Gravity Compensation with Dual Quaternions

Francisco Jesús ARJONILLA GARCÍA and Yuichi KOBAYASHI

Shizuoka University, Japan

Research question: Can we improve the algebraic framework used in robotics with geometric algebras?

Introduction

Geometric algebras have several advantages over matrices and trigonometry:

- Capable of representing lines, curves, planes, spheres, etc.
- More compact representations
  e.g. Maxwell equations may be represented with only 1 equation in geometric algebra.

In this work we explore using geometric algebra for gravity compensation by proposing a novel representation of centroids with dual-quaternions.

Notation

Name	Notation	Element of	Construction
Scalar	s	$\mathbb{R}$	
Vector	$\vec{v}$	$\mathbb{R}^3$	$\vec{v} = v_x\,\vec{\imath} + v_y\,\vec{\jmath} + v_z\,\vec{k}$
Matrix	M	$\mathbb{R}^{m\times n}$	
Coordinate system	O_i		
Quaternion	$\boldsymbol{q}$	$\mathbb{H} \cong \mathbb{R}^4$	$\boldsymbol{q} = s + \vec{v}$
Dual quaternion	$\boldsymbol{Q}$	$\mathbb{H}^2$	$\boldsymbol{Q} = \boldsymbol{r} + \varepsilon\boldsymbol{p}$

Quaternions are an extension of complex numbers

$$\vec{\imath}^{\,2} = \vec{\jmath}^{\,2} = \vec{k}^{\,2} = \vec{\imath}\,\vec{\jmath}\,\vec{k} = -1$$

points $\boldsymbol{p} = \vec{p}$

rotations $\boldsymbol{r} = s_r + \vec{v}_r = \cos\frac{\theta}{2} + \sin\frac{\theta}{2}\,\vec{r}$

rotation of points $\boldsymbol{p}' = \boldsymbol{r}\,\boldsymbol{p}\,\boldsymbol{r}^*$

Compared to rotation matrices, quaternions are:

- Numerically more stable
- More efficient in many operations
- Smaller memory footprint
- Capable of SLERP (Spherical interpolation)

Efficiency of composition of rotations

Matrix representation

$$\begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} s_{11} & s_{12} & s_{13} \\ s_{21} & s_{22} & s_{23} \\ s_{31} & s_{32} & s_{33} \end{bmatrix}$$

27 multiplications and 18 sums

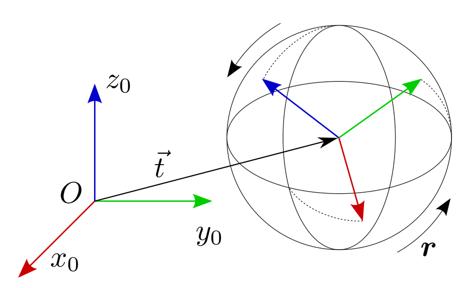
Quaternion representation

$$(r_w, r_x, r_y, r_z)\,(s_w, s_x, s_y, s_z)$$

16 multiplications and 12 sums

Dual quaternions are an extension of quaternions with a dual number $\varepsilon^2 = 0$

$$\boldsymbol{Q} = \boldsymbol{r} + \varepsilon\,\boldsymbol{p}$$



points $\boldsymbol{P} = 1 + \varepsilon\,\vec{p}$

rotations $\boldsymbol{R} = \boldsymbol{r}$

translations $\boldsymbol{T} = 1 + \varepsilon\frac{1}{2}\vec{t}$

rigid-body transformations $\boldsymbol{B} = \boldsymbol{T}\boldsymbol{R} = \boldsymbol{r} + \varepsilon\frac{1}{2}\vec{t}\,\boldsymbol{r}$

transformation of points $\boldsymbol{P}' = \boldsymbol{B}\boldsymbol{P}\boldsymbol{B}^{-*} = 1 + \varepsilon\ \vec{t} + \boldsymbol{r}\vec{p}\,\boldsymbol{r}^*$

composition of transformations $\boldsymbol{B}_{21} = \boldsymbol{B}_2\boldsymbol{B}_1 = \boldsymbol{r}_{21} + \varepsilon\frac{1}{2}\left(\vec{t}_2 + \boldsymbol{r}_2\vec{t}_1\,\boldsymbol{r}_2^*\ \ \boldsymbol{r}_{21}\right)$

The transformation function between links in a solid robot can be separated in fixed and variable transformations for efficiency.

$$\boldsymbol{B}_i = \boldsymbol{N}_i\boldsymbol{J}_i$$

$\boldsymbol{N}_i$: Transformation of link in neutral pose

$$\boldsymbol{J}_i = \cos\frac{q_i}{2} + \sin\frac{q_i}{2}\,\vec{k}$$

Centroid representation

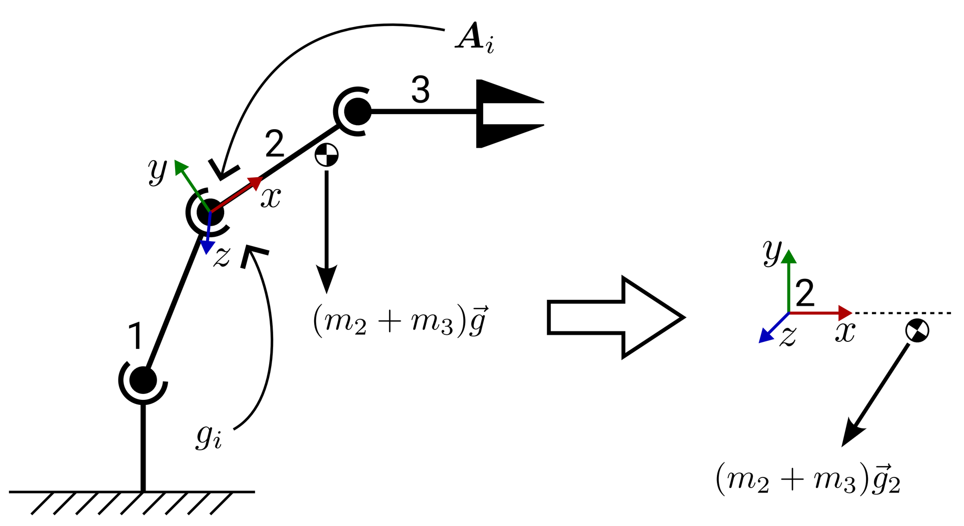
We incorporate the mass of a centroid into the dual-quaternion representation of the location of that centroid.

Proposition: The representation of a centroid C of mass m located at $\boldsymbol{P}_C = 1 + \varepsilon\,\vec{c}$ in dual-quaternion form is

$$\boxed{\boldsymbol{C} = m\boldsymbol{P}_C = m + \varepsilon\, m\vec{c}}$$

➡ It can be proved that centroids in dual-quaternion form are closed under addition and composition.

Streaming calculations in serial robots



For each link:

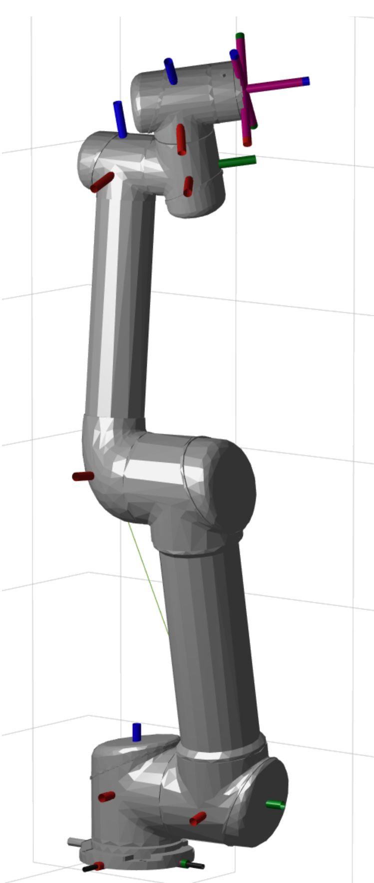
- Transform upstream centroids to the coordinates corresponding to the axis's link.
  $\boldsymbol{A}_{n-1} = \boldsymbol{C}_{n-1} + \boldsymbol{B}_n\boldsymbol{C}_n\boldsymbol{B}_n^{-*}$ $\boldsymbol{A}$: accumulated centroid
- Transform gravity vector to the coordinates corresponding to the axis's link.
  $\vec{g}_0 = \vec{g}$ $\vec{g}_i = \boldsymbol{r}_i^*\vec{g}_{i-1}\boldsymbol{r}_i$

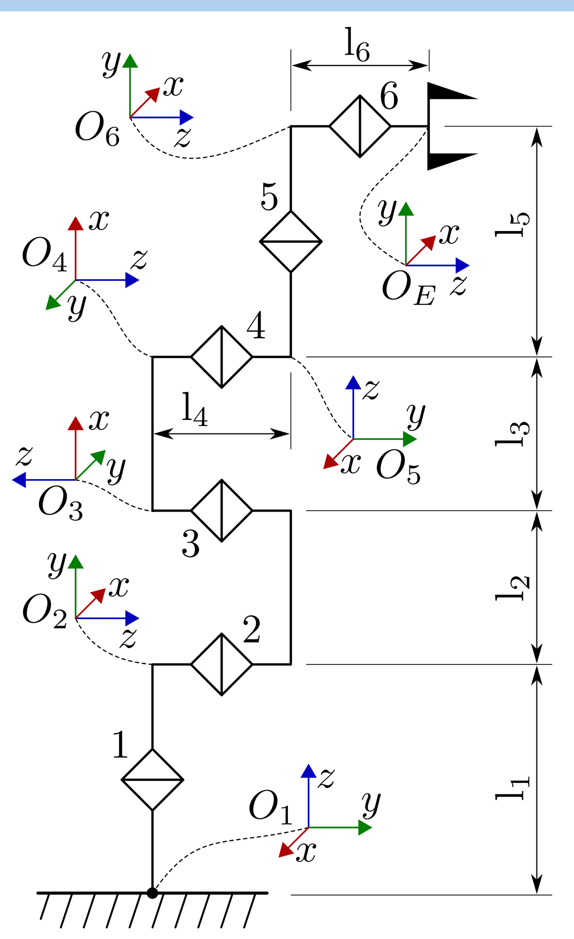
Then, gravity compensation torque or force for axis i is:

Torque (revolute axes): $\tau_i = -m_i\,(\vec{c}_i \times \vec{g}_i)\cdot\vec{k} = -\left(\mathrm{dual}\,(\boldsymbol{C}_i) \times \vec{g}_i\right)\cdot\vec{k}$

Forces (prismatic axes): $f_i = -m_i\vec{g}_i\cdot\vec{k} = -\mathrm{direct}\,(\boldsymbol{C}_i)\,\vec{g}_i\cdot\vec{k}$

Validation





D-H parameters of UR5 robot and axis values for validation

i	d_i [m]	θ_i [rad]	a_i [m]	α_i [rad]	q_i [rad]
1	–	–	–	–	0.1
2	l_1	π	0	$\frac{\pi}{2}$	0.2
3	0	$\frac{\pi}{2}$	l_2	π	0.3
4	0	0	l_3	π	0.4
5	l_4	$\frac{\pi}{2}$	0	$\frac{\pi}{2}$	0.5
6	l_5	π	0	$\frac{\pi}{2}$	0.6
e	l_6	0	0	0	–

Kinematic parameters and masses

i	l_i [m]	m_i [kg]	$\vec{c}_i$ [m]
1	0.089159	3.7	$(0, 0.00193, 0.063549)$
2	0.425	8.393	$(0, 0.2125, 0.11336)$
3	0.39225	2.33	$(0.24225, 0, -0.0265)$
4	0.10915	1.219	$(0.01634, 0, 0.10735)$
5	0.09465	1.219	$(0, 0.01634, 0.09285)$
6	0.0823	0.1879	$(0, 0, 0.081141)$

Rigid-body transformations in neutral pose for each axis

$$\boldsymbol{N}_1 = 1$$

$$\boldsymbol{N}_2 = \frac{\sqrt{2}}{2}\ \ \vec{\imath} + \vec{k}\ \ + \varepsilon\frac{\sqrt{2}}{4}l_1\ \ -1 - \vec{\imath}$$

$$\boldsymbol{N}_3 = \frac{\sqrt{2}}{2}\ \ \vec{\imath} + \vec{\jmath}\ \ + \varepsilon\frac{\sqrt{2}}{4}l_2\ \ -1 - \vec{k}$$

$$\boldsymbol{N}_4 = \vec{\imath} - \varepsilon\frac{1}{2}l_3$$

$$\boldsymbol{N}_5 = \frac{1}{2}\ \ 1 + \vec{\imath} + \vec{\jmath} + \vec{k}\ \ + \varepsilon\frac{1}{4}l_4\ \ -1 - \vec{\imath} + \vec{\jmath} + \vec{k}$$

$$\boldsymbol{N}_6 = \frac{\sqrt{2}}{2}\ \ \vec{\imath} + \vec{k}\ \ + \varepsilon\frac{\sqrt{2}}{4}l_5\ \ -1 - \vec{\imath}$$

Results

i	$\boldsymbol{A}_i$ [kg+kgm]	$\vec{g}_i$ [ms^{-2}]	τ_i [Nm]
1	$17.05 + \varepsilon\,(0.642, 1.336, 6.973)$	$(0, 0, -9.81)$	0
2	$13.35 + \varepsilon\,(0.473, 5.565, 1.328)$	$(-1.95, -9.61, 0)$	−6.29883
3	$4.96 + \varepsilon\,(1.740, -0.043, -0.377)$	$(-9.76, 0.98, 0)$	−1.28212
4	$2.626 + \varepsilon\,(0.151, -0.017, 0.315)$	$(-9.37, 2.90, 0)$	−0.27943
5	$1.407 + \varepsilon\,(0, 0.035, 0.131)$	$(2.54, -1.39, -9.37)$	0.089465
6	$0.188 + \varepsilon(0, 0, 0.015)$	$(-7.39, -6.30, -1.39)$	0

- We validated the proposal by calculating gravity terms for a UR5 robot with six revolute axis.
- These results agree with matlab's `gravityTorque()` and Universal Robot's tools.

Conclusions

This proposal has explored the use of dual quaternions in robotics for gravity compensation.

We have shown that centroids can be represented in dual-quaternion form and that they are compatible with common operations such as coordinate transformation.

This representation of centroids extends the advantages of quaternions to calculation of gravity-compensation terms.

Additionally, this representation facilitates calculations in modular robotics, robots with a high number of links, and robots with complex kinematic trees.

In the future, we hope that the robotics community will benefit from leveraging the potential of geometric algebra.

Promising efforts include extending dual quaternions to calculation of robot dynamics.

Selected references

W. R. Hamilton, On quaternions: on a new system of imaginaries in Algebra, The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science, 25(169), pp. 489–495, 1844.

A. A. Hayat, A. D. Udai and S. K. Saha, Kinematic chain of an industrial robot and its identification for gravity compensation, in International Conference on Machines and Mechanisms, IIT Roorkee, India, 2013.

K. Kanatani, Understanding geometric algebra: Hamilton, Grassman, and Clifford for computer vision and graphics, CRC Press, Okayama, Japan, 2015.

R. Kelly, V. Santibáñez Davila, A. Loría, Control of robot manipulators in joint space, Springer-Verlag, London, 2005.

Mathworks, Matlab Robotics System Toolbox Documentation, https://uk.matworks.com/help/robotics, accessed July, 2024.

E. Pennestri and P. P. Valentini, Dual quaternions as a tool for rigid body motion analysis: a tutorial with an application to biomechanics, The Archive of Mechanical Engineering, 57(2), 2010.

The POS library, URL https://gitlab.com/ninbot/pos, accessed July, 2024.

L. A. Radavelli, A comparative study of the kinematics of robot manipulators by Denavit-Hartenberg and dual quaternion, Mecánica Computacional, 31, pp. 2833–2848, 2012.

B. V. Adorno, Robot kinematic modeling and control based on dual quaternion algebra — Part I: Fundamentals, Hal-01478225, 2017.
V. Arakelian (ed.) Gravity compensation in robotics, Rennes, France, 2022.

E. Bayro-Corrochano, Geometric algebra applications Vol. II: Robot modelling and control, Springer Charm, Guadalajara, Mexico, 2020.

J. Boisclair, P.L. Richard, T. Laliberté and C. Gosselin, Gravity compensation of robotic manipulators using cylindrical Halbach arrays, IEEE/ASME Transactions on Mechatronics, 22(1): 457–464, 2017.

Universal Robots – DH parameters for calculations of kinematics and dynamics. URL https://www.universal-robots.com/articles/ur/application-installation/dh-parameters-for-calculations-of-kinematics-and-dynamics, accessed June, 2024.

Y.R. Chheta, T.M. Joshi, K.K. Gotewal and S.M. Manoah, A review on passive gravity compensation, in International Conference on Electronics, Communication and Aerospace Technology, Coimbatore, India, pp. 184–189, 2017.

X. Wang and H. Zhu, On the comparisons of unit dual quaternion and homogeneous transformation matrix, Advances in Applied Clifford Algebras, 24(1), pp. 213–229, 2014.

W. K. Clifford, Preliminary sketch of biquaternions, Proceedings of the London Mathematical Society, 1871, s1-4(I), pp. 381–395, 1978.