
Elementos básicos de matemáticas en la cinemática inversa

Oscar Guerrero Miramontes, Sergio Flores García María D. González Quezada, Juan Luna González, Juan E. Chávez Pierce, Luis L. Alfaro Avena

Universidad Autónoma de Ciudad Juárez

RESUMEN

En este artículo mostramos los hechos que determinan la configuración de enlaces en objetos sólidos. Es posible efectuar predicciones de estas configuraciones por medio de la aplicación de las leyes de transformación de coordenadas homogéneas, las cuales relacionan los marcos de referencia del movimiento. Las ecuaciones de movimiento pueden ser numéricamente resueltas utilizando el método algorítmico de Newton-Raphson en función de la unión de los ángulos para un sistema de brazo manipulador, en el plano con tres grados de libertad. Algunas simulaciones fueron realizadas en una animación flash macromedia desarrollada que nos permitió introducir los ángulos de unión para la posición deseada.

Palabras clave: Cinemática inversa, Newton-Raphson, ángulos de unión, cuerpos rígidos.

INTRODUCCIÓN

El área de conocimiento llamada cinemática inversa es el proceso de determinar los parámetros de una unión de objetos flexible (una cadena cinemática) con el propósito de adquirir una posición deseada (Grochow et al., 2004). Esta área de conocimiento es un tipo de planeación del movimiento, siendo también relevante para la programación de juegos y a la animación en 3D (Flores et al., 2008). La aplicación más común es conectar físicamente a los personajes con la vida real, por ejemplo, hacer que los pies toquen la tierra firmemente sobre un terreno. Un juego de parámetros matemáticos son articulados a través de varias representaciones matemáticas para desarrollar un conjunto físico de segmentos rígidos conectados en varios puntos (Goldstein, 2002). Como consecuencia, un número enorme de ángulos variantes pueden ser unidos para generar un número indefinido de configuraciones. Es

posible encontrar Información más técnica y detallada en muchas referencias en artículos de divulgación. Invitamos a los lectores curiosos a leer estos artículos para que amplíen su conocimiento en esta interesante área.

FUNDAMENTO MATEMÁTICO

Un cuerpo rígido puede ser considerado como un conjunto de puntos en el espacio cuando la distancia entre los dos permanece constante. Esta restricción es de relevante importancia, ya que reduce el número de grados de libertad necesaria para expresar el movimiento como $3N$. En este caso, N es el número de objetos que describen al objeto sólido a solamente 6. De estos seis grados de libertad, tres definen la translación y los restantes definen la rotación (Angeles, 2003).

La rotación de un cuerpo rígido se expresa por medio de los ángulos de Euler.

Usando la convención z-x-z. La rotación del cuerpo rígido es dado por:

$$\mathbf{R}(\theta, \beta, \alpha) = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \beta & \sin \beta \\ 0 & -\sin \beta & \cos \beta \end{bmatrix} \begin{bmatrix} \cos \alpha & \sin \alpha & 0 \\ -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} = \dots \quad 1$$

$$\begin{bmatrix} c_\alpha c_\gamma - c_\beta s_\alpha s_\gamma & c_\gamma s_\alpha + c_\alpha c_\beta s_\gamma & s_\beta s_\gamma \\ -c_\beta c_\gamma s_\alpha - c_\alpha s_\gamma & c_\alpha c_\beta c_\gamma - s_\alpha s_\gamma & c_\gamma s_\beta \\ s_\alpha s_\beta & -c_\alpha s_\beta & c_\beta \end{bmatrix}$$

en donde la matriz en la extrema derecha representa la rotación α alrededor del eje z del marco de referencia original. La matriz en el medio representa la rotación β alrededor de un eje intermedio x el cual es la "línea de". La matriz a la extrema izquierda representa la rotación θ alrededor del eje z del marco de referencia final. En las matrices en (1), C_α representa $\cos(\alpha)$, S_β representa $\sin(\beta)$, y de manera similar para los otros subíndices.

$$\begin{pmatrix} \cos \square & \sin \square & t_x \\ -\sin \square & \cos \square & t_y \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} R(\theta) | \vec{t} \\ 0 | 1 \end{pmatrix}, R(\square) \in \square^2 \quad 3$$

La misma representación es conveniente en un escenario en 3D, pero en este artículo nos concentramos en el escenario en donde el movimiento se encuentra restringido al plano x - y .

Para el caso especial de la contracción de la rotación alrededor del plano x - y en la ecuación (1) $\alpha = \beta = 0$. y entonces la rotación alrededor del plano x - y esta dado por:

Debido a las propiedades de los cuerpos rígidos (un cuerpo en el cual la posición relativa de todos sus puntos es coistante), la transformación (3) operando en el punto representado por p debe preservar la distancia entre pares de puntos como:

$$\mathbf{R}(\theta) = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad 2$$

$$(\mathbf{R}(\theta), \vec{t}): \vec{p} \mapsto \mathbf{R}(\theta)\vec{p} + \vec{t} \quad 4$$

Por conveniencia, los componentes translacionales y rotacionales se encuentran en una sola matriz:

por lo que una regla general para transformaciones sucesivas puede ser derivada (Peters, 2006):

$$(\mathbf{R}(\theta_1), t_1)(\mathbf{R}(\theta_2), t_2) \vec{p} = (\mathbf{R}(\theta_1)\mathbf{R}(\theta_2), \mathbf{R}(\theta_1)t_2 + t_1) \vec{p} \quad 5$$

Como existe un número infinito de ejes de rotación en donde el cuerpo rígido puede rotar, la necesidad de encontrar una relación para trasladar el eje a otro punto en el espacio es de gran importancia. Sea

$p = (p_x, p_y)$ el punto en donde la rotación toma lugar en el espacio. Entonces, la condición es que el punto p tiene que ser invariante bajo la transformación de rotación, y eso significa que:

$$\begin{pmatrix} R(\theta) & | & \vec{t} \\ 0 & | & 1 \end{pmatrix} = \begin{pmatrix} R(\theta) & | & \vec{p} - R(\theta)\vec{p} \\ 0 & | & 1 \end{pmatrix} \rightarrow \vec{t} = (I - R(\theta))\vec{p} \quad 6$$

de esta manera, encontramos el valor de t necesaria para mantener la transformación invariante para el punto p .

EL MANIPULADOR EN EL PLANO

Un sistema de un brazo manipulador en el plano consiste en un conjunto de cuerpos rígidos conectados por varias uniones. Los tipos de uniones para este caso especial son del tipo revolvente. Usando las capacidades del programa de animación Adobe Flash (Riley, 2006), una cadena cinemática de 3 grados de libertad fue implementada. La Figura 1 muestra la imagen del manipulador.

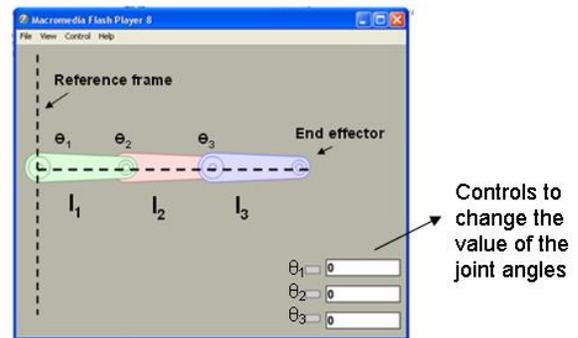


Fig. 1. Imagen de un manipulador y su configuración inicial, así como la posición del *end effector* en $(l_1 + l_2 + l_3, 0)$

El procedimiento para encontrar la posición del *end effector* como una función de los ángulos de unión es:

$$K(\theta) = \prod_{i=1}^3 A_i(\theta_i) = A_1(\theta_1)A_2(\theta_2)A_3(\theta_3) \quad 7$$

donde $A_1(\theta_1), A_2(\theta_2), A_3(\theta_3)$ son:

$$A_1(\theta_1) = \begin{pmatrix} R(\theta_1) & | & t_1 \\ 0 & | & 1 \end{pmatrix} = \begin{pmatrix} \cos \theta_1 & -\sin \theta_1 & t_{1x} \\ \sin \theta_1 & \cos \theta_1 & t_{1y} \\ 0 & 0 & 1 \end{pmatrix}$$

de manera que, el valor de \vec{t}_1 está dado por la relación $\vec{t}_1 = (I - R(\theta_1))\vec{p}$,

ahora si $\vec{p} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \rightarrow \vec{t}_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$.

$$A_2(\theta_2) = \begin{pmatrix} R(\theta_2) & t_2 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} \cos \theta_2 & -\sin \theta_2 & t_{2x} \\ \sin \theta_2 & \cos \theta_2 & t_{2y} \\ 0 & 0 & 1 \end{pmatrix}$$

de manera que, el valor de \vec{t}_2 está dado por la relación $\vec{t}_2 = (1 - R(\theta_2))\vec{p}$,

$$\text{ahora si } \vec{p} = \begin{pmatrix} I_1 \\ 0 \end{pmatrix} \rightarrow \vec{t}_2 = \begin{pmatrix} I_1(1 - \cos \theta_2) \\ -I_1 \sin \theta_2 \end{pmatrix}.$$

$$A_3(\theta_3) = \begin{pmatrix} R(\theta_3) & t_3 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} \cos \theta_3 & -\sin \theta_3 & t_{3x} \\ \sin \theta_3 & \cos \theta_3 & t_{3y} \\ 0 & 0 & 1 \end{pmatrix}$$

así, el valor de \vec{t}_2 está dado por la relación $\vec{t}_3 = (1 - R(\theta_3))\vec{p}$,

$$\text{ahora si } \vec{p} = \begin{pmatrix} I_1 + I_2 \\ 0 \end{pmatrix} \rightarrow \dots \vec{t}_3 = \begin{pmatrix} (I_1 + I_2)(1 - \cos \theta_3) \\ -(I_1 + I_2) \sin \theta_3 \end{pmatrix}$$

aquí A_1 , A_2 y A_3 constituyen la matriz que define la rotación alrededor de la unión definida por el parámetro \vec{P} . Para A_1 el punto es igual a $(0, 0)$, para A_2 , \vec{P} es $(I_1, 0)$ y finalmente para A_3 , \vec{P} debe ser igual a $(I_1 + I_2, 0)$. En donde, I_1, I_2 y I_3 son las longitudes de los enlaces del manipulador.

Después de desarrollar el producto de matrices, la matriz $K(\theta_i)$ es la siguiente:

$$K(\theta_1, \theta_2, \theta_3) = \begin{pmatrix} \cos(\theta_1 + \theta_2 + \theta_3) & -\sin(\theta_1 + \theta_2 + \theta_3) & t'_x \\ \sin(\theta_1 + \theta_2 + \theta_3) & \cos(\theta_1 + \theta_2 + \theta_3) & t'_y \\ 0 & 0 & 1 \end{pmatrix} \quad 8$$

aquí t'_x, t'_y tienen el siguiente valor:

$$\begin{aligned} t'_x &= I_1 C \theta_1 + I_2 C(\theta_1 + \theta_2) - (I_1 + I_2) C(\theta_1 + \theta_2 + \theta_3) \\ t'_y &= I_1 S \theta_1 + I_2 S(\theta_1 + \theta_2) - (I_1 + I_2) S(\theta_1 + \theta_2 + \theta_3) \end{aligned} \quad 9$$

ALGORITMO DE CINEMÁTICA INVERSA

Teniendo la matriz $K(\theta_i)$ dada en (3.2), podemos describir el movimiento del manipulador por:

$$K(\theta_i) : \vec{v} \rightarrow \vec{v}' \quad 10$$

En otras palabras, si aplicamos (8) a la posición del *end effector* v , podemos moverlo a una posición definida por v' . La

cinemática inversa utiliza otro acercamiento: Teniendo la posición $v' = (X_e, Y_e)$, debemos encontrar los valores de los ángulos de unión θ_i que pueden mover al “end effector” a la posición deseada.

$$K^{-1}(v') : v \rightarrow \theta \quad 11$$

Un algoritmo clásico para encontrar las raíces de un sistema de n ecuaciones con n variables es el método de Newton-Raphson (Hassani, 2003).

$$\theta^{(n+1)} = \theta^{(n)} - J^{-1}(\theta^{(n)})f(\theta^{(n)}) \quad 12$$

y como consecuencia, un sistema de n ecuaciones debe ser igual a cero si lo que queremos es encontrar las raíces:

$$K(\theta_i)v = v' \leftrightarrow K(\theta_i)v - v' = 0 \quad 13$$

Aplicando la definición a nuestra matriz K del manipulador, e igualando $v = (I_1 + I_2 + I_3, 0, 1)$, el siguiente sistema de ecuaciones es obtenido.

$$\begin{aligned} -x_e + l_1 C\theta_1 + l_2 C(\theta_1 + \theta_2) + l_3 C(\theta_1 + \theta_2 + \theta_3) &= 0 \\ -y_e + l_1 S\theta_1 + l_2 S(\theta_1 + \theta_2) + l_3 S(\theta_1 + \theta_2 + \theta_3) &= 0 \\ \theta_1 + \theta_2 + \theta_3 - \arctan(y_e/x_e) &= 0 \end{aligned} \quad 14$$

La ecuación $\theta_1 + \theta_2 + \theta_3 - \arctan(y_e/x_e) = 0$ se introduce para tener el juego completo de n ecuaciones y n variables para resolver el sistema (14). El ángulo formado por el eje de la referencia horizontal y la posición final del efector (y_e, x_e) , puede ser encontrado por simple observación trigonométrica, y es dada por la

suma de los ángulos de unión individuales θ_1, θ_2 y θ_3

Enseguida podemos resolver el sistema de ecuaciones usando el programa Mathematica a través de la función FindRoot. Las soluciones correspondientes son mostradas en la Figura 2.

```
f1 = -1.5 + Cos[θ1] + Cos[θ1 + θ2] + Cos[θ1 + θ2 + θ3];
f2 = Sin[θ1] + Sin[θ1 + θ2] + Sin[θ1 + θ2 + θ3];
f3 = θ1 + θ2 + θ3 - ArcTan[ye / xe];

FindRoot[{f1, f2, f3}, {{θ1, π/3}, {θ2, π/6}, {θ3, π/6}}]

{θ1 → -7.6013, θ2 → 15.2026, θ3 → -7.6013}
```

Fig. 2. Soluciones obtenidas con Mathematica.

Mathematica provee los valores de θ_1, θ_2 y θ_3 necesarias para mover el *end effector* al punto $v' = (\frac{l}{2}, 0)$. Desafortunadamente Mathematica no provee los resultados en términos de ángulos menores que 2π radianes, pero podemos obtener la conversión de manera sencilla. Aquí, los valores de las longitudes son $l_1 = l_2 = l_3 = 1$.

Repetiendo el mismo algoritmo para diferentes posiciones del *end effector* v' , es posible crear una trayectoria que puede ser seguida por nuestro manipulador, por lo que podemos observar su movimiento. La Figura 3 muestra una imagen con los arreglos de las posiciones de los enlaces necesarias para que la posición del *end effector* esté dada por $v' = (1.5, 0)$. La Tabla 1 ilustra las unions y los correspondientes ángulos.

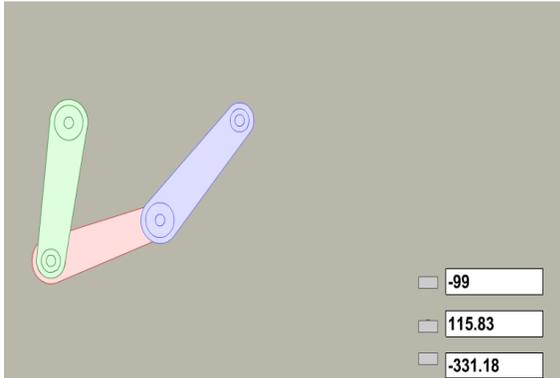


Fig. 3. Arreglos de las posiciones de los enlaces necesarias para que la posición del *end effector* esté dada por $v'=(1.5,0)$.

Tabla 1. Uniones y los ángulos correspondientes.

Unión	Valor del ángulo (en grados)
Unión 1 Θ_1	-99°
Unión 2 Θ_2	115.83°
Unión 3 Θ_3	-331.18°

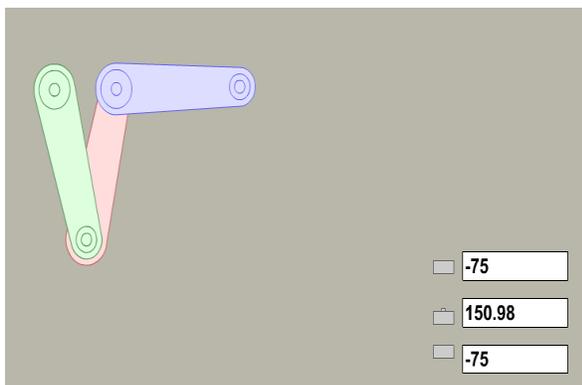


Fig. 4. Arreglo de los enlaces necesarios para que el *end effector* alcance la posición $v'=(1.5,0)$.

Tabla 2. Uniones y los ángulos correspondientes.

Unión	Valor del ángulo (en grados)
Unión 1 Θ_1	-75°
Unión 2 Θ_2	150.98°
Unión 3 Θ_3	-75°

La Figura 4 muestra otro arreglo de los enlaces necesarios para que el *end effector* alcance la posición $v'=(1.5,0)$. La Tabla 2 ilustra las uniones y los correspondientes ángulos.

Es importante mencionar que la solución encontrada usando el algoritmo IK no es única. En otras palabras, puede haber más de una configuración de uniones para mover el *end effector* al punto deseado v' . Aunque podemos reducir los tres grados de libertad del manipulador, manteniendo fijas las coordenadas del *end effector*, dadas por (x_e, y_e) , aún dejamos el parámetro angular que describe la rotación del sistema como una variable libre.

CONCLUSIONES

El método Newton-Rhapson es eficiente pero con ciertas limitaciones, ya que requiere una estimación inicial de los valores de unión. También, los cálculos de la inversa del Jacobiano puede ser considerada como una operación que consume mucho tiempo de la computadora. Este tipo de técnica matemática puede ser usada en varios campos de investigación como robótica, física moderna, diseño de prótesis médicas, animaciones gráficas y sistemas de producción

REFERENCIAS

Angeles J., (2003). Fundamentals of robotic mechanical systems, Springer, New York, p. 48

Flores S., Chávez J. E., Luna J., González M. D., M. González V. y Hernández A., (2008). El aprendizaje de la física y las matemáticas en contexto, CULCyT, **24**, 19-24.

Goldstein H. (2002). Classical mechanics, Addison Wesley, San Francisco, p. 150,

Grochow K., Martin S. L., Hertzmann A. y Popović Z. (2004). Style-based inverse kinematics,

ACM Transactions on Graphics (TOG) 23(3): 522–531.

Hassani S. (2003). Mathematical Methods Using *Mathematica*, Springer, Verlag, p. 15.

Peters K., (2006). Foundation ActionScript animation, (Friends of ED, Calif, p. 283.

Riley K. F., (2006). Mathematical methods for physics and engineering, (Cambridge University Press, Cambridge, p. 985.