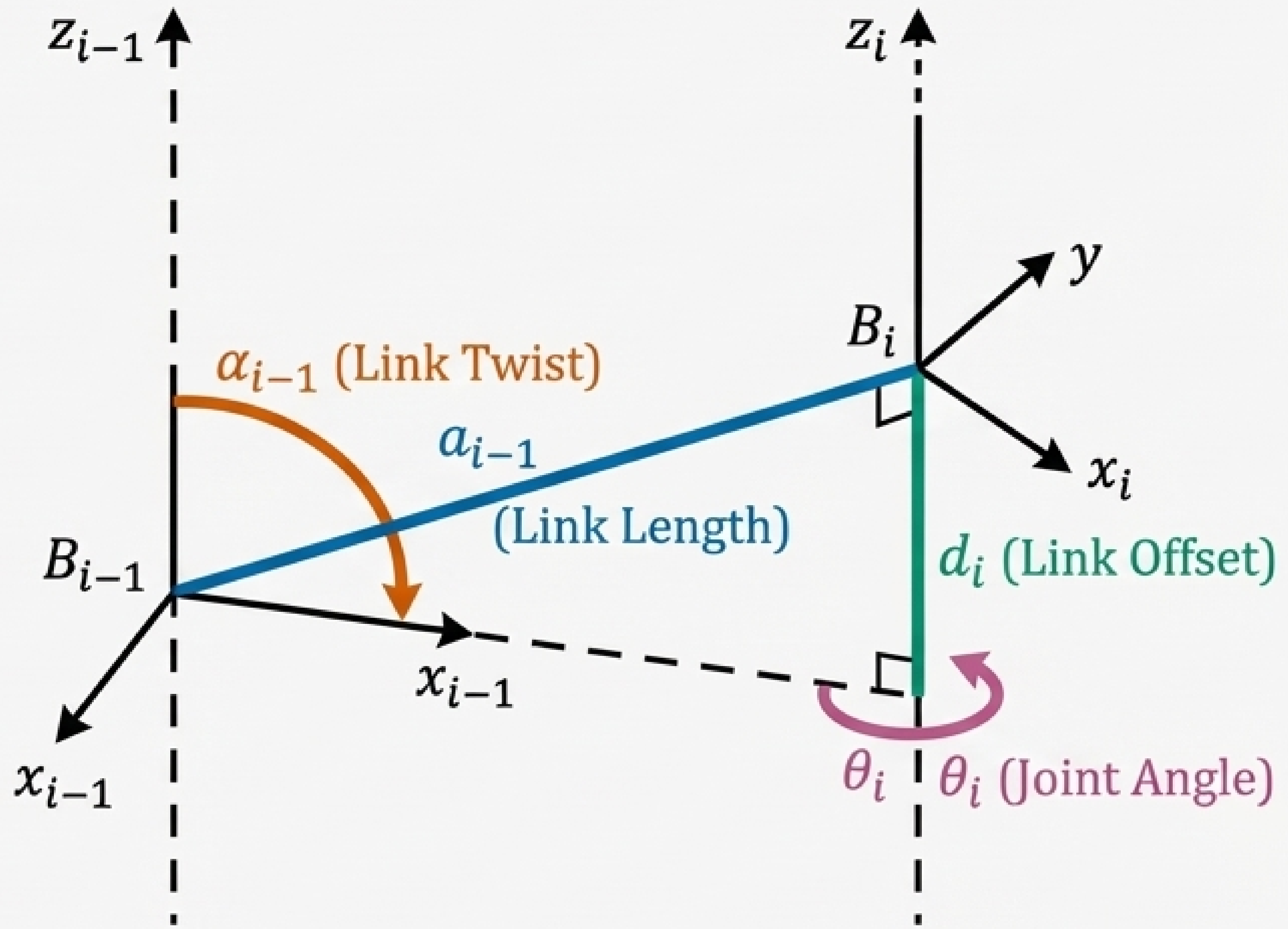
Advantage 이 좌표들은 로봇 제어기에서 직접 측정하고 제어하는 값들이므로 매우 실용적입니다.

로봇 기구학의 표준 언어: 데나빗-하텐버그(DH) 규약

DH 규약은 로봇 공학에서 두 인접한 링크 사이의 상대적 위치와 자세를 기술하기 위해 널리 사용되는 표준화된 방법입니다.

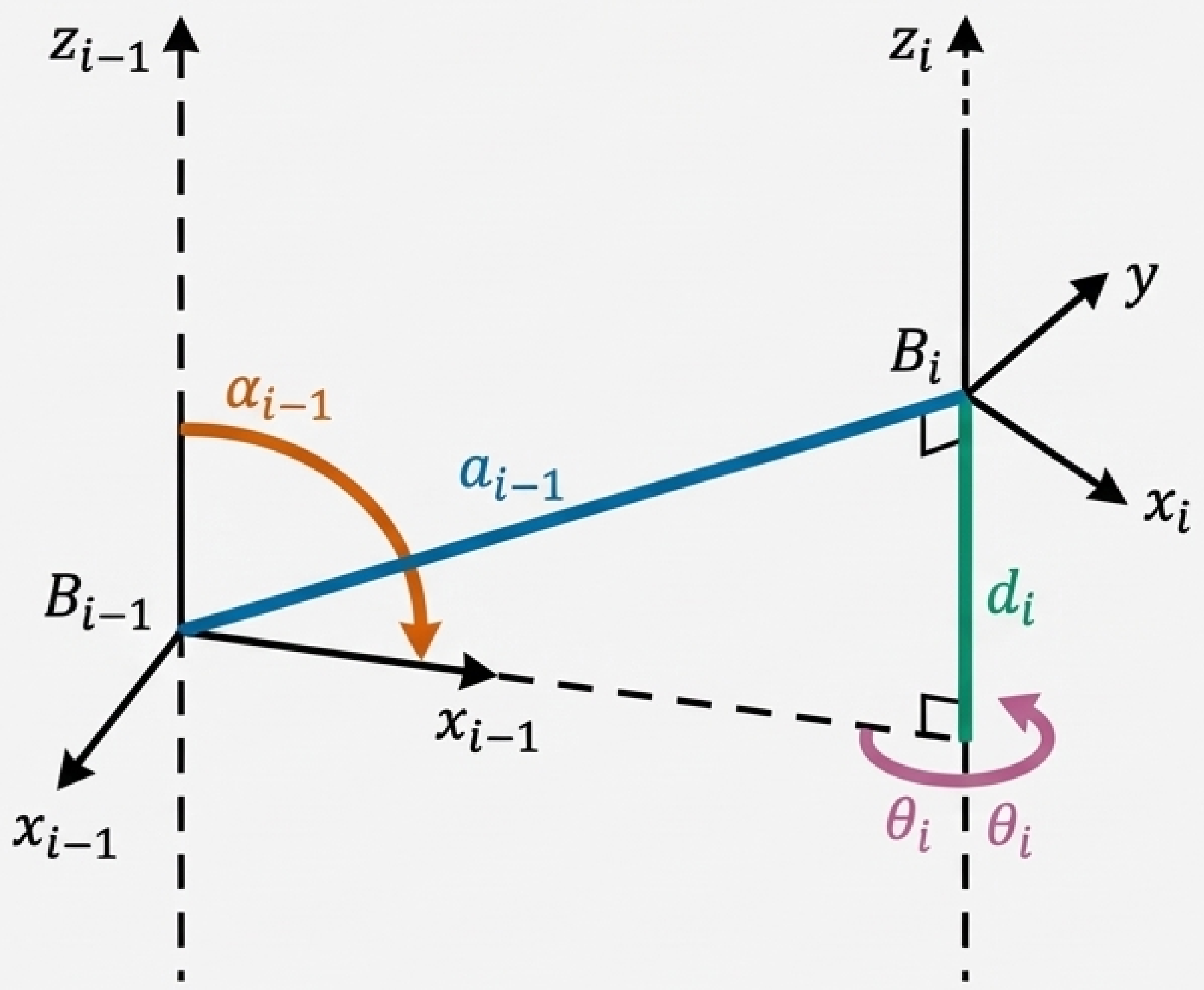
The Core Advantage

- **Problem**
  일반적으로 두 좌표계의 상대적 관계를 기술하려면 6개의 파라미터(이동 3, 회전 3)가 필요합니다.
- **DH Solution**
  좌표계의 위치와 방향 선택에 특정 규칙을 적용함으로써, 단 **4개의 파라미터**만으로 두 링크의 관계를 완벽하게 정의할 수 있습니다.



DH 규약은 복잡한 로봇의 기구학적 구조를 간결하고 체계적인 파라미터 표로 변환해줍니다.

DH 언어의 네 가지 핵심 파라미터



a_{i-1} (Link Length):

- **정의:** z_{i-1} 축과 z_i 축 사이의 공통 수선(common normal)의 길이. 이 거리는 x_{i-1} 축을 따라 측정됩니다.
- **설명:** 두 관절 축 사이의 거리.

α_{i-1} (Link Twist):

- **정의:** x_{i-1} 축을 중심으로 z_{i-1} 축에서 z_i 축까지 회전한 각도.
- **설명:** 두 관절 축 사이의 비틀림 각도.

d_i (Link Offset):

- **정의:** z_i 축을 따라 x_{i-1} 축에서 x_i 축까지의 거리.
- **설명:** 한 링크를 따라 측정된 관절 축의 오프셋.

θ_i (Joint Angle):

- **정의:** z_i 축을 중심으로 x_{i-1} 축에서 x_i 축까지 회전한 각도.
- **설명:** 관절의 회전 각도.

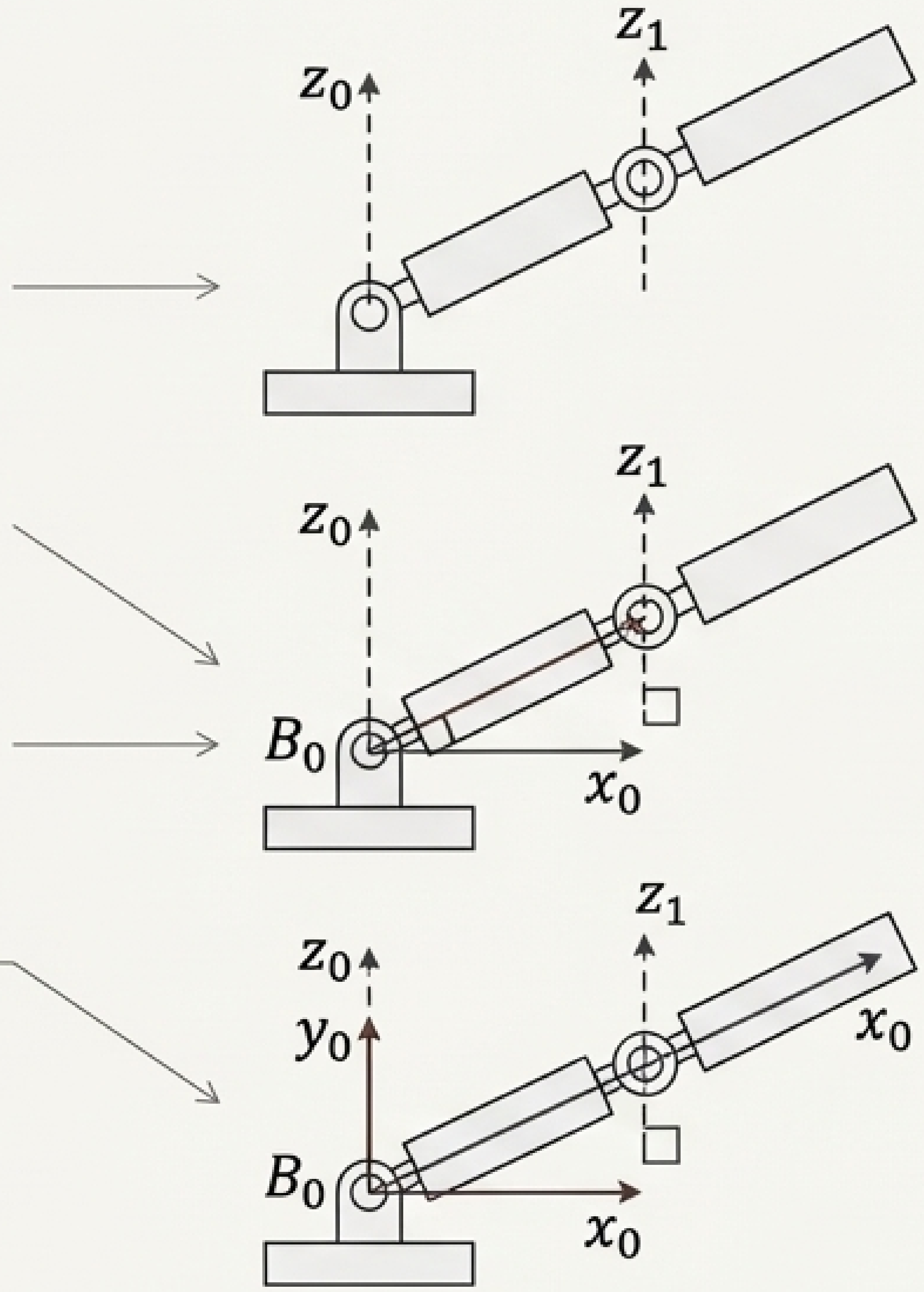
회전 관절(Revolute Joint)에서는 θ_i가 변수이고,
직선 관절(Prismatic Joint)에서는 d_i가 변수입니다.
나머지 세 파라미터는 로봇의 기계적 구조에 의해 고정됩니다.

DH 규약 적용 가이드: 좌표계 설정 규칙

주어진 로봇에 대해 DH 파라미터를 결정하기 위해서는 다음의 체계적인 절차에 따라 각 링크에 고정된 좌표계를 설정해야 합니다.

단계별 가이드 (Step-by-Step Guide)

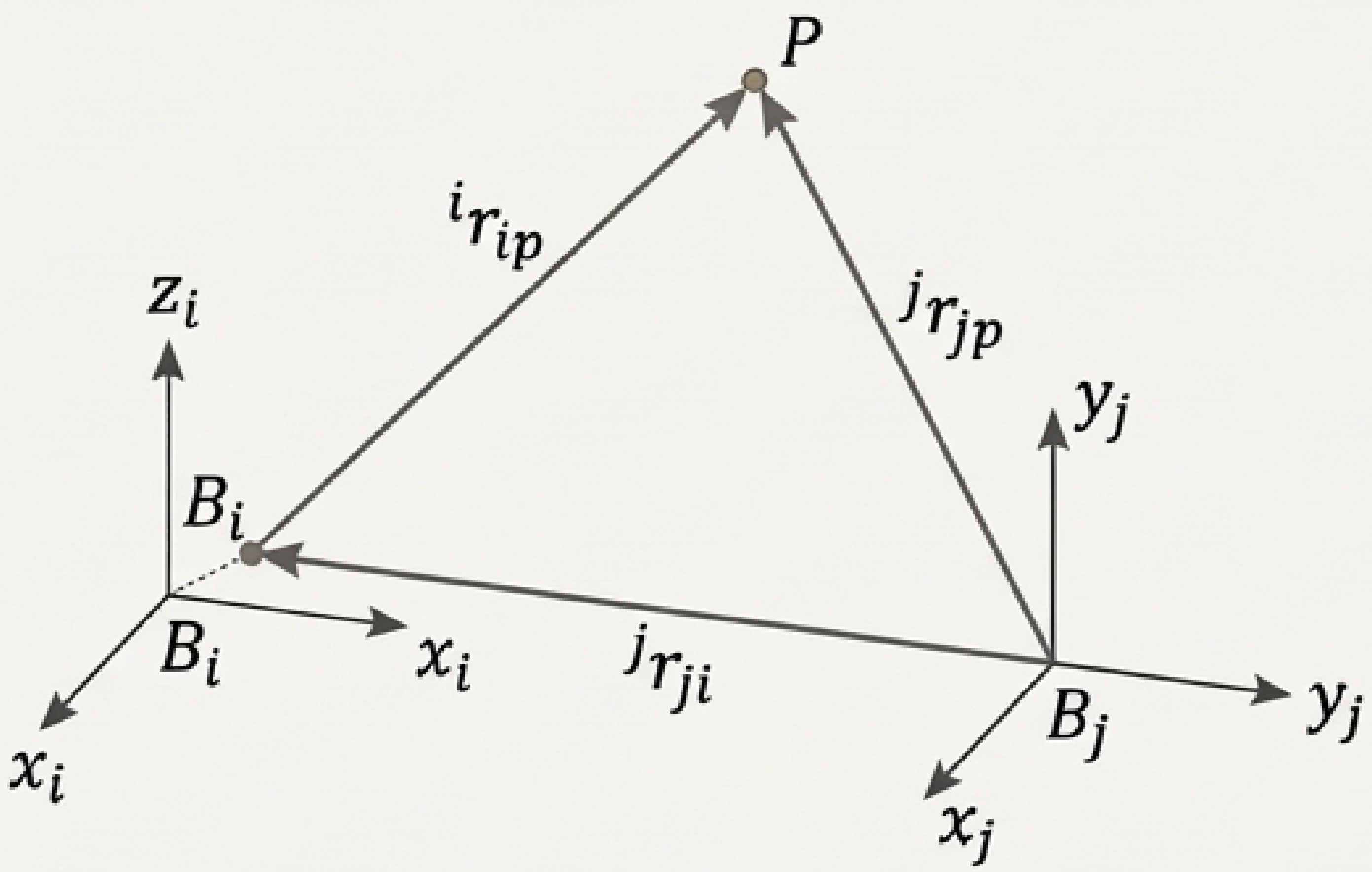
1. **관절 축 정의 (Define Joint Axes)**
   로봇의 각 관절(회전 또는 병진) 축 z_0부터 z_{n-1}까지를 식별하고 번호를 매깁니다. z_{i-1}은 관절 i의 축입니다.
2. **원점 설정 (Set Origins)**
   z_{i-1}과 z_i 축의 공통 수선이 z_{i-1}과 만나는 지점에 $i-1$ 좌표계의 원점을 설정합니다. 두 축이 교차하면 교차점이 원점입니다.
3. **x축 설정 (Set x-axis)**
   x_{i-1} 축은 z_{i-1}에서 z_i를 향하는 공통 수선을 따라 설정합니다.
   두 축이 평행하면 x_{i-1}의 방향은 자유롭게 선택할 수 있습니다.
4. **y축 설정 (Set y-axis)**
   y_{i-1} 축은 오른손 법칙($y = z \times x$)을 만족하도록 결정합니다.
5. **기저 및 끝단 좌표계 (Set Base and End Frames)**
   B_0는 $q_1 = 0$일 때 B_1과 일치하도록 설정합니다.
   B_n은 가능한 많은 DH 파라미터가 0이 되도록 편리하게 선택합니다.



회전과 이동을 하나로: 동차 변환

The Problem

지금까지 위치(벡터 **r**)와 자세(회전 행렬 **A**)를 별도로 다루었습니다. 이 둘을 하나의 행렬 연산으로 통합할 수 없을까요?



$$^{j}r_{jp} = {}^{j}r_{ji} + {}^{j}A_i * {}^{i}r_{ip}$$

The Solution: Homogeneous Transformation

$$
\begin{array}{cc}
 & \begin{array}{cc} \qquad {}^{j}A_i \qquad & \quad {}^{j}r_{ji} \end{array} \\
{}^{j}D_i = & \left(\begin{array}{cc} \begin{bmatrix} a_{11} & a_{122} & a_{13} \\ a_{21} & a_{222} & a_{23} \\ a_{31} & a_{322} & a_{33} \end{bmatrix} & \begin{bmatrix} r_{ii} \\ r_{ji} \\ r_{ji} \end{bmatrix} \\ \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} & \begin{bmatrix} 1 \end{bmatrix} \end{array} \right)
\end{array}
$$

- $^{j}A_i$ (3x3): i에서 j로의 회전 행렬 (Rotation matrix from i to j)
- $^{j}r_{ji}$ (3x1): j 좌표계에서 표현된 원점 i의 위치 벡터 (Position vector of origin i expressed in frame j)
- [0, 1] (1x4): 스케일링 및 투영을 위한 행

$$^{j}z_p = {}^{j}D_i * {}^{i}z_p$$

The Power of Homogeneous Coordinates

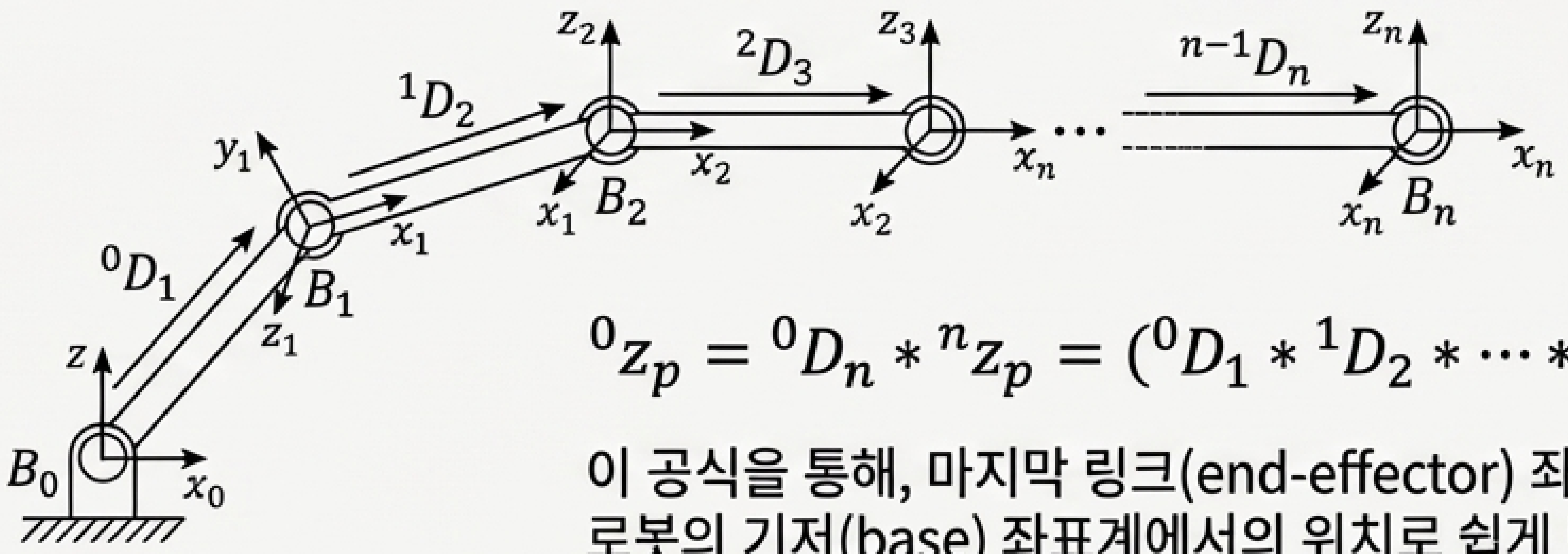
점 P의 좌표($^{i}r_p$)에 1을 추가한 동차 좌표($^{i}z_p$)를 사용하면, 복잡한 벡터 덧셈과 행렬 곱셈($^{j}r_{jp} = {}^{j}r_{ji} + {}^{j}A_i * {}^{i}r_{ip}$)이 단 한 번의 행렬 곱셈으로 단순화됩니다.

DH 파라미터로 동차 변환 행렬 만들기

네 개의 DH 파라미터 (a_{i-1}, α_{i-1}, d_i, θ_i)는 **네 번의 연속적인 기본 변환**(회전 2, 이동 2)으로 해석될 수 있으며, 이들을 곱하여 하나의 **동차 변환 행렬** ${}^{i-1}D_i$ 를 얻습니다.

$$
{}^{i-1}D_i = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} & -s\alpha_{i-1} d_i \\ s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & c\alpha_{i-1} & c\alpha_{i-1} d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

로봇 전체의 변환은 각 링크의 변환 행렬을 순서대로 곱하여 얻습니다.

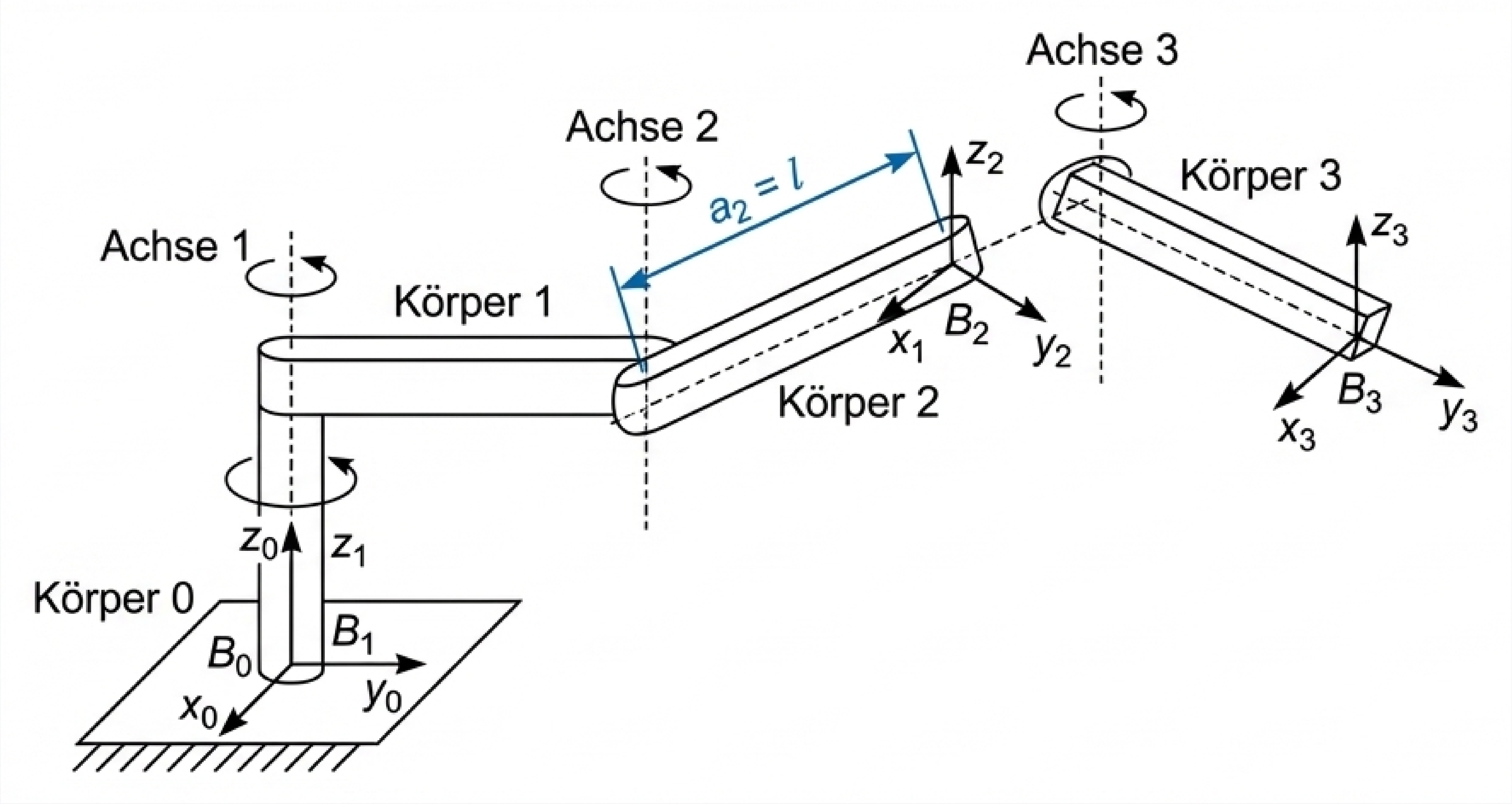


$$
{}^0z_p = {}^0D_n * {}^nz_p = ({}^0D_1 * {}^1D_2 * \cdots * {}^{n-1}D_n) * {}^nz_p
$$

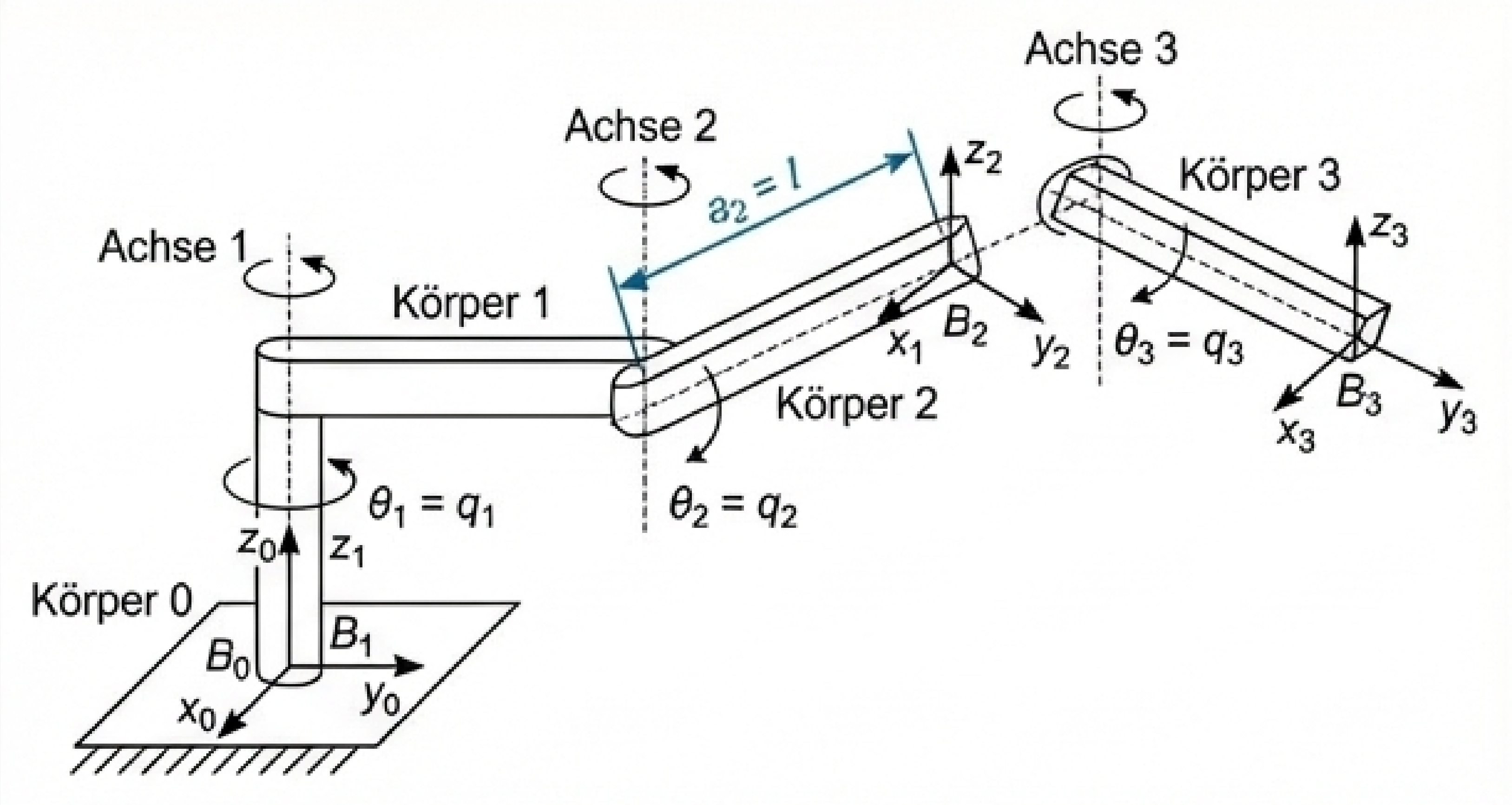
이 공식을 통해, 마지막 링크(end-effector) 좌표계에서 표현된 점의 위치를 로봇의 기저(base) 좌표계에서의 위치로 쉽게 변환할 수 있습니다.

사례 연구: 3-DOF RRR 매니퓰레이터 모델링

Objective 지금까지 배운 DH 규약과 동차 변환을 실제 3-회전관절(RRR) 로봇에 적용하여,
로봇 끝점(end-effector)의 위치와 자세를 관절 각도의 함수로 표현해 보겠습니다.



1단계: RRR 로봇의 DH 파라미터 표 작성하기



i	$\alpha^{(i-1)}$	$a^{(i-1)}$	$d(i)$	$\theta(i)$
1	0	0	0	$\theta_1 = q_1$
2	$-\pi/2$	0	0	$\theta_2 = q_2$
3	0	$a_2 = l$	0	$\theta_3 = q_3$

- For i=1: z_0와 z_1은 평행하고(α_0=0), x_0 축상에서 만나므로(a_0=0), x_0과 x_1 사이의 오프셋도 없습니다(d_1=0). 변수는 θ_1입니다.
- For i=2: z_1에서 z_2로 가려면 x_1 축을 중심으로 −90°($-\pi/2$) 회전해야 합니다(α_1=$-\pi/2$). 나머지 파라미터는 0입니다.
- For i=3: z_2와 z_3은 평행합니다(α_2=0). z_2에서 z_3까지의 거리는 x_2 축을 따라 l입니다(a_2=l). 오프셋은 없습니다(d_3=0).

2단계: 최종 변환 행렬 및 끝점 위치 계산

각 링크의 DH 파라미터를 동차 변환 행렬 공식에 대입한 후, 이들을 모두 곱하여 기저 좌표계 B_0에서 끝점 좌표계 B_3 까지의 전체 변환 행렬 ${}^0D_3$을 구합니다. ${}^0D_3 = \theta = {}^0D_1 * {}^1D_2 * {}^2D_3$

$$
{}^0D_3 = \begin{bmatrix} cq_1 * cq_{23} & -cq_1 * sq_{23} & -sq_1 & cq_1 l * cq_2 \\ sq_1 * cq_{23} & -sq_1 * sq_{23} & cq_1 & sq_1 l * cq_2 \\ -sq_{23} & -cq_{23} & 0 & -l * sq_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

이 ${}^0D_3$ 행렬은 로봇의 **순기구학(Forward Kinematics)** 해답입니다.

회전 부분 (Rotation Part)
좌상단 3x3 부분(${}^0A_3$)은 끝점의 자세를 나타냅니다.

위치 부분 (Position Part)
우측 상단 3x1 벡터(${}^0r_{03}$)는 끝점의 위치를 나타냅니다.

Engineering Goal Achieved: 이제 우리는 세 개의 관절 각도(q_1, q_2, q_3)만 알면 로봇 끝점의 3차원 공간상 위치와 자세를 정확히 계산할 수 있는 수학적 모델을 완성했습니다. 이것이 로봇 제어의 첫걸음입니다.

$$
{}^0r_{03} = \left[l * cq_1 * cq_2,\ \ l * sq_1 * cq_2,\ \ -l * sq_2 \right]^T
$$