



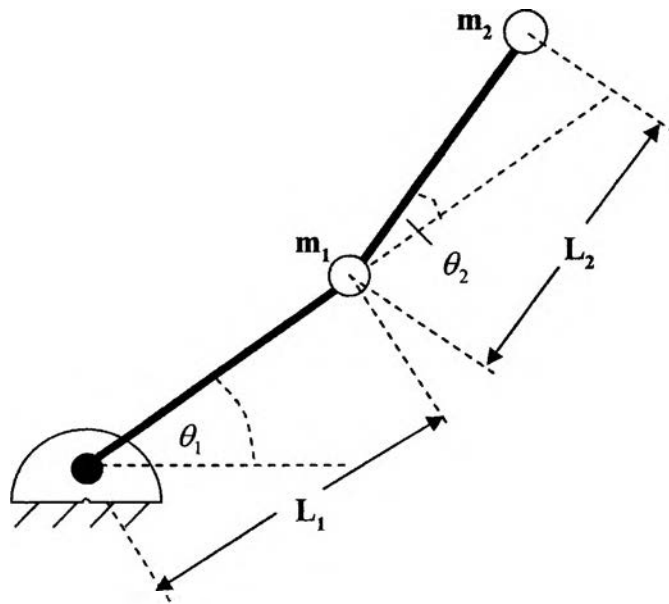
## บทที่ 4

### การจำลองควบคุม

#### 4.1 นำเรื่อง

ในการจำลองควบคุมหุ่นยนต์อุตสาหกรรมสามารถนำแบบจำลองทางคณิตศาสตร์มาใช้อธิบายรูปแบบโครงสร้างและการเคลื่อนไหวลักษณะต่างๆ เพื่อนำสู่การหาตำแหน่งของปลายแขนกลในรูปแบบของจลนศาสตร์ไปข้างหน้า (Forward Kinematics) การหาตำแหน่งของข้อต่อ (Joint) จากจลนศาสตร์ย้อนกลับ (Inverse Kinematics) หรือหาความเร็วและแรงกระทำที่ข้อต่อหรือปลายแขนกลได้จากจาโคเบียนความเร็วและแรง (Velocity and Force Jacobian) และจลนศาสตร์ไปข้างหน้าและย้อนกลับ (Forward and Inverse Dynamics) เพื่อใช้เป็นข้อมูลประกอบการควบคุมหุ่นยนต์อุตสาหกรรม โดยกล่าวถึงการจำลองควบคุมหุ่นยนต์ 2-Link Planar Arm และ Articulated Robot บนเส้นทางเดินแบบวงกลมและแบบสี่เหลี่ยมเพื่อหลบหลีกสิ่งกีดขวาง โดยใช้การประมาณจาโคเบียนขณะที่หุ่นยนต์เคลื่อนที่ การปรับแต่งเส้นทางเดินใหม่ด้วยระเบียบวิธี Gradient Projection

#### 4.2 การจำลองควบคุมแขนกล 2-Link Planar Arm เพื่อหลบหลีกสิ่งกีดขวาง



รูปที่ 4.1 Kinematics Model ของแขนกลแบบ 2-Link Planar Arm

4.2.1 พารามิเตอร์ของหุ่นยนต์ (Robot Parameter) สามารถแสดงดังรูปที่ 4.1 กำหนดให้

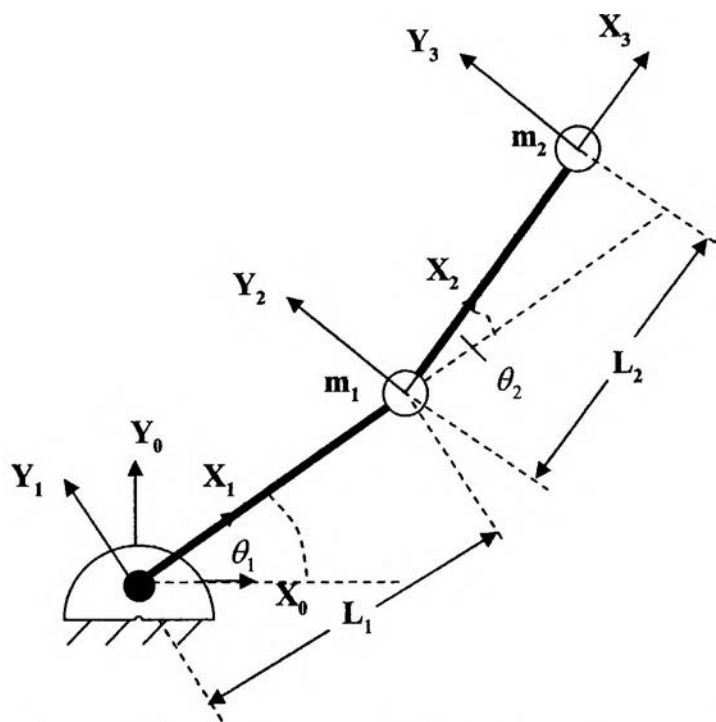
ความยาวของ Link  $a_1 = a_2 = 1 \text{ m.}$

ระยะทางจุดศูนย์กลางมวลจากจุดหมุน  $l_1 = l_2 = 0.5 \text{ m.}$

มวลของ Link  $m_{11} = m_{12} = 1 \text{ kg.}$

#### 4.2.2 ดี-เอชพารามิเตอร์ (D-H parameter)

วิธี D-H parameter ช่วยในการหาจลนศาสตร์ไปข้างหน้า (Forward Kinematics) โดยจะตั้งเฟรมอ้างอิงที่แต่ละข้อต่อ แล้วหาค่าความสัมพันธ์ระหว่างแกน และข้อต่อ ดังแสดงในรูปที่ 4.2 และค่าตัวแปรที่เกี่ยวข้องกับ D-H parameter แสดงในตารางที่ 4.1



รูปที่ 4.2 การตั้งแกนอ้างอิงของแขนกลแบบ 2-Link Planar Arm

ตารางที่ 4.1 D-H parameter ของ 2-Link Planar Arm

Link No.(i)	Link Twist ( $\alpha_{i-1}$ )	Link Length( $a_{i-1}$ )	Link Offset (d <sub>i</sub> )	Joint Angle ( $\theta_i$ )
1	0	0	0	$\theta_1$
2	0	$L_1$	0	$\theta_2$
3	0	$L_2$	0	0

โดยที่  $\alpha_{i-1}, a_{i-1}$  เป็นค่าคงที่  
 $\theta_i, d_i$  เป็นตัวแปรตามขึ้นอยู่กับเคลื่อนที่ว่าเป็นข้อต่อแบบเลื่อน (Prismatic Joint) หรือข้อต่อแบบหมุน (Revolute Joint)

โดยวิธีของ Denavit-Hartenberg ซึ่งในปี ค.ศ.1955 R. S. Hartenberg และ J. Denavit [19] ได้เสนอ วิธี D-H parameter จากรูปทั่วไปของ Homogeneous Transformation Matrix ดังนี้

$${}^{i-1}T_i = \begin{bmatrix} \cos \theta_i & -\sin \theta_i & 0 & a_{i-1} \\ \sin \theta_i \cos \alpha_{i-1} & \cos \theta_i \cos \alpha_{i-1} & -\sin \alpha_{i-1} & -d_i \sin \alpha_{i-1} \\ \sin \theta_i \sin \alpha_{i-1} & \cos \theta_i \sin \alpha_{i-1} & \cos \alpha_{i-1} & d_i \cos \alpha_{i-1} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

#### 4.2.3 จลนศาสตร์ไปข้างหน้าและย้อนกลับ (Forward and Inverse Kinematics)

##### จลนศาสตร์ไปข้างหน้า (Forward Kinematics)

เป็นการโอนย้ายความสัมพันธ์ของตำแหน่งแขนกลจากการอ้างอิงเชิงมุมของแต่ละ Joint ในแกนอ้างอิงแบบตัวแปรข้อต่อ (Joint Space) มาเป็นการอ้างอิงเชิงเส้น 3 มิติแกน X, Y, Z (Cartesian Space) เพื่อบอกตำแหน่งปลายแขนกลเทียบกับเฟรมอ้างอิงเริ่มต้น (Base Frame) แทนด้วยความสัมพันธ์ดังนี้

$${}_{3}^{0}T = \begin{bmatrix} \cos(\theta_1 + \theta_2) & -\sin(\theta_1 + \theta_2) & 0 & \cos \theta_1 L_1 + \cos(\theta_1 + \theta_2) L_2 \\ \sin(\theta_1 + \theta_2) & \cos(\theta_1 + \theta_2) & 0 & \sin \theta_1 L_1 + \sin(\theta_1 + \theta_2) L_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}_{3}^{0}T = \begin{bmatrix} \ddots & \ddots & P_x \\ & R & P_y \\ \ddots & \ddots & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

พิจารณา  ${}_{3}^{0}T$  พบว่า

[R] = Rotation Matrix ของ 2 Link Planar Arm

[P] = Position Vector บอกตำแหน่งของปลายแขนกลเชิงเส้นในแนวแกน x, y, z เทียบกับฐานของหุ่นยนต์

## จลนศาสตร์ย้อนกลับ (Inverse Kinematics)

เป็นการโอนย้ายความสัมพันธ์ย้อนกลับของตำแหน่งแขนกล จากการอ้างอิงเชิงเส้น 3 มิติ แกน  $x, y, z$  ใน Cartesian Space ของปลายแขนกล มาเป็นการอ้างอิงในแกนอ้างอิงแบบตัวแปร ข้อต่อ เพื่อบอกตำแหน่งปลายแขนกลเทียบกับเฟรมอ้างอิงเริ่มต้น ซึ่งจะพบว่าสามารถหาผลเฉลยได้หลายคำตอบขึ้นอยู่กับเงื่อนไขในการทำงานของแขนกล เช่น สิ่งกีดขวาง จากการศึกษาสามารถหาผลเฉลยได้หลายวิธี เช่น Algebraic, Geometric ซึ่งในที่นี้กล่าวถึงแบบ Geometric

### วิธีหาผลเฉลยแบบ Geometric

#### Joint 1

$\therefore \theta_1 = A \tan 2(y, x) - A \tan 2(k_2, k_1)$  (ต้องพิจารณาเงื่อนไขประกอบเนื่องจากคำตอบมี 2 ค่า) โดยที่

$$x = k_1 \cos \theta_1 - k_2 \sin \theta_1$$

$$y = k_1 \sin \theta_1 + k_2 \cos \theta_1$$

$$k_1 = L_1 + L_2 \cos \theta_2$$

$$k_2 = L_2 \sin \theta_2$$

#### Joint 2

$$\therefore \theta_2 = A \tan 2(2L_1L_2\sqrt{1 - (\cos \theta_2)^2}, x^2 + y^2 - L_1^2 - L_2^2)$$

(ต้องพิจารณาเงื่อนไขประกอบเนื่องจากคำตอบมี 2 ค่า)

### 4.2.4 จาโคเบียน (Jacobian)

การที่หุ่นยนต์มีการเคลื่อนที่ปฏิสัมพันธ์กับสิ่งแวดล้อม คุณลักษณะต่างๆจะเปลี่ยนแปลงอย่างต่อเนื่องตลอดการเคลื่อนไหว ทำให้เกิดการเปลี่ยนแปลงของระยะทาง แรงในแนวเชิงเส้น เชิงมุมเมื่อเทียบกับแกนอ้างอิง จึงได้ทำการหาวิธีการโอนย้ายความสัมพันธ์ความเร็ว แรงของตำแหน่งต่างๆบนแขนกล เช่น ความเร็วเชิงมุม ในกรณีที่เป็นข้อต่อแบบหมุน (Revolute Joint) และความเร็วเชิงเส้น ในกรณีที่เป็นข้อต่อแบบเลื่อน (Prismatic Joint) หรือปลายแขนกล

#### จาโคเบียนความเร็ว (Velocity Jacobian)

เป็นโอนย้ายความสัมพันธ์ความเร็วของตำแหน่งต่างๆบนแขนกล เช่น ความเร็วเชิงมุม (ในกรณีที่เป็นข้อต่อแบบหมุน) หรือความเร็วเชิงเส้น (ในกรณีที่เป็นข้อต่อแบบเลื่อนหรือปลายแขน) ระหว่างการอ้างอิงที่ฐานของหุ่นยนต์และการอ้างอิงที่ปลายแขนกล

$${}^0v_3 = \begin{bmatrix} -L_1 \sin \theta_1 - L_2 \sin(\theta_1 + \theta_2) & -L_2 \sin(\theta_1 + \theta_2) \\ L_1 \cos \theta_1 + L_2 \cos(\theta_1 + \theta_2) & L_2 \cos(\theta_1 + \theta_2) \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix}$$

$$\therefore {}^0J(\theta) = \begin{bmatrix} -L_1 \sin \theta_1 - L_2 \sin(\theta_1 + \theta_2) & -L_2 \sin(\theta_1 + \theta_2) \\ L_1 \cos \theta_1 + L_2 \cos(\theta_1 + \theta_2) & L_2 \cos(\theta_1 + \theta_2) \end{bmatrix}$$

#### จาโคเบียนแรง (Force Jacobian)

เป็นอินทรีย์ความสัมพันธ์แรงของตำแหน่งต่างๆบนแกนกล เช่น แรงบิด (ในกรณีที่เป็นข้อต่อแบบหมุน) หรือแรงเชิงเส้น (ในกรณีที่เป็นข้อต่อแบบเลื่อนหรือปลายแกน) ระหว่างการอ้างอิงที่ฐานของหุ่นยนต์และการอ้างอิงที่ปลายแกนกล

$$\therefore \tau = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = \begin{bmatrix} L_1 \sin \theta_1 & L_2 + L_1 \cos \theta_2 \\ 0 & L_2 \end{bmatrix} \begin{bmatrix} f_x \\ f_y \end{bmatrix}$$

#### 4.2.5 พลศาสตร์ไปข้างหน้าและย้อนกลับ (Forward and Inverse Dynamics)

พลศาสตร์ของแกนกล เป็นความสัมพันธ์ของแรงบิด (Joint Torque) ที่กระทำที่ข้อต่อกับการเคลื่อนที่ของแกนกลใน Cartesian Space โดยสามารถคำนวณหาได้ 2 วิธี คือ วิธี Lagrangian Formulation และวิธี Recursive Newton-Euler Formulation ซึ่งในที่นี้ได้ทำการคำนวณด้วยวิธี Recursive Newton-Euler Formulation (สามารถศึกษารายละเอียดได้ในภาคผนวก ก)

#### พลศาสตร์ไปข้างหน้า (Forward Dynamics)

เป็นการคำนวณหาแรงบิด (Joint Torque) ที่ต้องการเพื่อใช้ในการขับเคลื่อนหุ่นยนต์ให้เคลื่อนที่ไปยังตำแหน่งที่ต้องการ

$$\tau_1 = \begin{bmatrix} \left( \frac{1}{3} m_1 l^2 + \frac{4}{3} m_2 l^2 + m_2 \cos^2 \theta_2 l^2 \right) \ddot{\theta}_1 + \left( \frac{1}{3} m_1 l^2 + \frac{1}{2} m_2 \cos \theta_2 l^2 \right) \ddot{\theta}_2 \\ - m_2 \sin \theta_2 l^2 \dot{\theta}_1 \dot{\theta}_2 - \frac{1}{2} m_2 \sin \theta_2 l^2 \dot{\theta}_2^2 \\ + \frac{1}{2} m_1 g l \cos \theta_1 + \frac{1}{2} m_2 g l \cos(\theta_1 + \theta_2) + m_2 g l \cos(\theta_1 + \theta_2) \end{bmatrix}$$

$$\tau_2 = \left( \frac{1}{3} m_2 l^2 + \frac{1}{2} m_2 \cos \theta_1 l^2 \right) \ddot{\theta}_1 + \left( \frac{1}{3} m_2 l^2 \right) \ddot{\theta}_2 + \frac{1}{2} m_2 \sin \theta_2 l^2 \dot{\theta}_1^2 + \frac{1}{2} m_2 g l \cos(\theta_1 + \theta_2)$$

### พลศาสตร์ย้อนกลับ (Inverse Dynamics)

เป็นการคำนวณหาความเร่งที่จุดศูนย์กลางมวลของแขน (Link) ที่เกิดขึ้นจากการขับเคลื่อนหุ่นยนต์ให้เคลื่อนที่ไปยังตำแหน่งที่ต้องการ

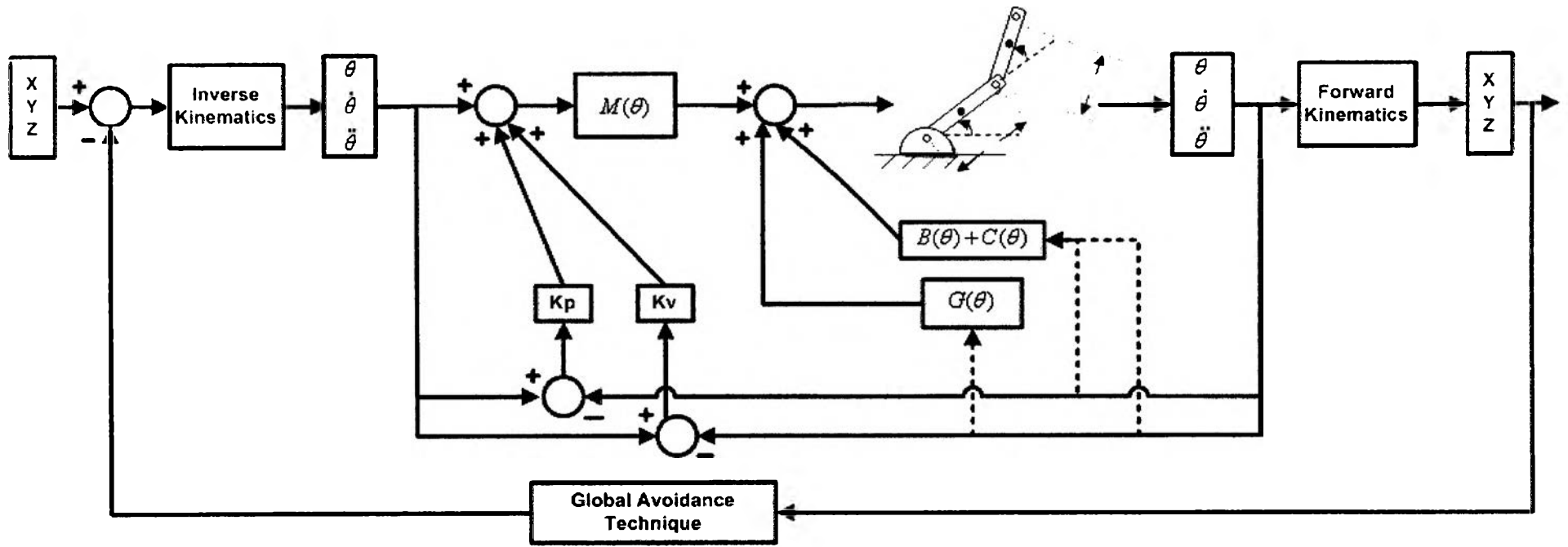
$${}^1R_0a_1 = \begin{bmatrix} -\frac{1}{2}\dot{\theta}_1^2 + g \sin \theta_1 \\ \frac{1}{2}\dot{\theta}_1^2 + g \cos \theta_1 \\ 0 \end{bmatrix}$$

$${}^2R_0a_2 = \begin{bmatrix} l((\sin \theta_2)\ddot{\theta}_1^2 - (\cos \theta_2)\dot{\theta}_1^2 - \frac{1}{2}\dot{\theta}_1^2 - \frac{1}{2}\dot{\theta}_2^2 - \dot{\theta}_1\dot{\theta}_2) + g \sin(\theta_1 + \theta_2) \\ l((\cos \theta_2)\ddot{\theta}_1 - (\sin \theta_2)\dot{\theta}_1^2 + \frac{1}{2}\ddot{\theta}_1 + \frac{1}{2}\ddot{\theta}_2) + g \cos(\theta_1 + \theta_2) \\ 0 \end{bmatrix}$$

#### 4.2.6 วิธีการควบคุมเส้นทางเดินเพื่อหลบหลีกสิ่งกีดขวาง

##### 1) วิธี Global Collision Avoidance

เป็นวิธีการพิจารณาการควบคุมที่ทำการตรวจสอบระยะพิกัด XYZ ก่อนจะทำการสั่งควบคุม ซึ่งปรับแก้ไขการเคลื่อนที่ทุกจุดของการเคลื่อนที่ที่พบอุปสรรคสิ่งกีดขวางใน Cartesian Space (Off-line Collision Avoidance) โดยเส้นทางเดินที่เกิดขึ้นจะถูกแก้ไขก่อนการเคลื่อนที่ เพื่อหลบหลีกสิ่งกีดขวางจริงและปรับค่าเกน Kp และ Kv ให้เหมาะสม ดังแสดงในรูปที่ 4.3

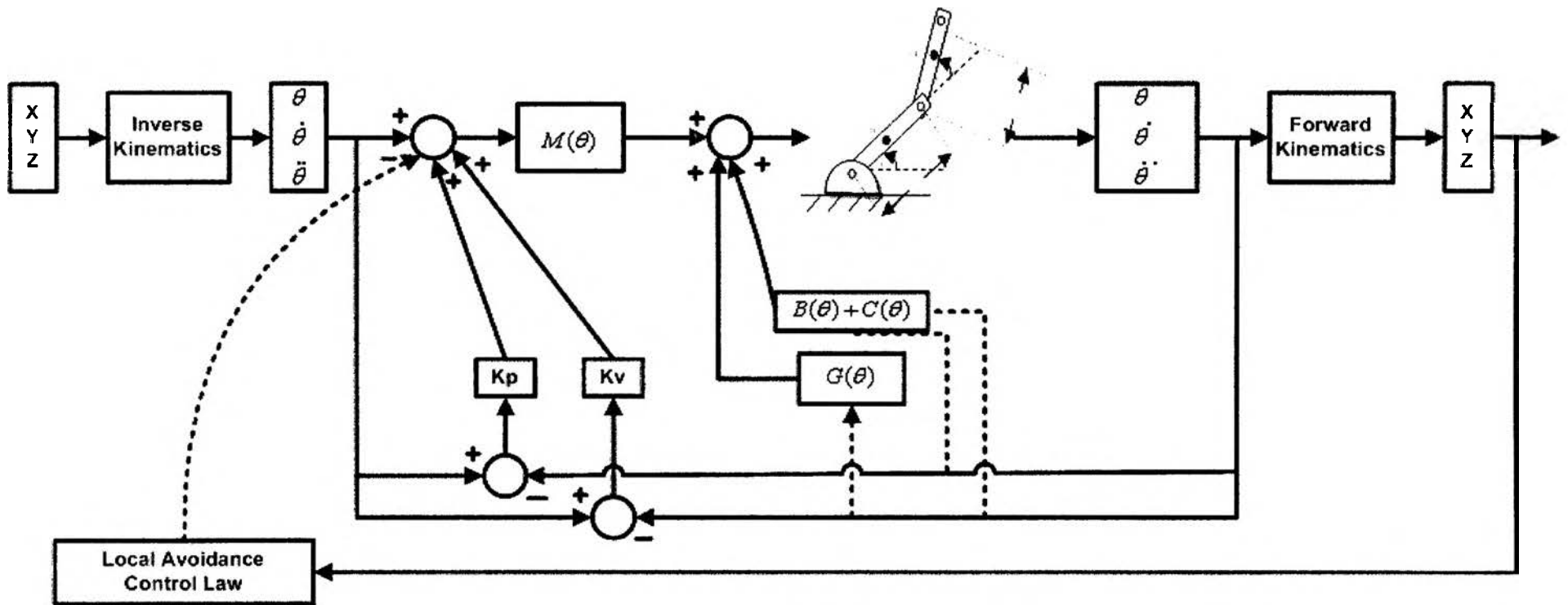


รูปที่ 4.3 แผนผังการควบคุม 2-Link Planar Arm ด้วยวิธี Global Collision Avoidance

## 2) วิธี Local Collision Avoidance

เป็นวิธีการพิจารณาการควบคุมที่ทำการตรวจสอบระยะพิกัด XYZ ในขณะที่ทำการสั่งควบคุม ซึ่งปรับแก้ไขการเคลื่อนที่ที่ละจุดของการเคลื่อนที่ที่พบอุปสรรคสิ่งกีดขวางใน Cartesian Space (Real-time Collision Avoidance) โดยเส้นทางเดินที่เกิดขึ้นจะถูกแก้ไขก่อนการเคลื่อนที่จริงและปรับค่าเกน  $K_p$  และ  $K_v$  ให้เหมาะสม เพื่อหลบหลีกสิ่งกีดขวาง ดังแสดงในรูปที่ 4.4





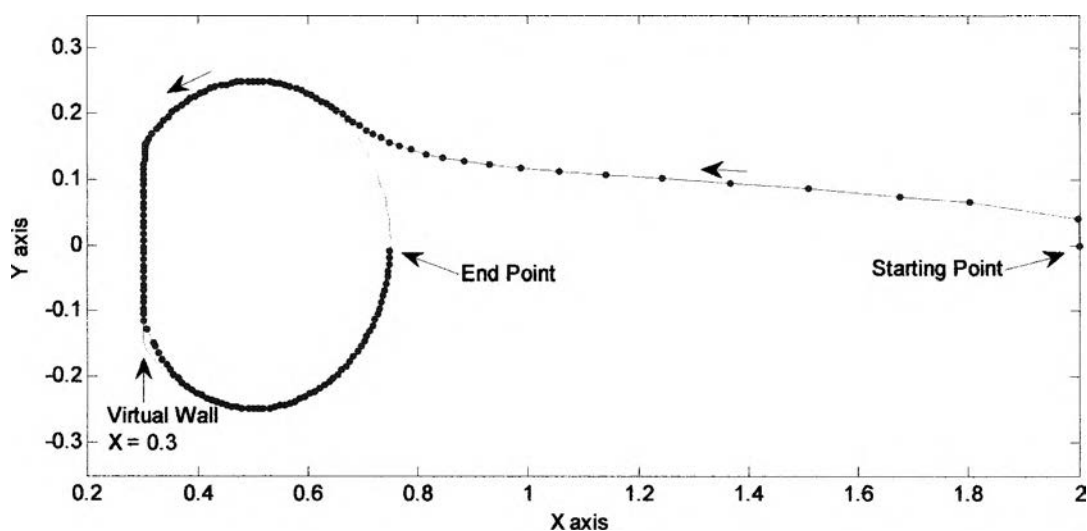
รูปที่ 4.4 แผนผังการควบคุม 2-Link Planar Arm ด้วยวิธี Local Collision Avoidance

#### 4.2.7 ผลการจำลองควบคุมเส้นทางเดินแบบต่างๆ เพื่อหลบหลีกสิ่งกีดขวาง (สามารถดูรายละเอียดของโปรแกรมได้ใน ภาคผนวก ค.2)

##### 1) ผลการจำลองควบคุมด้วยวิธี Global Collision Avoidance

##### เส้นทางเดินแบบวงกลม

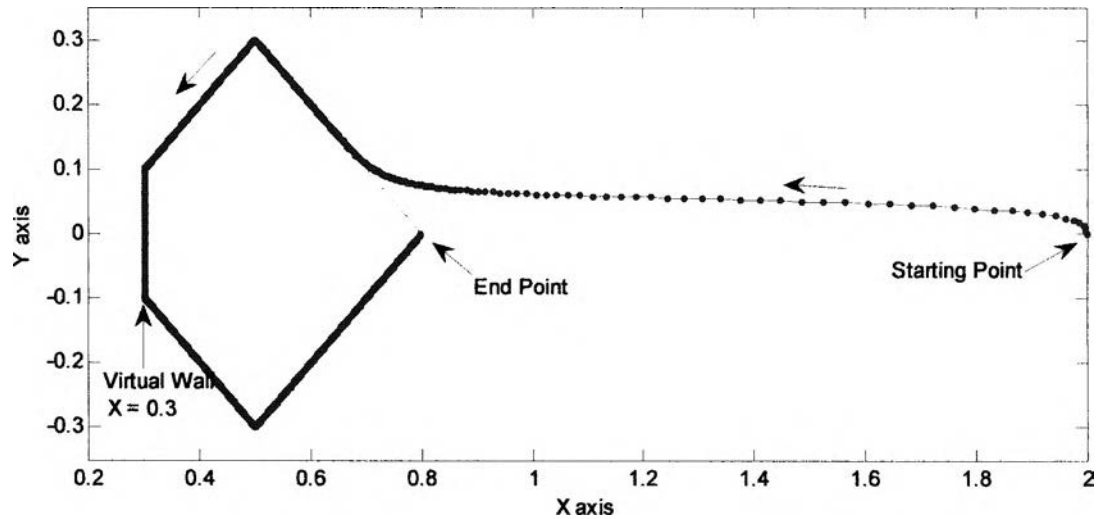
โดยที่จุดศูนย์กลางวงกลมอยู่ที่ (0.5,0) รัศมี 0.25 เมตร บนระนาบ  $Z = 0$  เมตร และแนวกำแพงอยู่ที่  $X = 0.3$  เมตร เริ่มเคลื่อนที่จากจุด (2,0) ไปหยุดที่จุด (0.75,0) ในทิศทางทวนเข็มนาฬิกา ด้วยความเร็วเชิงมุม  $= \pi/8$  rad/sec. ซึ่งผลการจำลองสามารถแสดงได้ดังรูปที่ 4.5



รูปที่ 4.5 เส้นทางเดินวงกลมของหุ่นยนต์หลบหลีกสิ่งกีดขวางด้วยวิธี Global เมื่อค่า  $K_p=100$ ,  $K_v=20$

##### เส้นทางเดินแบบสี่เหลี่ยม

โดยที่จุดศูนย์กลางของสี่เหลี่ยมมุมฉากอยู่ที่ (0.5,0) ความยาวแต่ละด้านเท่ากับ 0.4243 เมตร บนระนาบ  $Z = 0$  เมตร และแนวกำแพงอยู่ที่  $X = 0.3$  เมตร เริ่มเคลื่อนที่จากจุด (2,0) ไปหยุดที่จุด (0.8,0) ในทิศทางทวนเข็มนาฬิกา ด้วยความเร็ว 0.1697 m/s. ซึ่งผลการจำลองสามารถแสดงได้ดังรูปที่ 4.6

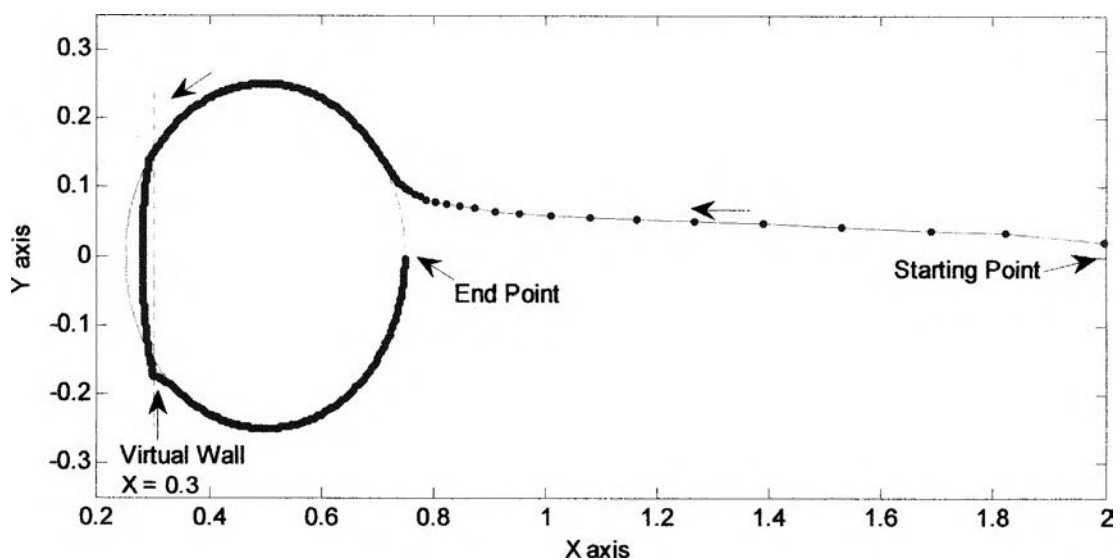


รูปที่ 4.6 เส้นทางเดินสี่เหลี่ยมของหุ่นยนต์หลบหลีกสิ่งกีดขวางด้วยวิธี Global  
เมื่อค่า  $K_p=100$ ,  $K_v=20$

## 2) ผลการจำลองควบคุมด้วยวิธี Local Collision Avoidance

### เส้นทางเดินแบบวงกลม

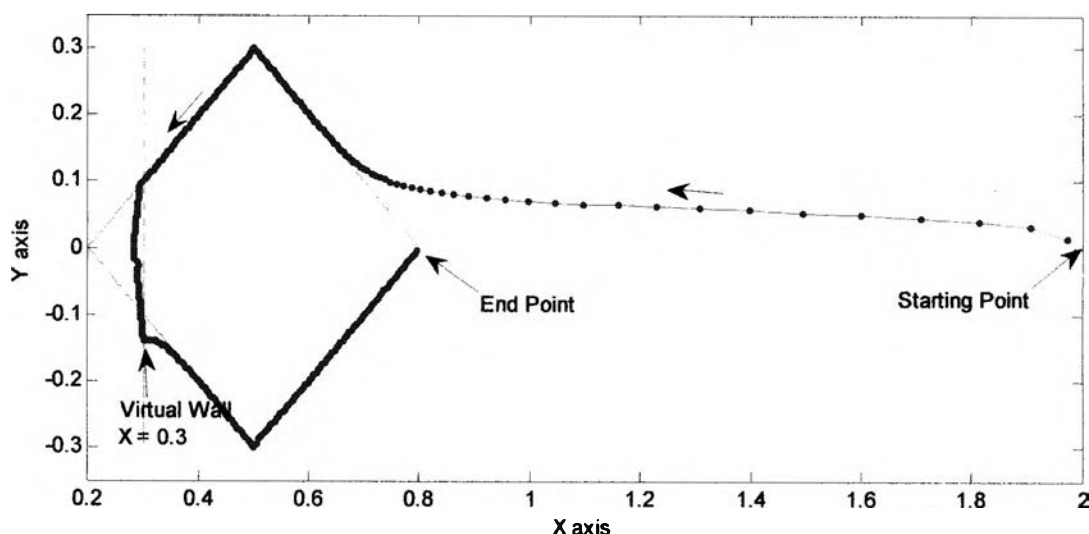
โดยที่จุดศูนย์กลางวงกลมอยู่ที่ (0.5,0) รัศมี 0.25 เมตร บนระนาบ  $Z = 0$  เมตร และแนวกำแพงอยู่ที่  $X = 0.3$  เมตร เริ่มเคลื่อนที่จากจุด (2,0) ไปหยุดที่จุด (0.75,0) ในทิศทางทวนเข็มนาฬิกา ด้วยความเร็วเชิงมุม  $= \pi/8$  rad/sec. ซึ่งผลการจำลองสามารถแสดงได้ดังรูปที่ 4.7



รูปที่ 4.7 เส้นทางเดินวงกลมของหุ่นยนต์หลบหลีกสิ่งกีดขวางด้วยวิธี Local  
เมื่อค่า  $K_p = 95$ ,  $K_v = 19.5$

### เส้นทางเดินแบบสี่เหลี่ยม

โดยที่จุดศูนย์กลางของสี่เหลี่ยมมุมฉากอยู่ที่ (0.5,0) ความยาวแต่ละด้านเท่ากับ 0.4243 เมตร บนระนาบ  $Z = 0$  เมตร และแนวกำแพงอยู่ที่  $X = 0.3$  เมตร เริ่มเคลื่อนที่จากจุด (2,0) ไปหยุดที่จุด (0.8,0) ในทิศทางทวนเข็มนาฬิกา ด้วยความเร็ว 0.1697 m/s. ซึ่งผลการจำลองสามารถแสดงได้ดังรูปที่ 4.8



รูปที่ 4.8 เส้นทางเดินสี่เหลี่ยมของหุ่นยนต์หลบหลีกสิ่งกีดขวางด้วยวิธี Local

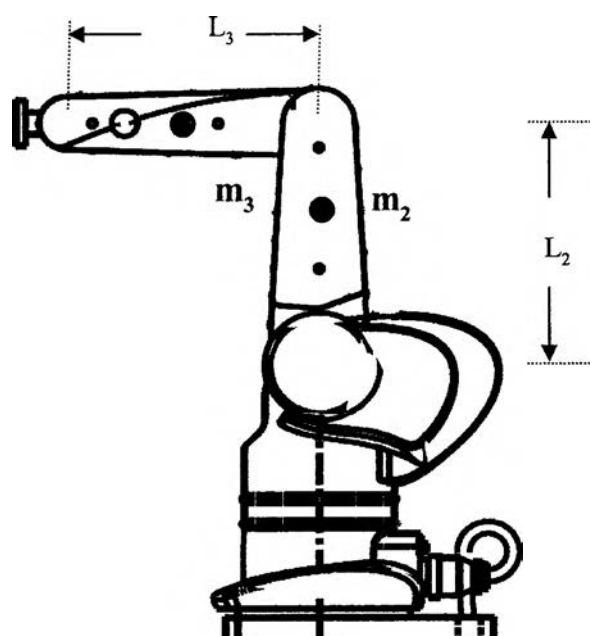
เมื่อค่า  $K_p = 35$ ,  $K_v = 11.83$

#### 4.2.8 สรุปผลการจำลองควบคุมเส้นทางเดินแบบวงกลมและสี่เหลี่ยม

จากการจำลองควบคุมแขนหุ่นยนต์ 2-Link Planar Arm บนเส้นทางเดินแบบวงกลมและแบบสี่เหลี่ยม ทั้ง 2 แบบนั้น การควบคุมเส้นทางเดินแบบสี่เหลี่ยมของทั้ง 2 วิธี เส้นทางที่ได้จากการควบคุมเพื่อหลบหลีกสิ่งกีดขวางนั้น ไม่เรียบเท่ากับการเคลื่อนที่บนเส้นทางที่เป็นวงกลม เนื่องจากในบริเวณที่เป็นมุมจะมีอัตราส่วนของ  $X$  และ  $Y$  ที่ไม่สม่ำเสมอ แม้ว่าจะทำการปรับค่าเกน  $K_p$  และ  $K_v$  ให้เหมาะสม ทำให้การปรับแก้ไขทางเดินไม่คงที่เมื่อเทียบกับเส้นทางเดินแบบวงกลม

ส่วนการควบคุมด้วยวิธี Global Collision Avoidance จะมีเสถียรภาพมากกว่าการควบคุมทางเดินด้วยวิธี Local Collision Avoidance เนื่องจากการแก้ไขเส้นทางเดินถูกปรับปรุงใหม่ทั้งหมดก่อนที่จะทำการขับหุ่นยนต์เพื่อหลบสิ่งกีดขวาง แต่พบว่าจะทำให้เวลาในการประมวลผลเพื่อแก้ไขเส้นทางเดินค่อนข้างนานเมื่อเทียบกับวิธี Local Collision Avoidance

### 4.3 การจำลองควบคุมแขนกล Articulated Robot เพื่อหลบหลีกสิ่งกีดขวาง



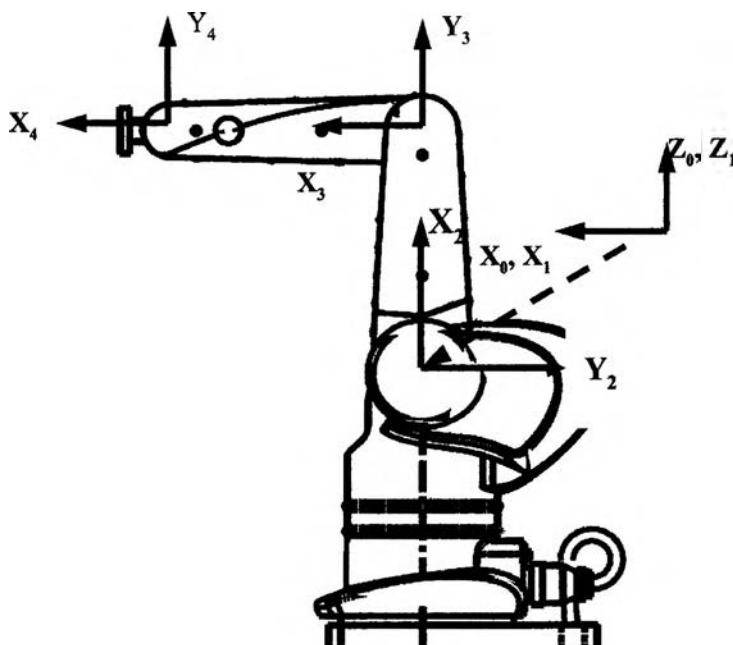
รูปที่ 4.9 Kinematics Model ของแขนกลแบบ Articulated Robot

#### 4.3.1 พารามิเตอร์ของหุ่นยนต์ (Robot Parameter) สามารถแสดงดังรูปที่ 4.9 กำหนดให้

ความยาวของ Link	$a_1 = a_2 = a_3$	= 10 inch.
ระยะทางจุดศูนย์กลางมวลจากจุดหมุน	$l_1 = l_2 = l_3$	= 5 inch.
มวลของ Link	$m_{l1} = m_{l2} = m_{l3}$	= 1 kg.
Inertia ของ Motor และ Link	$J_{ml1} = J_{ml2}$	= $200 \times 10^{-6}$ kg.m. <sup>2</sup>

#### 4.3.2 ดี-เอชพารามิเตอร์ (D-H parameter)

วิธี D-H parameter ช่วยในการหาจลนศาสตร์ไปข้างหน้า (Forward Kinematics) โดยจะตั้งเฟรมอ้างอิงที่แต่ละข้อต่อ แล้วหาค่าความสัมพันธ์ระหว่างแขนและข้อต่อ ดังแสดงในรูปที่ 4.10 และค่าตัวแปรที่เกี่ยวข้องกับ D-H parameter แสดงในตารางที่ 4.2



รูปที่ 4.10 การตั้งแกนอ้างอิงของแขนกลแบบ Articulated Robot

ตารางที่ 4.2 D-H parameter ของ Articulated Robot

Link No.(i)	Link Twist ( $\alpha_{i-1}$ )	Link Length( $a_{i-1}$ )	Link Offset ( $d_i$ )	Joint Angle ( $\theta_i$ )
1	0	0	0	$\theta_1$
2	-90	0	0	$\theta_2$
3	0	$L_2$	0	$\theta_3$
4	0	$L_3$	0	$\theta_4$

โดยที่  $\alpha_{i-1}, a_{i-1}$  เป็นค่าคงที่  
 $\theta_i, d_i$  เป็นตัวแปรตามขึ้นอยู่กับเคลื่อนที่ว่าเป็น Prismatic หรือ Revolute Joint

โดยวิธีของ Denavit-Hartenberg ซึ่งในปี ค.ศ.1955 R. S. Hartenberg และ J. Denavit [19] ได้เสนอวิธี D-H parameter จากรูปทั่วไปของ Homogeneous Transformation Matrix ดังนี้

$${}^{i-1}T_i = \begin{bmatrix} \cos \theta_i & -\sin \theta_i & 0 & a_{i-1} \\ \sin \theta_i \cos \alpha_{i-1} & \cos \theta_i \cos \alpha_{i-1} & -\sin \alpha_{i-1} & -d_i \sin \alpha_{i-1} \\ \sin \theta_i \sin \alpha_{i-1} & \cos \theta_i \sin \alpha_{i-1} & \cos \alpha_{i-1} & d_i \cos \alpha_{i-1} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

### 4.3.3 จลนศาสตร์ไปข้างหน้าและย้อนกลับ (Forward and Inverse Kinematics)

#### จลนศาสตร์ไปข้างหน้า (Forward Kinematics)

เป็นการโอนย้ายความสัมพันธ์ของตำแหน่งแขนกลจากการอ้างอิงเชิงมุมของแต่ละ Joint ในแกนอ้างอิงแบบตัวแปรข้อต่อ (Joint Space) มาเป็นการอ้างอิงเชิงเส้น 3 มิติแกน X, Y, Z (Cartesian Space) เพื่อบอกตำแหน่งปลายแขนกลเทียบกับเฟรมอ้างอิงเริ่มต้น (Base Frame) (สามารถศึกษารายละเอียดได้ในภาคผนวก ก)

$$\therefore {}^0_4T = \begin{bmatrix} \cos\theta_1 \cos(\theta_2 + \theta_3) & -\cos\theta_1 \sin(\theta_2 + \theta_3) & -\sin\theta_1 & T14 \\ \sin\theta_1 \cos(\theta_2 + \theta_3) & -\sin\theta_1 \sin(\theta_2 + \theta_3) & \cos\theta_1 & T24 \\ -\sin(\theta_2 + \theta_3) & -\cos(\theta_2 + \theta_3) & 0 & T34 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

โดยที่

$$L_2 = L_3 = l$$

$$T14 = (l \cos\theta_2 + l \cos(\theta_2 + \theta_3)) \cos\theta_1$$

$$T24 = (l \cos\theta_2 + l \cos(\theta_2 + \theta_3)) \sin\theta_1$$

$$T34 = -l \sin\theta_2 - l \sin(\theta_2 + \theta_3)$$

#### จลนศาสตร์ย้อนกลับ (Inverse Kinematics)

เป็นการโอนย้ายความสัมพันธ์ย้อนกลับของตำแหน่งแขนกล จากการอ้างอิงเชิงเส้น 3 มิติ แกน x, y, z ใน Cartesian Space ของปลายแขนกล มาเป็นการอ้างอิงในแกนอ้างอิงแบบตัวแปรข้อต่อ (Joint Space) เพื่อบอกตำแหน่งปลายแขนกลเทียบกับเฟรมอ้างอิงเริ่มต้น (Base Frame) ซึ่งจะพบว่าสามารถหาผลเฉลยได้หลายคำตอบขึ้นอยู่กับเงื่อนไขในการทำงานของแขนกล เช่น สิ่งกีดขวาง

#### วิธีหาผลเฉลยแบบ Geometric (สามารถศึกษารายละเอียดได้ในภาคผนวก ก)

##### Joint 1

$$\therefore \theta_1 = \text{atan2}(y, x) \quad (\text{ต้องพิจารณาเงื่อนไขประกอบเนื่องจากคำตอบมี 2 ค่า})$$

โดยที่

$$x = (l \cos\theta_2 + l \cos(\theta_2 + \theta_3)) \cos\theta_1$$

$$y = (l \cos\theta_2 + l \cos(\theta_2 + \theta_3)) \sin\theta_1$$

$$z = l \sin\theta_2 + l \sin(\theta_2 + \theta_3)$$

Joint 2

$$\beta = \cos^{-1} \left[ \frac{l^2 - l^2 - r^2}{-2lr} \right]$$

$$\psi = \sin^{-1}(z/r)$$

$$\therefore \theta_2 = \beta + \psi$$

โดยที่  $L_2 = L_3 = l$

Joint 3

$$\therefore \theta_3 = 180 - \cos^{-1} \left( \frac{l^2 + l^2 - r^2}{2ll} \right)$$

โดยที่  $L_2 = L_3 = l$

#### 4.3.4 จาโคเบียน (Jacobian)

การที่หุ่นยนต์มีการเคลื่อนที่ปฏิสัมพันธ์กับสิ่งแวดล้อม คุณสมบัติต่างๆจะเปลี่ยนแปลงอย่างต่อเนื่องตลอดการเคลื่อนไหว ทำให้เกิดการเปลี่ยนแปลงของระยะทาง แรงในแนวเชิงเส้น เชิงมุมเมื่อเทียบกับแกนอ้างอิง จึงได้ทำการหาวิธีการโอนย้ายความสัมพันธ์ความเร็ว แรงของตำแหน่งต่างๆบนแขนกล เช่น ความเร็วเชิงมุม ในกรณีที่เป็นข้อต่อแบบหมุน (Revolute Joint) และความเร็วเชิงเส้น ในกรณีที่เป็นข้อต่อแบบเลื่อน (Prismatic Joint) หรือปลายแขนกล (สามารถศึกษารายละเอียดได้ในภาคผนวก ก)

#### จาโคเบียนความเร็ว (Velocity Jacobian)

เป็นโอนย้ายความสัมพันธ์ความเร็วของตำแหน่งต่างๆบนแขนกล เช่น ความเร็วเชิงมุม ในกรณีที่เป็นข้อต่อแบบหมุน หรือความเร็วเชิงเส้น ในกรณีที่เป็นข้อต่อแบบเลื่อนหรือปลายแขน ระหว่างการอ้างอิงที่ฐานของหุ่นยนต์และการอ้างอิงที่ปลายแขนกล

$${}^0v_4 = {}^0J(\theta)\dot{\theta}$$

$${}^0v_4 = \begin{bmatrix} -(\cos\theta_2 + \cos(\theta_2 + \theta_3))\sin\theta_1 l & -(\sin\theta_2 + \sin(\theta_2 + \theta_3))\cos\theta_1 l & -\cos\theta_1(\sin(\theta_2 + \theta_3))l \\ (\cos(\theta_2 + \theta_3) + \cos\theta_2)\cos\theta_1 l & -(\sin\theta_2 + \sin(\theta_2 + \theta_3))\sin\theta_1 l & -\sin\theta_1(\sin(\theta_2 + \theta_3))l \\ 0 & -(\cos\theta_2 + \cos(\theta_2 + \theta_3))l & -\cos(\theta_2 + \theta_3)l \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix}$$

$$\therefore {}^0J(\theta) = \begin{bmatrix} -(\cos\theta_2 + \cos(\theta_2 + \theta_3))\sin\theta_1 l & -(\sin\theta_2 + \sin(\theta_2 + \theta_3))\cos\theta_1 l & -\cos\theta_1(\sin(\theta_2 + \theta_3))l \\ (\cos(\theta_2 + \theta_3) + \cos\theta_2)\cos\theta_1 l & -(\sin\theta_2 + \sin(\theta_2 + \theta_3))\sin\theta_1 l & -\sin\theta_1(\sin(\theta_2 + \theta_3))l \\ 0 & -(\cos\theta_2 + \cos(\theta_2 + \theta_3))l & -\cos(\theta_2 + \theta_3)l \end{bmatrix}$$



### จาโคเบียนแรง (Force Jacobian)

เป็นอินทรีย์ความสัมพันธ์แรงของตำแหน่งต่างๆบนแกนกล เช่น แรงบิด (ในกรณีที่เป็นข้อต่อแบบหมุน) หรือแรงเชิงเส้น (ในกรณีที่เป็นข้อต่อแบบเลื่อนหรือปลายแขน) ระหว่างการอ้างอิงที่ฐานของหุ่นยนต์และการอ้างอิงที่ปลายแขนกล

$$\therefore \tau = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} = \begin{bmatrix} 0 & 0 & L(\sin(\theta_2 + \theta_3) + \sin \theta_2) \\ L \sin \theta_3 & L(\cos \theta_3 + 1) & 0 \\ 0 & 0 & L(\cos(\theta_2 + \theta_3) + \cos \theta_2) \end{bmatrix} \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix}$$

### 4.3.5 พลศาสตร์ไปข้างหน้าและย้อนกลับ (Forward and Inverse Dynamics)

พลศาสตร์ของแขนกล เป็นความสัมพันธ์ของแรงบิด (Joint Torque) ที่กระทำที่ข้อต่อกับการเคลื่อนที่ของแขนกลใน Cartesian Space โดยสามารถคำนวณหาได้ 2 วิธี คือ วิธี Lagrangian Formulation และวิธี Recursive Newton-Euler Formulation ซึ่งในที่นี้ได้ทำการคำนวณด้วยวิธี Recursive Newton-Euler Formulation (สามารถศึกษารายละเอียดได้ในภาคผนวก ก)

#### พลศาสตร์ไปข้างหน้า (Forward Dynamics)

เป็นการคำนวณหาแรงบิด (Joint Torque) ที่ต้องการเพื่อใช้ในการขับเคลื่อนหุ่นยนต์ให้เคลื่อนที่ไปยังตำแหน่งที่ต้องการ

$$\tau_1 = \begin{bmatrix} \left( \frac{1}{3} m_1 l^2 + \frac{4}{3} m_2 l^2 + m_2 \cos \theta_2 l^2 \right) \ddot{\theta}_1 + \left( \frac{1}{3} m_1 l^2 + \frac{1}{2} m_2 \cos \theta_2 l^2 \right) \ddot{\theta}_2 \\ - m_2 \sin \theta_2 l^2 \dot{\theta}_1 \dot{\theta}_2 - \frac{1}{2} m_2 \sin \theta_2 l^2 \dot{\theta}_2^2 \\ \frac{1}{2} m_1 g l \cos \theta_1 + \frac{1}{2} m_2 g l \cos(\theta_1 + \theta_2) + m_2 g l \cos(\theta_1 + \theta_2) \end{bmatrix}$$

$$\tau_2 = \left( \frac{1}{3} m_2 l^2 + \frac{1}{2} m_2 \cos \theta_1 l^2 \right) \ddot{\theta}_1 + \left( \frac{1}{3} m_2 l^2 \right) \ddot{\theta}_2 + \frac{1}{2} m_2 \sin \theta_2 l^2 \dot{\theta}_1^2 + \frac{1}{2} m_2 g l \cos(\theta_1 + \theta_2)$$

$$\tau_3 = \left( \frac{1}{3} m_3 l^2 \right) \ddot{\theta}_3 + \frac{1}{2} m_2 (\sin \theta_2) l^2 + \frac{1}{2} m_2 g l \cos(\theta_1 + \theta_2)$$

## พลศาสตร์ย้อนกลับ (Inverse Dynamics)

เป็นการคำนวณหาความเร่งที่จุดศูนย์กลางมวลของแขน (Link) ที่เกิดขึ้นจากการควบคุมหุ่นยนต์ให้เคลื่อนที่ไปยังตำแหน่งที่ต้องการ

$${}^1R_0a_1 = \begin{bmatrix} 0 \\ 0 \\ \ddot{\theta}_1 \end{bmatrix} \times \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \ddot{\theta}_1 \end{bmatrix} \times \left\{ \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_1 \end{bmatrix} \times 0 \right\} + \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix}$$

$${}^2R_0a_2 = \begin{bmatrix} \frac{l}{2} \cos \theta_2 (\sin(\theta_1 + \theta_2 + \theta_3)) \ddot{\theta}_{12} \\ -\frac{l}{2} \sin \theta_2 (\sin(\theta_1 + \theta_2 + \theta_3)) \ddot{\theta}_{12} \\ \frac{l}{2} (\cos(\theta_1 + \theta_2 + \theta_3)) \ddot{\theta}_{12} \end{bmatrix} + \begin{bmatrix} \frac{l}{2} \cos \theta_2 (\cos(\theta_1 + \theta_2 + \theta_3)) (\dot{\theta}_{12})^2 \\ -\frac{l}{2} \sin \theta_2 (\cos(\theta_1 + \theta_2 + \theta_3)) (\dot{\theta}_{12})^2 \\ -\frac{l}{2} (\sin(\theta_1 + \theta_2 + \theta_3)) (\dot{\theta}_{12})^2 \end{bmatrix} + \begin{bmatrix} -l((\cos \theta_2) \dot{\theta}_{12})^2 + g \sin \theta_2 \\ l((\cos \theta_2) \dot{\theta}_{12})^2 + g \cos \theta_2 \\ -l(\cos \theta_2) \dot{\theta}_{12} \end{bmatrix}$$

$${}^3R_0a_3 = \frac{l}{2} \begin{bmatrix} \sin \theta_1 (\cos(\theta_2 + \theta_3)) \ddot{\theta}_{12} - (\cos \theta_1) \ddot{\theta}_3 + \cos \theta_1 (\cos(\theta_2 + \theta_3)) ((\dot{\theta}_{12})^2 + (\dot{\theta}_3)^2) \\ -\sin \theta_1 (\sin(\theta_2 + \theta_3)) \ddot{\theta}_{12} - (\cos \theta_1) \ddot{\theta}_3 - \cos \theta_1 (\cos(\theta_2 + \theta_3)) (\dot{\theta}_{12})^2 + \sin \theta_1 (\cos(\theta_2 + \theta_3)) \dot{\theta}_2 \dot{\theta}_3 - \cos \theta_1 \sin 2\theta_3 (\dot{\theta}_3)^2 \\ \sin(\theta_2 + \theta_3) + \cos(\theta_2 + \theta_3) (\cos \theta_1) \ddot{\theta}_{12} + ((\cos(\theta_2 + \theta_3)) - (\sin(\theta_2 + \theta_3)) - 2(\cos(\theta_2 + \theta_3)) (\sin(\theta_2 + \theta_3)) \cos \theta_1) \dot{\theta}_2 \dot{\theta}_3 - \sin \theta_1 (\dot{\theta}_{12})^2 \end{bmatrix}$$

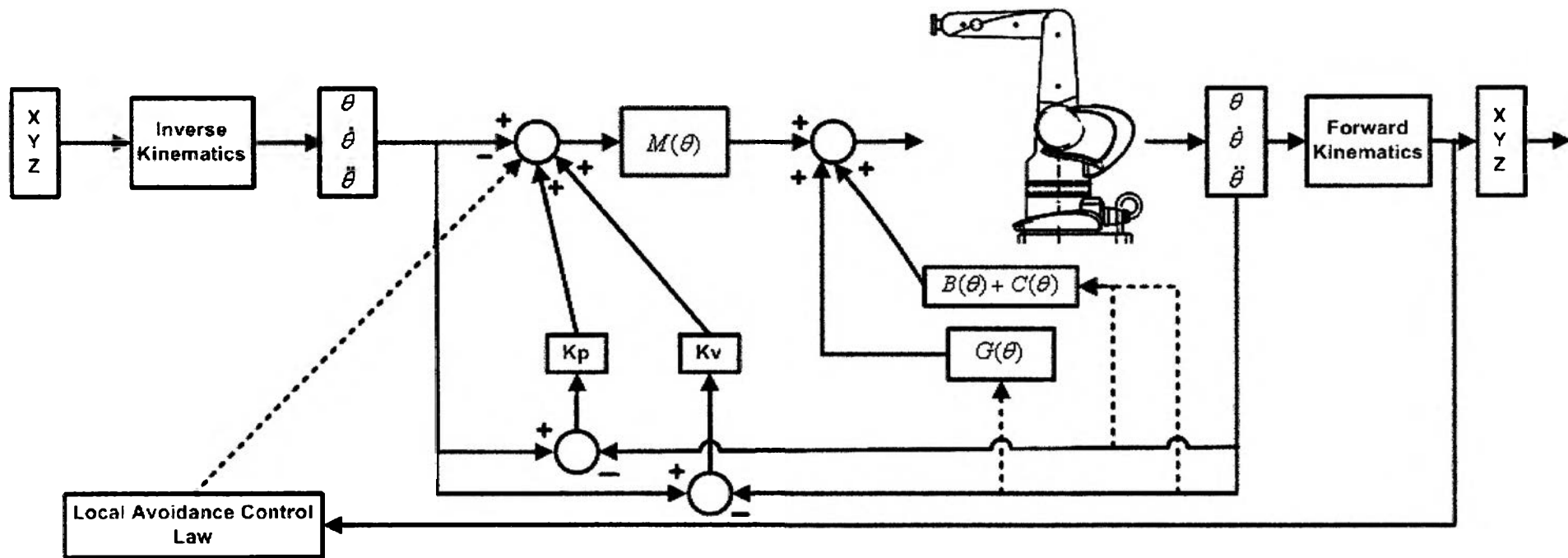
$${}^4R_0a_4 = \begin{bmatrix} \sin \theta_1 (\cos(\theta_2 + \theta_3)) \ddot{\theta}_{12} - (\cos \theta_1) \ddot{\theta}_3 + \cos \theta_1 (\cos(\theta_2 + \theta_3)) ((\dot{\theta}_{12})^2 + (\dot{\theta}_3)^2) \\ -\sin \theta_1 (\sin(\theta_2 + \theta_3)) \ddot{\theta}_{12} - (\cos \theta_1) \ddot{\theta}_3 - \cos \theta_1 (\cos(\theta_2 + \theta_3)) (\dot{\theta}_{12})^2 + \sin \theta_1 (\cos(\theta_2 + \theta_3)) \dot{\theta}_2 \dot{\theta}_3 - \cos \theta_1 \sin(\theta_2 + \theta_3) (\dot{\theta}_3)^2 \\ (\sin(\theta_2 + \theta_3) + \cos(\theta_2 + \theta_3)) (\cos \theta_1) \ddot{\theta}_{12} + (\cos(\theta_2 + \theta_3) - \sin(\theta_2 + \theta_3) - 2 \cos(\theta_2 + \theta_3) \sin(\theta_2 + \theta_3)) (\cos \theta_1) \dot{\theta}_2 \dot{\theta}_3 - (\sin \theta_1) (\dot{\theta}_{12})^2 \end{bmatrix}$$

### 4.3.6 วิธีการควบคุมเส้นทางเดินเพื่อหลบหลีกสิ่งกีดขวาง

ในการจำลองควบคุมหุ่นยนต์แบบ Articulated Robot นี้ เป็นการจำลองลักษณะของหุ่นยนต์ CRS Robot ที่พิจารณาองศาความอิสระเท่ากับ 3 ซึ่งจะเลือกจำลองควบคุมด้วยวิธีการ Local Collision Avoidance เพียงวิธีเดียวเนื่องจากเป็นการปรับแก้ไขเส้นทางเดินเพื่อหลบหลีกสิ่งกีดขวางที่มีการเคลื่อนที่ด้วย ซึ่งเป็นการจำลองปัญหาเกี่ยวกับการมีปฏิสัมพันธ์กับสิ่งแวดล้อมที่ตรงตามวัตถุประสงค์ของวิทยานิพนธ์นี้ (สามารถศึกษารายละเอียดได้ในภาคผนวก ค.2)

#### วิธี Local Collision Avoidance

เป็นวิธีการพิจารณาการควบคุมที่ทำการตรวจสอบระยะพิกัด XYZ ในขณะที่ทำการสั่งควบคุม ซึ่งปรับแก้ไขการเคลื่อนที่ที่ละจุดของการเคลื่อนที่ที่พบอุปสรรคสิ่งกีดขวางใน Cartesian Space (Real-time Collision Avoidance) โดยเส้นทางเดินที่เกิดขึ้นจะถูกแก้ไขก่อนการเคลื่อนที่จริงและปรับค่าเกน Kp และ Kv ให้เหมาะสม เพื่อหลบหลีกสิ่งกีดขวาง ดังแสดงในรูปที่ 4.11

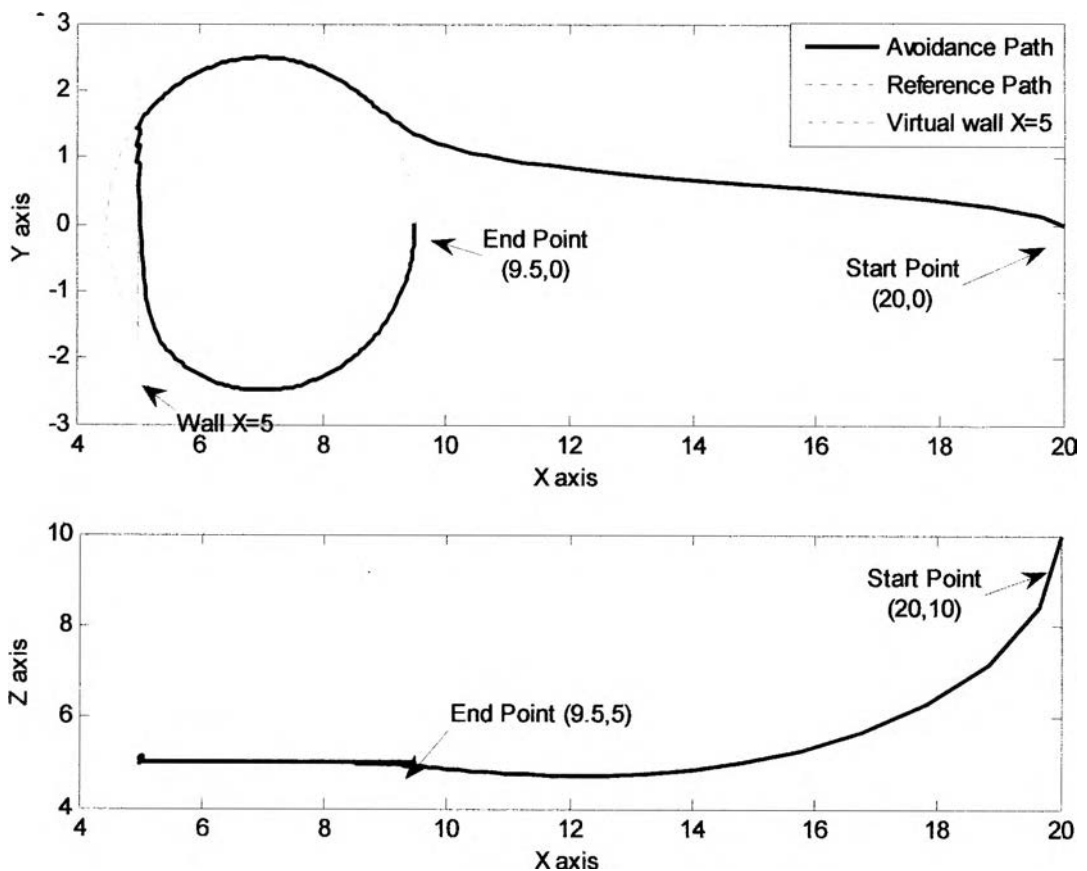


รูปที่ 4.11 แผนผังการควบคุม Articulated Robot ด้วยวิธี Local Collision Avoidance

#### 4.3.7 ผลการจำลองควบคุมเส้นทางเดินแบบวงกลมเพื่อหลบหลีกสิ่งกีดขวาง

ผลการจำลองควบคุมด้วยวิธี Local Collision Avoidance บนเส้นทางเดินแบบวงกลมบนระนาบ XY

โดยที่จุดศูนย์กลางวงกลมอยู่ที่ (7,0,5) รัศมี 2.5 นิ้ว ที่แนวระดับ  $Z = 5$  นิ้ว และแนวกำแพงอยู่ที่  $X = 5$  นิ้ว เริ่มเคลื่อนที่จากจุด (20,0,10) ไปหยุดที่จุด (9.5,0,5) ในทิศทางทวนเข็มนาฬิกา ด้วยความเร็วเชิงมุม  $= \pi/6$  radian/sec. ซึ่งผลการจำลองสามารถแสดงได้ดังรูปที่ 4.12



รูปที่ 4.12 เส้นทางเดินของหุ่นยนต์เพื่อหลบหลีกสิ่งกีดขวาง เมื่อค่า  $K_p=45$ ,  $K_v=13.42$

#### 4.3.8 สรุปผลการจำลองควบคุมเส้นทางเดินแบบวงกลม

จากการจำลองควบคุมเส้นทางเดินแบบวงกลมของหุ่นยนต์แบบ Articulated บนระนาบ XY จุดศูนย์กลางอยู่ที่ (7,0,5) โดยปลายแขนกลเริ่มต้นเคลื่อนที่จากตำแหน่ง (20,0,10) และมีเงื่อนไขควบคุมที่ระยะ  $X=5$  นิ้ว พบว่าการควบคุมด้วยวิธี Local Collision Avoidance จะทำการเคลื่อนที่ไปที่ละจุดและเมื่อพบสิ่งกีดขวางก็จะทำการปรับแก้ไขทีละจุด ซึ่งจากการจำลองพบว่าถ้าด้วยวิธี Local Collision Avoidance สามารถใช้ร่วมกับระเบียบวิธีเกรเดียนโปรเจกชันเพื่อทำการปรับเส้นทางเดินเพื่อหลบหลีกสิ่งกีดขวางแบบ On-line ได้อย่างมีประสิทธิภาพมากขึ้น ถ้าทำการปรับค่าของระยะก้าว (Step Size) ให้มีค่าน้อยลง แต่จะใช้เวลาในการประมวลผลที่นานขึ้น

#### 4.4 สรุปผลการจำลองควบคุม

จากผลการจำลองควบคุมแขนกลแบบ 2-Link Planar Arm และ Articulated Robot ด้วยวิธี Global Collision Avoidance ซึ่งเป็นวิธีการในการปรับแก้ไขเส้นทางเดินของแขนกลบนเส้นทางเดินทั้งแบบวงกลมและสี่เหลี่ยม โดยพิจารณาเส้นทางเดินที่ต้องทำการแก้ไขเพื่อหลบหลีกสิ่งกีดขวางทั้งหมดทุกจุดก่อน หลังจากนั้นถึงทำการแปลงเส้นทางเดินให้อยู่ในรูปแบบแกนอ้างอิงตัวแปรข้อต่อ เพื่อใช้ในการจับควบคุมหุ่นยนต์ตามกฎการควบคุมต่อไป พบว่าค่อนข้างใช้เวลาในการประมวลผลนานและเรียกว่าเป็น Off-line Collision Avoidance ส่วนวิธี Local Collision Avoidance เป็นวิธีการปรับแก้ไขเส้นทางเดินของหุ่นยนต์ โดยทำการพิจารณาเส้นทางเดินขณะที่กำลังเคลื่อนที่ ถ้าตรงตามเงื่อนไขของการกีดขวางเส้นทางเดิน ก็จะทำกรปรับแก้ไขในการเคลื่อนที่ครั้งต่อไปก่อนที่จะทำการเคลื่อนที่ ซึ่งวิธีดังกล่าวใช้เวลาในการประมวลผลไม่นานและเรียกได้ว่าเป็น On-line Collision Avoidance โดยที่ทั้งวิธี Global และ Local Collision Avoidance สามารถเคลื่อนที่บนเส้นทางเดินทั้งแบบวงกลมและสี่เหลี่ยม เพื่อหลบหลีกกำแพงซึ่งกีดขวางเส้นทางเดิน โดยใช้เทคนิคการประมาณจาโคเบียนขณะที่หุ่นยนต์เคลื่อนที่แทนการใช้ตัวแปรพารามิเตอร์ต่างๆของหุ่นยนต์ เพื่อนำข้อมูลความสัมพันธ์ของตัวแปรควบคุมที่จำเป็นมาแก้ไขค่าความผิดพลาดโดยระเบียบวิธี Gradient Projection เพื่อทำการปรับเส้นทางเดินของหุ่นยนต์ใหม่ ซึ่งทำให้ระบบควบคุมหุ่นยนต์ระบบดังกล่าวมีความยืดหยุ่นและสามารถนำไปใช้ทำการจำลองควบคุมกับหุ่นยนต์อุตสาหกรรมแบบอื่นๆได้