LOS CUATERNIONES EN VISIÓN ROBOTICA

Kamlofsky Jorge A., Bergamini María L.

CAETI - Universidad Abierta Interamericana
Av. Montes de Oca 725 – Buenos Aires – Argentina.
{jorge.kamlofsky,maria.bergamini}@uai.edu.ar

Resumen: Los cuaterniones son números complejos de 4 componentes. A pesar de poseer propiedades algebraicas y operacionales muy buenas, poco después de su invención fueron opacados por el desarrollo del cálculo vectorial. Sin embargo en estos últimos años crece notablemente la cantidad de implementaciones de cuaterniones en diversas disciplinas atraídas por notables ventajas relacionadas con la simplicidad, eficiencia y características algebraicas. Este trabajo detalla su uso en visión robótica para representar rotaciones y compara tiempos de cálculo con otro método tradicionalmente usado para este fin.

Palabras claves: *rotaciones, cuaterniones, álgebra no conmutativa*
2000 AMS Subjects Classification: 21A54 - 55P5T4

1. INTRODUCCIÓN

Los cuaterniones son números hipercomplejos de cuatro componentes; una componente real y tres componentes imaginarias. Se los puede denotar de la forma $q = w + x.i + y.j + z.k$ con (x,y,z) vector en el espacio, w parte real y (i,j,k) base imaginaria.
Los cuaterniones tienen una notación compacta y son muy fáciles de componer. Las operaciones usando cuaterniones son muy eficientes en comparación con las que usan matrices, ya que requieren menor cantidad de operaciones básicas y menor espacio de almacenamiento. Conforman una representación no singular y solucionan el problema de “pérdida de dimensionalidad” conocido como gimbal lock (Salmeron Quiroz et al., 2009a). Gracias a todo esto se los usa en aplicaciones que requieren rotaciones o ubicaciones en el espacio bajo demanda en tiempo real.
Los cuaterniones forman una estructura de anillo con división, es decir, un anillo donde todos sus elementos no nulos son inversibles. En particular, los cuaterniones forman estructura de anillo con división no conmutativo. El conjunto de todos los cuaterniones de norma uno (llamados cuaterniones unitarios) forman un grupo multiplicativo no conmutativo conocido como SU(2) el cual es isomórfico con las matrices complejas de 2x2 de determinante uno.
Fueron usados por la física a fines del Siglo XIX y principios del siglo XX. En estos últimos años vuelven a ser tenidos en cuenta. Actualmente se usan en robótica, visión artificial en 3D (interpretación automática de imágenes digitales), navegación aeroespacial, criptografía y demás. Si bien el desarrollo vectorial posterior a la invención de los cuaterniones dominó el estudio del espacio n-dimensional, el álgebra de cuaterniones posee ventajas notables si se desea trabajar en el espacio tridimensional. Gracias a ello, los cuaterniones son usados en importantes implementaciones de diferentes disciplinas.

2. APLICACIONES DE CUATERNIONES

Gracias a su notación compacta y al hecho de operarse más rápido que mediante matrices, el cuaternión está siendo usado cada vez más en implementaciones que requieran velocidad y eficiencia de cálculo. En Sanchez Peña y Alonso (2010) se presentan ejemplos de control de navegación aeroespacial. En éste se presentan ejemplos de control de navegación de vehículos cohete sonda y de satélites de observación de la tierra usando cuaterniones. En Serrano et al. (2014) se presentan detalles del uso de cuaterniones para la orientación de vehículos aeroespaciales. En Barcala et al. (2004) se presenta una aplicación de detección de plásticos para reciclado de residuos urbanos mediante análisis espectrográfico.
El hecho de operarse más rápido que con matrices y que aún el producto no sea conmutativo pueden ser algunos de los argumentos por los cuales en los últimos años se han presentado algunos trabajos de criptografía usando cuaterniones. En Anand et al., (2009) se plantea el uso de cuaterniones y conjuntos de Julia para generar claves simétricas en tiempo real para criptografía. En Shankar y Selvakumar (2014) se

presenta la idea de combinar fracciones de Farey con las propiedades algebraicas de los cuaterniones para obtener un esquema criptográfico de alta seguridad.
El algoritmo de Shor (Shor, 1994) sobre computación cuántica ha logrado reducir a tiempo polinómico la complejidad de los problemas del logaritmo discreto y la factorización. Con ello deja de ser segura gran parte de la criptografía clásica que hoy se encuentra en uso. Es interesante el desarrollo de nuevos esquemas criptográficos que mediante el uso de álgebras diferentes puedan ser inmunes a ataques cuánticos.
Al poseer cuatro coordenadas, bien puede pensarse que un cuaternión puede aplicarse al universo, donde la primera coordenada se usa para el tiempo y las tres complejas se usan para el espacio. Pero esto no es directamente así ya que el álgebra cuaterniónica no cumple con algunos aspectos de la teoría de la relatividad de Einstein (Rodriguez Bouza, 2012).
Una aplicación muy extendida de los cuaterniones se da en el campo de la Visión en 3D y computación gráfica, orientada a localización, seguimiento y control automatizado de objetos (Salmerón Quiroz et al., 2009b).

3. NOTACIONES Y OPERACIONES CON CUATERNIONES

Un cuaternión, como se dijo, es un número hipercomplejo $\boldsymbol{q} = w + x.i + y.j + z.k$ donde i, j y k son unidades imaginarias; y w, x, y, z son números reales. También se lo suele denotar $\boldsymbol{q} = (w, \mathbf{v})$, con $\mathbf{v} = (x, y, z)$. Ésta es la notación vectorial, mientras que la notación cartesiana es $\mathbf{q} = (w, x, y, z)$.
La suma y resta de cuaterniones se define componente a componente.
El producto de dos cuaterniones $\boldsymbol{q_1} = (w_1, x_1, y_1, z_1)$ y $\boldsymbol{q_2} = (w_2, x_2, y_2, z_2)$, es el cuaternión $\boldsymbol{q_3} = \boldsymbol{q_1}.\boldsymbol{q_2} = (w_3, x_3, y_3, z_3)$ con

$$\begin{gathered}
w_3 = w_1.w_2 - x_1.x_2 - y_1.y_2 - z_1.z_2 \\
x_3 = w_1.x_2 + x_1.w_2 + y_1.z_2 - z_1.y_2 \\
y_3 = w_1.y_2 - x_1.z_2 + y_1.w_2 + z_1.x_2 \\
z_3 = - w_1.z_2 + x_1.y_2 - y_1.x_2 + z_1.w_2
\end{gathered}$$

Como puede comprobarse fácilmente, este producto no es conmutativo.
Si se representan los cuaterniones con $\boldsymbol{q_1} = (w_1, \mathbf{v_1})$ y $\boldsymbol{q_2} = (w_2, \mathbf{v_2})$, sin dificultad puede demostrarse que el producto es $\boldsymbol{q_3} = (w_1w_2 - \mathbf{v_1}\cdot\mathbf{v_2},\ w_1\mathbf{v_2} + w_2\mathbf{v_1} + \mathbf{v_1}\times\mathbf{v_2})$. Siendo los números reales cuaterniones de la forma $(w, \mathbf{0})$, queda definido también el producto de un real por un cuaternión.
El conjugado del cuaternión $\boldsymbol{q} = (w, \mathbf{v})$ es $\boldsymbol{q}^* = (w, -\mathbf{v})$. La norma de un cuaternión se define como $|\boldsymbol{q}| = (\boldsymbol{q}.\boldsymbol{q}^*)^{1/2} = (w^2 + x^2 + y^2 + z^2)^{1/2}$, y un cuaternión unitario es aquel que tiene norma 1. Todo cuaternión unitario se puede escribir en la forma $\mathbf{q} = (\cos(\alpha/2), \mathbf{v}'.\mathrm{sen}(\alpha/2))$ con $\mathbf{v'}$ versor en R^3.
De esta forma, un cuaternión arbitrario $\mathbf{q} = (w, x, y, z)$ se puede denotar en notación trigonométrica como $\mathbf{q} = |\mathbf{q}|.(\cos(\alpha/2), \mathbf{v}'.\mathrm{sen}(\alpha/2))$ con $\mathbf{v'}$ versor de $\mathbf{v} = (x,y,z)$ y $\alpha = 2 \arccos(w/|\mathbf{q}|)$.
El inverso de un cuaternión es $\boldsymbol{q}^{-1} = \boldsymbol{q}.(1/|\boldsymbol{q}|^2)$ y entonces el cociente entre dos cuaterniones es $\boldsymbol{q_1}/\boldsymbol{q_2} = \boldsymbol{q_1}.\boldsymbol{q_2}^{-1}$.

4. REPRESENTACIÓN ALGEBRAICA DE ROTACIONES

4.1. ROTACIÓN CON CUATERNIONES

Los puntos en el espacio son representados con cuaterniones que tienen parte real nula, llamados cuaterniones puros.
Los cuaterniones unitarios proporcionan una notación matemática para representar las orientaciones y las rotaciones de objetos en tres dimensiones. Un cuaternión unitario se puede denotar $\mathbf{q} = (\cos(\alpha/2), \mathbf{v}'.\mathrm{sen}(\alpha/2))$, siendo $\mathbf{v}'$ un versor en R^3. Así, todo cuaternión unitario define implícitamente una dirección $\mathbf{v'}$ y un ángulo α, que puede asociarse a una rotación alrededor del origen.
Puede demostrarse (Torres del Castillo, 1999) que siendo P un punto del espacio (o cuaternión puro), la transformación

$$T_{\boldsymbol{q}}(P) = \mathbf{q}.P.\mathbf{q}^* \qquad \text{(Ec. 1)}$$

da el punto $P' = T_{\mathbf{q}}(P)$, que es la imagen de P por la rotación con eje $\mathbf{v'}$ un ángulo α.

4.2. ROTACIÓN CON MATRICES DE EULER

Uno de los métodos más conocidos para representar rotaciones es usando los ángulos de Euler. Una matriz de rotación se puede construir a partir de matrices de rotaciones básicas sobre los ejes cartesianos. (Hughes, 2012). Así, el producto $R = R_{(Z,\gamma)}.\ R_{(Y,\beta)}.\ R_{(X,\alpha)}$ de un ángulo α alrededor del eje X, seguida de una rotación de un ángulo β alrededor del eje Y y finalmente una rotación con un ángulo γ alrededor del eje Z, donde

$$R_{(X,\alpha)} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\operatorname{sen}(\alpha) \\ 0 & \operatorname{sen}(\alpha) & \cos(\alpha) \end{bmatrix};\ R_{(Y,\beta)} = \begin{bmatrix} \cos(\beta) & 0 & \operatorname{sen}(\beta) \\ 0 & 1 & 0 \\ -\operatorname{sen}(\beta) & 0 & \cos(\beta) \end{bmatrix};$$

$$R_{(Z,\gamma)} = \begin{bmatrix} \cos(\gamma) & -\operatorname{sen}(\gamma) & 0 \\ \operatorname{sen}(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Este cálculo representa un giro en sentido antihorario del sistema de referencias.
La desventaja más importante de esta representación es la ocurrencia de singularidades debido al *gimbal lock*. Por este motivo, este método no es usado en aplicaciones automáticas.

4.3. MATRIZ DE ROTACIÓN DE OLINDE RODRIGUES

La expresión de Olinde Rodrigues para la matriz de rotación permite calcular la rotación de un punto un ángulo α alrededor de un eje arbitrario en la dirección **u** = (u_x, u_y, u_z) (Serrano et al., 2014).

$$R = \begin{bmatrix} \cos\left(\frac{\alpha}{2}\right) + (1-\cos(\frac{\alpha}{2}))u_x^2 & u_x u_y\left(1-\cos\left(\frac{\alpha}{2}\right)\right) - u_z sen(\alpha) & u_x u_z\left(1-\cos\left(\frac{\alpha}{2}\right)\right) + u_y sen(\alpha) \\ u_x u_y\left(1-\cos\left(\frac{\alpha}{2}\right)\right) + u_z sen(\alpha) & \cos\left(\frac{\alpha}{2}\right) + (1-\cos(\frac{\alpha}{2}))u_y^2 & u_y u_z\left(1-\cos\left(\frac{\alpha}{2}\right)\right) - u_x sen(\alpha) \\ u_x u_z\left(1-\cos\left(\frac{\alpha}{2}\right)\right) - u_y sen(\alpha) & u_y u_z\left(1-\cos\left(\frac{\alpha}{2}\right)\right) + u_x sen(\alpha) & \cos\left(\frac{\alpha}{2}\right) + (1-\cos(\frac{\alpha}{2}))u_z^2 \end{bmatrix}$$

Las coordenadas del punto P’, obtenido de rotar P un ángulo α alrededor del eje cuya dirección viene dada por el vector unitario **u** utilizando es $P' = RP$.

5. EXPERIMENTACIÓN COMPARATIVA

En (Heuer y Chacón, 2015) se ha realizado una comparación de rotaciones mediante cuaterniones y matrices, centrándose en la cantidad de operaciones básicas y el uso de memoria.
En este trabajo, realizamos una experimentación comparativa, focalizándonos en el tiempo que consume cada enfoque en lograr 10000 rotaciones de puntos aleatorios del espacio, alrededor de ciertos vectores y ciertos ángulos. Se comparan el cálculo mediante rotaciones y mediante la matriz de Olinde Rodrigues, ya que ambos métodos parten de los mismos datos (eje de rotación y ángulo).
Se implementaron los métodos en una netbook Acer Aspire One con procesador Intel Atom CPU Z520 a 1,33GHz con 2Gb de memoria RAM, bajo Ubuntu 10.04 LTS con núcleo Linux Debian. Se programó en Python 2.6.
La Tabla 1 muestra los tiempos empleados para ejecutar 10.000 rotaciones de puntos aleatorios alrededor de ciertos vectores y ángulos.
El promedio de realizar 10.000 rotaciones de puntos aleatorios usado matrices emplea un tiempo promedio de 4863 ms, mientras que realizar las mismas operaciones con cuaterniones usa en promedio 1285 ms.

6. CONCLUSIONES

Los cuaterniones se destacan por presentar una notación compacta y numéricamente eficiente. Los resultados experimentales aquí mostrados muestran categóricamente el impacto de estas ventajas: realizar una rotación mediante cuaterniones insume casi 25% del tiempo en comparación con el uso de matrices de rotación, o lo que es similar: el uso de cuaterniones es aproximadamente cuatro veces más rápido que el uso de matrices. Puede entonces, resultar una buena idea, experimentar su uso en aplicaciones donde se requiera un gran volumen de cálculo en tiempo real, como es, por ejemplo, el seguimiento de un objeto por visión artificial o el desplazamiento de dispositivos autónomos.

Tabla 1: Tiempo de cálculo de 10.000 rotaciones usando matrices y cuaterniones**.**

Vector	Angulo	Tiempo (en ms)		Vector	Angulo	Tiempo (en ms)	
		Matrices	Cuaterniones			Matrices	Cuaterniones
(1,0,0)	0°	4057	1048	(1,1,10)	20°	5019	1357
(1,0,0)	30°	4198	1265	(1,1,10)	50°	4916	1277
(1,0,0)	75°	4827	1267	(1,1,10)	95°	4818	1513
(1,0,0)	150°	4683	1168	(1,1,10)	170°	5659	1324
(1,0,0)	225°	4904	1292	(1,1,10)	245°	5112	1236
(2,2,1)	10°	5144	1357	(0,8,1)	30°	4870	1285
(2,2,1)	40°	4978	1369	(0,8,1)	60°	4759	1250
(2,2,1)	85°	5006	1172	(0,8,1)	105°	5017	1184
(2,2,1)	160°	5071	1372	(0,8,1)	180°	4502	1449
(2,2,1)	235°	5037	1267	(0,8,1)	255°	4695	1255

REFERENCIAS

ANAND, PM RUBESH, GAURAV BAJPAI, VIDHYACHARAN BHASKAR. *Real-time symmetric crypto graphy using quaternion julia set*. Int J Comp Sci Network Security 9.3, 20-26 (2009).

BARCALA J.M., FERNANDEZ J.L., ALBERDI J., JIMENEZ J., LAZARO J.C., NAVARRETE J.J.OLLER J.C. *Identification of plastics using wavelets and quaternion numbers*, Measurement, Science and Technology, 15. 371 (2004).

HEUER, G Y CHACÓN, R. *Orientation & Quaternions*, ldc.usb.ve/~alacruz/cursos/ci5321/exposiciones/informe.docx, consultado 15-01.2015

HUGHES, PETER C. *Spacecraft attitude dynamics*. Courier Dover Publications, 2012.

RODRIGUEZ BOUZA, VICTOR. *Sobre los cuaterniones, álgebras de Lie, y matrices de Pauli. Teoría básica y aplicaciones físicas*. (2012).

SALMERON QUIROZ, B, FIGUEROA GARCIA M., GUERRERO CASTELLANOS J., MENDOZA NUÑEZ M., PACHECO MARTINEZ J., BARRETO CORTES S. *Reporte del desarrollo técnico de la Investigación proyecto SIP 20082294 Proyecto Autogeneración de equipo Educativo en Robótica, usando Modelado vía Cuaterniones* (2009a).

SALMERON QUIROZ B. GODIN CHRISTELLE, LESECQ SUZANNE. *Fusión de datos de multicaptores para la captura de movimiento*. AMCA 2009, Congreso Nacional de Control Automatica, México, (2009b).

SANCHEZ-PEÑA, R. S. Y ROBERTO J. ALONSO. *Control de vehículos espaciales*. RIAII 2.3 6-24 (2005).

SERRANO, EDUARDO, RICARDO OSCAR SIRNE, GUILLERMO LA MURA. *Rotaciones, secuencia aeroespacial y cuaterniones Una revisión de las relaciones fundamentales*. Ciencia y Tecnología 1.14 (2014).

SANKAR, VIJAY Y SELVAKUMAR, ARUL LAWRENCE. *Analyse and implement of cryptography with high security using cuaternion*. Int. J. of Innovative Research in Information Security. 1: 2 (2014).

SHOR, PETER W. *Algorithms for quantum computation: discrete logarithms and factoring*. Foundations of Computer Science, 1994 Proceedings., 35th Annual Symposium on. IEEE, (1994).

TORRES DEL CASTILLO G.F. *La representación de rotaciones mediante cuaterniones*. Miscelánea Matemática 29 , 43-50 (1999)