

ОБЩИЕ ЧИСЛЕННЫЕ
МЕТОДЫ

УДК 519.16

АППАРАТ ГЕОМЕТРИЧЕСКОЙ АЛГЕБРЫ И КВАТЕРНИОНОВ
В СИСТЕМАХ СИМВОЛЬНЫХ ВЫЧИСЛЕНИЙ ДЛЯ ОПИСАНИЯ
ВРАЩЕНИЙ В ЕВКЛИДОВОМ ПРОСТРАНСТВЕ¹⁾

© 2023 г. Т. Р. Велиева^{1,*}, М. Н. Геворкян^{1,**}, А. В. Демидова^{1,***},
А. В. Королькова^{1,****}, Д. С. Кулябов^{1,2,*****}

¹ 117198 Москва, ул. Миклухо-Маклая, 6, Российский университет дружбы народов, Россия

² 141980 Дубна, М.о., ул. Жолио-Кюри, 6, Объединенный институт ядерных исследований, Россия

*e-mail: velieva-tr@rudn.ru

**e-mail: gevorgyan-mn@rudn.ru

***e-mail: demidova-av@rudn.ru

****e-mail: korolkova-av@rudn.ru

*****e-mail: kulyabov-ds@rudn.ru

Поступила в редакцию 15.04.2022 г.

Переработанный вариант 15.04.2022 г.

Принята к публикации 17.09.2022 г.

В физических и технических задачах достаточно распространенным в применении математическим аппаратом является тензорный формализм (и его частный случай – векторный формализм). Хотя этот формализм и считается достаточно универсальным и подходящим для описания многих пространств, порой требуется применение специального математического аппарата. Например, задача вращения в трехмерном пространстве достаточно плохо описывается в тензорном представлении, и для ее решения более целесообразно использовать формализм представлений алгебры Клиффорда, в частности, кватернионов и геометрической алгебры. В статье средствами компьютерной алгебры демонстрируется решение задачи вращения в трехмерном пространстве с использованием как формализма кватернионов, так и формализма геометрической алгебры. Показано, что при всей принципиальной схожести формализмов кватернионов и геометрической алгебры последний представляется более наглядным как при проведении вычислений, так и при интерпретации результатов. Библ. 32. Фиг. 1.

Ключевые слова: геометрическая алгебра, кватернионы, компьютерная алгебра, мультивектор, вращения в трехмерном пространстве.

DOI: 10.31857/S0044466923010143, EDN: LEANHI

1. ВВЕДЕНИЕ

Задача описания вращения твердого тела в трехмерном пространстве восходит еще к Эйлеру [1] и актуальна для многих физических и инженерных задач. Основная задача сводится к вращению трехмерного вектора вокруг начала декартовой системы координат. Можно выделить два классических подхода к ее решению. Первый подход опирается на теорию ортогональных операторов и представляет оператор вращения в виде матрицы, а второй подход основан на применении математического аппарата кватернионов.

Матричный подход имеет ряд недостатков, основным из которых является так называемый “эффект складывания рамок” (gimbal lock) [2], который приводит к вычислительной нестабильности. Также в силу теоремы Эйлера любое вращение твердого тела может быть представлено как последовательность не более чем трех вращений вокруг координатных осей, где никакие два вращения подряд не производятся вокруг одной и той же оси координат [3].

Учитывая это утверждение, а также некоммутативность вращений, получаем, что все возможные тройки вращений насчитывают двенадцать комбинаций.

¹⁾ Работа выполнена при финансовой поддержке Программы стратегического академического лидерства РУДН.

Наличие двенадцати комбинаций приводит к тому, что в инженерной или компьютерной системе приходится реализовывать или все двенадцать вариантов, или же выбирать одну конкретную комбинацию. Первый подход очень трудоемкий, поэтому исторический прижился второй вариант, что привело к использованию разных комбинаций в разных областях с разной терминологией.

Приведем три примера. В авиации [3] используется последовательность вращений ZYX (вокруг оси OZ , затем вокруг оси OY и, наконец, вокруг оси OX) и углы носят названия крена, тангажа и рыскания. В астрономии используются несколько последовательностей. Так, последовательность углов Эйлера для орбит небесных тел [3] описывает вращение ZXY и углы именуется восхождением, склонением и широтой. Другая последовательность – ZYX носит название эфемерид [3] с названием углов долгота, широта и направление пути эфемериды. Хотя эта комбинация похожа на ту, которая используется в авиации, но угол поворота вокруг оси OY (широта) берется с отрицательным знаком.

Эти недостатки привели к тому, что в инженерных задачах, в частности в области компьютерной графики, матричный подход был практически вытеснен кватернионным способом описания вращений, который свободен от эффекта складывания рамок и требует задания четырех параметров: трех компонент направляющего вектора оси вращения и угла вращения вокруг этой оси, из которых только три параметра являются независимыми, так как вектор оси вращения должен быть единичным. Заметим, что абсолютно твердое тело с одной закрепленной точкой имеет три степени свободы. Следовательно, для параметризации вращения формально достаточно задать только три параметра.

Хотя кватернионы и лишены недостатков вращений с помощью матриц с вычислительной точки зрения, они имеют некоторые недостатки с точки зрения геометрической интерпретации. Так, например, сложно дать геометрически ясное объяснение удвоению угла вращения.

В данной статье мы проводим сравнительный анализ кватернионного способа описания поворотов с альтернативным (сравнительно новым) способом, который начал развиваться лишь в XXI веке. Речь идет об описании поворотов с помощью алгебры мультивекторов, которая рассматривается в рамках геометрической алгебры. Сравнение проводится с использованием систем компьютерной алгебры.

Геометрическая алгебра основывается на работах Г.Г. Грассмана [4], У.Р. Гамильтона [3] и У.К. Клиффорда [5]. В геометрической алгебре рассматривается конкретная реализация алгебры Клиффорда, основанная на мультивекторах, которая включает в себя реализацию алгебры Грассмана в виде внешней алгебры p -векторов (контравариантных антисимметричных тензоров с операцией внешнего умножения). Алгебра кватернионов также является частным случаем алгебры мультивекторов.

Долгое время работы Клиффорда не привлекали особого внимания физиков и математиков. Алгебра Клиффорда была известна [6], но математический аппарат не был разработан. Д. Хинстейн [7], судя по всему, является первым исследователем, который формализовал разработки Клиффорда в виде современного математического аппарата. Основная волна исследований на эту тему началась уже в XXI веке.

Первоначально алгебра мультивекторов формулировалась с прицелом на использование в физике [8], [9], однако довольно быстро она обрела большую популярность у специалистов по компьютерной графике [10], [11].

В настоящее время сведения из геометрической алгебры входят во многие учебники по компьютерной графике [12], [13], а методы описания вращений с помощью бивекторов могут потенциально вытеснить кватернионы и бикватернионы [14].

В первых двух частях данной статьи авторы конспективно излагают основные формулы математического аппарата кватернионов и дают более подробное изложение алгебры мультивекторов, ограничиваясь только трехмерным пространством. В третьей части приводится конкретный пример поворота вектора на заданный угол с использованием кватернионов и алгебры мультивекторов. Вычисления для случая кватернионов проводятся с помощью пакета компьютерной алгебры SymPy [15] для языка Python [16], а в случае алгебры мультивекторов – с помощью модуля Galgebra [17], основанного на SymPy. Авторы умышленно использовали простой пример, так как он позволяет не отвлекаться на технические детали, и обсудить разницу между подходами, которая в большей степени относится к области теории и геометрической интерпретации.

2. КВАТЕРНИОНЫ

2.1. Определение и основные операции

Кватернионом называют гиперкомплексное число вида: $q = a + bi + cj + dk$, где a, b, c, d — действительные числа, а i, j, k — кватернионные мнимые единицы, определяемые соотношением $i^2 = j^2 = k^2 = ijk = -1$.

Выделяют *скалярную* часть кватерниона $q_0 = a$ и *векторную* часть $\mathbf{q} = bi + cj + dk$, называемую также *чисто мнимой* или просто *чистой*. Каждому чисто мнимому кватерниону можно взаимно однозначно сопоставить вектор в трехмерном пространстве. Кватернион часто записывают в виде $q = q_0 + \mathbf{q}$.

Кватернионы образуют векторное пространство размерностью четыре над полем вещественных чисел. Кватернионы получаются из эллиптических комплексных чисел посредством процедуры Кэли–Диксона [18]. Согласно модели Кэли–Клайна [19]–[21], для проективных пространств задается три типа мероопределения: эллиптическое, параболическое и гиперболическое. Каждому типу мероопределения соответствует свой тип комплексных чисел [22], [23]. Наше евклидово пространство соответствует эллиптическому мероопределению углов, поэтому для его описания и используются эллиптические комплексные числа (это привычные нам комплексные числа). Например галилеево пространство имеет параболическое мероопределение углов, а пространство Минковского — гиперболическое. Для корректного описания этих пространств следует применять соответствующие комплексные числа.

Множество кватернионов часто обозначают через \mathbb{H} в честь ирландского математика У.Р. Гамильтона (1805–1865 г.), разработавшего теорию кватернионов [24]–[26].

Для двух кватернионов $q_1 = a_1 + b_1i + c_1j + d_1k$ и $q_2 = a_2 + b_2i + c_2j + d_2k$ сложение определяется следующим образом:

$$q_1 + q_2 = (a_1 + a_2) + (b_1 + b_2)i + (c_1 + c_2)j + (d_1 + d_2)k.$$

Для умножения кватернионов необходимо составить таблицу умножения для мнимых единиц i, j, k , которая выводится из формулы $i^2 = j^2 = k^2 = ijk = -1$:

$$\begin{array}{c|ccc} & i & j & k \\ \hline i & -1 & k & -j \\ j & -k & -1 & i \\ k & j & -i & -1 \end{array} \quad (1)$$

Так как операция умножения кватернионов в общем случае не коммутативна, следует отметить, что в строке таблицы указан первый сомножитель, а в столбце — второй.

Используя таблицу (1), можно найти результат кватернионного умножения, который удобно выразить через скалярное и векторное произведения:

$$q_1 \cdot q_2 = a_1a_2 - (\mathbf{q}_1, \mathbf{q}_2) + a_2\mathbf{q}_1 + a_1\mathbf{q}_2 + \mathbf{q}_1 \times \mathbf{q}_2. \quad (2)$$

Кватернион q^* называется *сопряженным* к кватерниону $q = q_0 + \mathbf{q} = q_0 + q_1i + q_2j + q_3k$, если $q^* = q_0 - \mathbf{q} = q_0 - q_1i - q_2j - q_3k$. Норма векторной части кватерниона $\|\mathbf{q}\|$ совпадает с нормой вектора \mathbf{q} и вычисляется как $\|\mathbf{q}\| = \sqrt{(\mathbf{q}, \mathbf{q})} = \sqrt{q_1^2 + q_2^2 + q_3^2}$. Модуль кватерниона определяется формулой

$$|q| = \sqrt{qq^*} = \sqrt{q_0^2 + \|\mathbf{q}\|^2} = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2}.$$

Сопряженный кватернион позволяет найти обратный по умножению кватернион q^{-1} и ввести операцию деления кватернионов:

$$q^{-1} = \frac{q^*}{qq^*} = \frac{q^*}{|q|^2}, \quad \frac{q_1}{q_2} = q_1q_2^{-1} = \frac{q_1q_2^*}{|q_2|^2}.$$

2.2. Чистый и единичный кватернионы

Рассмотрим два чистых кватерниона $q_1 = 0 + \mathbf{q}_1$ и $q_2 = 0 + \mathbf{q}_2$ и перемножим их по формуле (2):

$$q_1 \cdot q_2 = (0 + \mathbf{q}_1)(0 + \mathbf{q}_2) = -(\mathbf{q}_1, \mathbf{q}_2) + \mathbf{q}_1 \times \mathbf{q}_2.$$

Кватернионное произведение чистого кватерниона на самого себя дает норму вектора q со знаком минус:

$$qq = q^2 = (0 + \mathbf{q})(0 + \mathbf{q}) = -(\mathbf{q}, \mathbf{q}) + \underbrace{\mathbf{q} \times \mathbf{q}}_{=0} = -(\mathbf{q}, \mathbf{q}) - \mathbf{q}^2.$$

Единичным или нормированным называется кватернион с модулем, равным единице: $|q| = q_0^2 + \|\mathbf{q}\|^2 = 1$, где $q = q_0 + q_1i + q_2j + q_3k$.

Для единичного кватерниона можно записать так называемую “тригонометрическую форму”. Всегда существует такое число $\vartheta \in [0, \pi)$, что $q_0^2 = \cos^2 \vartheta$, $\|\mathbf{q}\|^2 = \sin^2 \vartheta$. Следовательно, $q_0^2 + \|\mathbf{q}\|^2 = \cos^2 \vartheta + \sin^2 \vartheta = 1$ и единичный кватернион записывается в виде

$$q = \cos \vartheta + \mathbf{u} \sin \vartheta = \cos \vartheta + \sin \vartheta u_1 i + \sin \vartheta u_2 j + \sin \vartheta u_3 k, \quad \mathbf{u} = \frac{\mathbf{q}}{\|\mathbf{q}\|}.$$

Единичный чистый кватернион $u = 0 + \mathbf{u}$ обладает свойством эллиптической мнимой единицы [22]:

$$uu = u^2 = (0 + \mathbf{u})(0 + \mathbf{u}) = -\underbrace{(\mathbf{u}, \mathbf{u})}_{\|\mathbf{u}\|^2} + \underbrace{\mathbf{u} \times \mathbf{u}}_0 - \|\mathbf{u}\|^2 = -1,$$

что дает возможность определить экспоненту от единичного чистого кватерниона по аналогии с формулой Эйлера для комплексных чисел:

$$e^{u\vartheta} = \cos \vartheta + \mathbf{u} \sin \vartheta.$$

2.3. Использование кватернионов для вращения

Замечательным свойством кватернионов оказалась возможность использовать их для задания оператора вращения вектора \mathbf{v} вокруг оси с единичным направляющим вектором \mathbf{u} (чистый кватернион $u = 0 + \mathbf{u}$). Результатом вращения является вектор \mathbf{w} ($w = 0 + \mathbf{w}$):

$$w = \exp\left(\frac{\vartheta}{2} \mathbf{u}\right) \mathbf{v} \exp\left(-\frac{\vartheta}{2} \mathbf{u}\right) = q \mathbf{v} q^*, \quad (3)$$

где

$$q = \exp\left(\frac{\vartheta}{2} \mathbf{u}\right) = \cos \frac{\vartheta}{2} + \sin \frac{\vartheta}{2} \mathbf{u} \quad \text{и} \quad q^* = \exp\left(-\frac{\vartheta}{2} \mathbf{u}\right) = \cos \frac{\vartheta}{2} - \sin \frac{\vartheta}{2} \mathbf{u}.$$

Формулу (3) часто называют *сэндвич*-оператором. Доказательство формулы см., например, в работах [1]–[3].

3. АЛГЕБРА МУЛЬТИВЕКТОРОВ В ТРЕХМЕРНОМ ПРОСТРАНСТВЕ

Геометрическая алгебра является реализацией абстрактной алгебры Клиффорда [5], где элементами являются *мультивекторы*, а операцией “умножения” является операция *геометрического умножения*.

Мультивектор является градуированным объектом, представляющим собой линейную комбинацию разложимых p -векторов (кососимметричных ковариантных тензоров). Сами p -векторы совместно с операцией внешнего произведения \wedge являются реализацией абстрактной алгебры Грассмана [4]. Основным отличительным свойством внешнего произведения векторов является свойство *кососимметричности*: $\mathbf{v} \wedge \mathbf{v} = 0$. Более подробно о внешней алгебре см. [9], [12], [27].

Будем рассматривать трехмерное евклидово пространство \mathbb{E} с ортонормированным базисом $\{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$. Пространство E и операция внешнего произведения \wedge порождают три внешние алгебры

ры: алгебру 1-векторов (совпадает с E), алгебру бивекторов $\Lambda^2(E)$ и алгебру тривекторов $\Lambda^3(L)$. Алгебры большого ранга (градуировки, grade) тривиальны и состоят из скалярных нулей. Заметим, что скаляр также считается p -вектором с $p = 0$.

В $\Lambda^2(E)$ существует три базисных бивектора: $\mathbf{e}_{12} = \mathbf{e}_1 \wedge \mathbf{e}_2$, $\mathbf{e}_{13} = \mathbf{e}_1 \wedge \mathbf{e}_3$ и $\mathbf{e}_{23} = \mathbf{e}_2 \wedge \mathbf{e}_3$, и любой бивектор представим в виде:

$$\mathbf{U} = u^{12} \mathbf{e}_1 \wedge \mathbf{e}_2 + u^{23} \mathbf{e}_2 \wedge \mathbf{e}_3 + u^{13} \mathbf{e}_1 \wedge \mathbf{e}_3,$$

где u^{12} , u^{13} и u^{23} — действительные числа, компоненты бивектора. В трехмерном пространстве любой бивектор *разложим*, т.е. может быть представлен как внешнее произведение двух векторов: $\mathbf{U} = \mathbf{a} \wedge \mathbf{b}$. Геометрической интерпретацией бивектора может служить ориентированная площадь параллелограмма, построенного на векторах \mathbf{a} и \mathbf{b} .

В $\Lambda^3(E)$ существует только один базисный тривектор $\mathbf{e}_{123} = \mathbf{e}_1 \wedge \mathbf{e}_2 \wedge \mathbf{e}_3$ и любые другие тривекторы имеют вид: $u^{123} \mathbf{e}_1 \wedge \mathbf{e}_2 \wedge \mathbf{e}_3$. В трехмерном пространстве любой тривектор также разложим и может быть представлен как $\mathbf{a} \wedge \mathbf{b} \wedge \mathbf{c}$, и геометрической интерпретацией для него является ориентированный объем параллелепипеда, построенного на векторах \mathbf{a} , \mathbf{b} и \mathbf{c} .

Операцию *геометрического умножения* для трехмерных векторов \mathbf{u} и \mathbf{v} можно определить конструктивно следующим образом:

$$\mathbf{uv} = (\mathbf{u}, \mathbf{u}) + \mathbf{u} \wedge \mathbf{v},$$

т.е. как сумму скалярного и внешнего произведений. Отметим, что геометрическое произведение обычно никаким знаком не обозначается.

Для ортогонального базиса из определения геометрического умножения выводятся следующие два важных соотношения: $\mathbf{e}_i \mathbf{e}_j = -\mathbf{e}_j \mathbf{e}_i$, $i \neq j$ и $\mathbf{e}_i \mathbf{e}_i = 1$. Также $\mathbf{e}_i \mathbf{e}_j = \mathbf{e}_i \wedge \mathbf{e}_j$ и $\mathbf{e}_i \mathbf{e}_j \mathbf{e}_k = \mathbf{e}_i \wedge \mathbf{e}_j \wedge \mathbf{e}_k$, где $i, j, k = 1, 2, 3$. Это позволяет представить любой мультивектор как линейную комбинацию базисных p -векторов, записанных через геометрическое умножение. Так, в трехмерном пространстве мультивектор в общем случае записывается в следующей форме:

$$\mathbf{U} = u^0 + u^1 \mathbf{e}_1 + u^2 \mathbf{e}_2 + u^3 \mathbf{e}_3 + u^{12} \mathbf{e}_1 \mathbf{e}_2 + u^{23} \mathbf{e}_2 \mathbf{e}_3 + u^{13} \mathbf{e}_1 \mathbf{e}_3 + u^{123} \mathbf{e}_1 \mathbf{e}_2 \mathbf{e}_3.$$

Можно определить обратный относительно геометрического умножения вектор в виде $\mathbf{u}^{-1} = \frac{\mathbf{u}}{\|\mathbf{u}\|^2}$. Легко проверить, что $\mathbf{u} \mathbf{u}^{-1} = 1$. Если мультивектор представлен как произведение двух векторов \mathbf{ab} , то $(\mathbf{ab})^{-1} = \mathbf{b}^{-1} \mathbf{a}^{-1}$.

Базисный тривектор $\mathbf{e}_1 \mathbf{e}_2 \mathbf{e}_3$ обозначим как \mathbf{I} , так как он обладает свойствами эллиптической мнимой единицы [22]:

$$\mathbf{I} = \mathbf{e}_1 \mathbf{e}_2 \mathbf{e}_3 \mathbf{e}_1 \mathbf{e}_2 \mathbf{e}_3 = \mathbf{e}_1 \mathbf{e}_1 \mathbf{e}_2 \mathbf{e}_2 \mathbf{e}_3 \mathbf{e}_3 = -\mathbf{e}_1 \mathbf{e}_1 \mathbf{e}_2 \mathbf{e}_2 \mathbf{e}_3 \mathbf{e}_3 = -1 \Rightarrow \mathbf{I}^2 = -1.$$

Тривектор \mathbf{I} коммутирует со всеми базисными векторами и бивекторами: $\mathbf{I} \mathbf{e}_i = \mathbf{e}_i \mathbf{I}$, где $i = 1, 2, 3$, и $\mathbf{I} \mathbf{e}_i \mathbf{e}_j = \mathbf{e}_i \mathbf{e}_j \mathbf{I}$, где $i, j = 1, 2, 3$, $i < j$. Из этого следует, что произвольный мультивектор \mathbf{U} в трехмерном пространстве коммутирует с \mathbf{I} : $\mathbf{IU} = \mathbf{UI}$.

Также с помощью \mathbf{I} можно определить операцию дополнения (подробнее см. [12]):

$$\bar{\mathbf{u}} = \mathbf{I} \mathbf{u} \quad \text{и} \quad \bar{\mathbf{U}} = -\mathbf{I} \mathbf{U},$$

где \mathbf{u} — вектор, а \mathbf{U} — бивектор. Данная операция позволяет найти дополнение к вектору или бивектору до элемента максимального ранга в данном пространстве. Так, для вектора в трехмерном пространстве это будет бивектор, а для бивектора — вектор.

Если обозначить $\mathbf{i} = \mathbf{e}_1\mathbf{e}_2$, $\mathbf{j} = \mathbf{e}_2\mathbf{e}_3$ и $\mathbf{k} = \mathbf{e}_1\mathbf{e}_3$, то $\mathbf{ijk} = -1$ и таблица умножения бивекторов превратится в таблицу умножения кватернионных мнимых единиц:

$$\begin{array}{c|ccc} & \mathbf{e}_{12} & \mathbf{e}_{13} & \mathbf{e}_{23} \\ \hline \mathbf{e}_{12} & -1 & -\mathbf{e}_{23} & \mathbf{e}_{13} \\ \mathbf{e}_{13} & \mathbf{e}_{23} & -1 & -\mathbf{e}_{12} \\ \mathbf{e}_{23} & -\mathbf{e}_{13} & \mathbf{e}_{12} & -1 \end{array} \Rightarrow \begin{array}{c|ccc} & \mathbf{i} & \mathbf{k} & \mathbf{j} \\ \hline \mathbf{i} & -1 & -\mathbf{j} & \mathbf{k} \\ \mathbf{k} & \mathbf{j} & -1 & -\mathbf{i} \\ \mathbf{j} & -\mathbf{k} & \mathbf{i} & -1 \end{array} \Rightarrow \begin{array}{c|ccc} & \mathbf{i} & \mathbf{j} & \mathbf{k} \\ \hline \mathbf{i} & -1 & \mathbf{k} & -\mathbf{j} \\ \mathbf{j} & -\mathbf{k} & -1 & \mathbf{i} \\ \mathbf{k} & \mathbf{j} & -\mathbf{i} & -1 \end{array}$$

Таким образом в трехмерном пространстве множество мультивекторов вида

$$\mathbf{V} = v^0 + \mathbf{v}_2 = v^0 + v^{12}\mathbf{e}_{12} + v^{13}\mathbf{e}_{13} + v^{23}\mathbf{e}_{23} = v^0 + v^{12}\mathbf{i} + v^{13}\mathbf{k} + v^{23}\mathbf{j},$$

изоморфно множеству кватернионов. Стоит обратить внимание на то, что в двухмерном пространстве мультивекторы такого вида были изоморфны множеству комплексных чисел. Стоило повысить размерность пространства на единицу, как те же мультивекторы стали изоморфны кватернионам. Это фактически соответствует процедуре Кэли–Диксона [18].

3.1. Повороты в пространстве

Рассмотрим два вектора в трехмерном пространстве \mathbf{a} и \mathbf{b} . Угол между ними равен ϑ . Бивектор $\mathbf{a} \wedge \mathbf{b}$ задает плоскость вращения. Мультивектор $\mathbf{R} = \mathbf{ba}$ называется *ротатором* и определяет оператор вращения вектора \mathbf{v} по формуле

$$\mathbf{w} = \mathbf{RvR}^{-1},$$

где $\mathbf{R} = \mathbf{ba} = (\mathbf{a}, \mathbf{b}) - \mathbf{a} \wedge \mathbf{b}$, а \mathbf{w} – вектор, в который перешел вектор \mathbf{v} в результате вращения. У данной формулы есть наглядная интерпретация – двойное отражение вектора \mathbf{v} сначала относительно вектора \mathbf{a} , а затем относительно вектора \mathbf{b} .

Формула упрощается, если использовать нормированный мультивектор \mathbf{R} , т.е. такой, что $\|\mathbf{R}\| = \mathbf{RR}^* = 1$, где сопряженный ротор вычисляется как $\mathbf{R}^* = (\mathbf{a}, \mathbf{b}) + \mathbf{a} \wedge \mathbf{b}$. Тогда $\mathbf{R}^{-1} = \mathbf{R}^*$ и $\mathbf{w} = \mathbf{RvR}^*$.

Рассмотрим произвольный мультивектор \mathbf{U} , состоящий из скалярной и бивекторной частей в трехмерном пространстве:

$$\mathbf{U} = u^0 + \mathbf{a}_2 = u^0 + a^{12}\mathbf{e}_{12} + a^{13}\mathbf{e}_{13} + a^{23}\mathbf{e}_{23},$$

где бивектор помечен числом 2. Сопряженный мультивектор \mathbf{U}^* имеет вид

$$\mathbf{U}^* = u^0 + \mathbf{a}_2^* = u^0 + a^{12}\mathbf{e}_{21} + a^{13}\mathbf{e}_{31} + a^{23}\mathbf{e}_{32} = u^0 - a^{12}\mathbf{e}_{12} - a^{13}\mathbf{e}_{13} - a^{23}\mathbf{e}_{23} = u^0 - \mathbf{a}_2.$$

Вычислим норму мультивектора $\|\mathbf{U}\| = \mathbf{UU}^*$:

$$\|\mathbf{U}\| = \mathbf{UU}^* = (u^0)^2 + (a^{12})^2 + (a^{13})^2 + (a^{23})^2.$$

Если предположить, что мультивектор \mathbf{U} имеет единичную норму, то без потери общности его можно представить в тригонометрическом виде:

$$\mathbf{U} = \cos \vartheta + \sin \vartheta (u^{12}\mathbf{e}_{21} + u^{13}\mathbf{e}_{31} + u^{23}\mathbf{e}_{32}) = \cos \vartheta + \sin \vartheta \mathbf{u}_2,$$

где $(u^{12})^2 + (u^{13})^2 + (u^{23})^2 = 1$. По аналогии с формулой Эйлера для комплексных чисел его можно записать и в экспоненциальном виде, определив экспоненту от бивектора следующим образом:

$$\exp\left(\vartheta \frac{\mathbf{u}}{2}\right) = \cos \vartheta + \sin \vartheta \frac{\mathbf{u}}{2}.$$

Ротор \mathbf{R} также представим в тригонометрическом (или экспоненциальном) виде, если $\|\mathbf{a}\| = \|\mathbf{b}\| = 1$:

$$\mathbf{R} = (\mathbf{a}, \mathbf{b}) - \mathbf{a} \wedge \mathbf{b} = \exp\left(\frac{\vartheta}{2} \frac{\mathbf{u}}{2}\right) = \cos \frac{\vartheta}{2} - \sin \frac{\vartheta}{2} \frac{\mathbf{u}}{2},$$

где $(\mathbf{a}, \mathbf{b}) = \|\mathbf{a}\| \|\mathbf{b}\| \cos \frac{\vartheta}{2}$, а компоненты бивектора \mathbf{u} связаны с бивектором $\mathbf{a} \wedge \mathbf{b}$ следующим образом:

$$\begin{aligned} a^1 b^2 - a^2 b^1 &= \sin \frac{\vartheta}{2} u^{12}, \\ a^1 b^3 - a^3 b^1 &= \sin \frac{\vartheta}{2} u^{13}, \\ a^2 b^3 - a^3 b^2 &= \sin \frac{\vartheta}{2} u^{23}. \end{aligned}$$

Если задана ось вращения с направляющим вектором \mathbf{u} , то бивектор, задающий плоскость вращения, можно задать, найдя правое или левое дополнение к вектору \mathbf{u} :

$$\bar{\mathbf{u}} = u^1 \bar{\mathbf{e}}_1 + u^2 \bar{\mathbf{e}}_2 + u^3 \bar{\mathbf{e}}_3 = u^1 \mathbf{e}_{23} - u^2 \mathbf{e}_{13} + u^3 \mathbf{e}_{12}.$$

4. ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ВРАЩЕНИЙ С ПОМОЩЬЮ КОМПЬЮТЕРНОЙ АЛГЕБРЫ

4.1. Описание существующих программных модулей

Рассмотрим программную реализацию вращений с помощью кватернионов и мультивекторов. Для этого используем систему компьютерной алгебры SymPy [15].

Для языка Python существует несколько реализаций кватернионов. Можно выделить два наиболее проработанных модуля: `Quaternions` [28] и `pyquaternion` [29]. Оба этих модуля ориентированы на численные вычисления. Первый из них тесно интегрирован с библиотекой NumPy и поддерживает программный интерфейс этой библиотеки. Использование NumPy позволяет повысить производительность и легко преобразовывать одномерные массивы из четырех элементов в кватернионы и обратно.

Модуль `pyquaternion` также реализует численные вычисления для кватернионов, но реализован без использования внешних библиотек, что делает его более компактным, однако менее производительным в случае высокой интенсивности вычислений.

Для символьных вычислений с кватернионами существует встроенный в SymPy подмодуль `sympy.algebras`, в котором реализован класс `Quaternion`. В нем реализованы все алгебраические операции с кватернионами, включая нахождение модуля, нормы чистого кватерниона, сопряженного кватерниона и ряд других простых операций.

Мультивекторы и операции геометрической алгебры определены в модуле `Galgebra` [17], который необходимо установить отдельно. В своей работе он использует SymPy и хорошо интегрируется с функциями из него. При использовании оболочки Jupyter Notebook или Jupyter Lab выдача результата вычислений будет осуществляться в формате LaTeX и выглядеть как обычная формула, поэтому в данной статье мы приводим результаты работы функций сразу в виде формул.

Также можно указать на смешанный, символьно-численный подход к описанию формализма геометрической алгебры, реализованный на языке Julia [30].

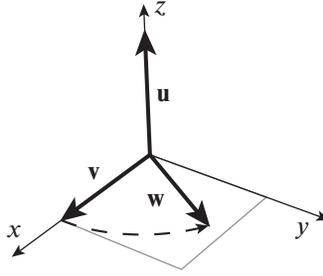
4.2. Пример вращения с помощью кватернионов

Рассмотрим решение задачи поворота вектора $\mathbf{v} = (1, 0, 0)^T$ на угол $\frac{\pi}{3}$ вокруг оси Oz [3]. Поворот изображен на фиг. 1.

Пример сознательно выбран так, чтобы можно было проверить правильность решения визуально, так как результат поворота интуитивно очевиден.

Вектору \mathbf{v} будет соответствовать чистый кватернион $v = 0 + i$. Единичный кватернион поворота q записывается в виде

$$q = \cos \frac{\pi}{6} + \mathbf{u} \sin \frac{\pi \sqrt{3}}{6} + \frac{1}{2} \mathbf{u},$$



Фиг. 1. Поворот $v = e_x$ вокруг Oz на угол, равный 60° .

где векторная часть \mathbf{u} является чистым кватернионом, задающим ось вращения $u = 0 + e_z = 0i + 0j + 1k = k$, т.е. $u = k$. Следовательно,

$$q = \frac{\sqrt{3}}{2} + \frac{1}{2}k \quad \text{и} \quad q^* = \frac{\sqrt{3}}{2} - \frac{1}{2}k.$$

Для использования класса Quaternion импортируем необходимые модули:

```
import sympy as sp
from sympy.algebras import Quaternion
```

Далее зададим угол ϑ , равный половине желаемого угла поворота, ось вращения \mathbf{u} и вектор \mathbf{v} в виде кватернионов:

```
p = sp.pi
q = p/6
u = Quaternion(0, 0, 0, 1).normalize()
v = Quaternion(0, 1, 0, 0).
```

Метод `normalize` здесь избыточен, но если бы мы задали ось вращения не в виде единичного вектора, то данный метод позволил бы автоматически нормировать кватернион для дальнейшего использования.

Далее зададим кватернион q . Можно использовать тригонометрический вид:

```
q = sp.cos(q) + sp.sin(q)*u
```

или же экспоненциальный $q = \exp(\vartheta \mathbf{u})$, как в формуле (3):

```
q = (q*u).exp()
```

Для вычисления результатов вращения также есть две опции. Первая опция – использование встроенного метода `rotate_point`, который принимает вращаемый вектор в виде кортежа (аргумент `pin`), а не кватерниона, и выдает результат также в виде кортежа

```
Quaternion.rotate_point(pin=(1, 0, 0), r=q)
```

$$\left(\frac{1}{2}, \frac{\sqrt{3}}{2}, 0\right).$$

Вторая опция – непосредственное использование формулы (3):

```
q*v*q.conjugate()
```

$$\frac{1}{2}i + \frac{\sqrt{3}}{2}j.$$

В результате применения формулы (3) мы получим из кватерниона $v = i$ кватернион $w = \frac{1}{2}i + \frac{\sqrt{3}}{2}j$, которому соответствует вектор $\mathbf{w} = (1/2, \sqrt{3}/2, 0)$. Из кватерниона можно извлечь ось вращения и угол вращения с помощью соответствующих методов:

```
q.to_axis_angle()
```

$$\left((0, 0, 1), \frac{\pi}{3}\right),$$

```
q.to_rotation_matrix()
```

$$\begin{pmatrix} \frac{1}{2} & -\frac{\sqrt{3}}{2} & 0 \\ \frac{\sqrt{3}}{2} & \frac{1}{2} & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Отметим также, что хотя мы задавали конкретные численные значения, они обрабатывались в символьном виде. Все числовые значения можно заменить на буквенные, используя функцию `SymPy sp.symbols`.

4.3. Краткое описание модуля *Galgebra*

Прежде чем переходить к вычислению поворотов с помощью компьютерной алгебры, опишем основные функции модуля *Galgebra* [17]. Так как *Galgebra* использует символьные вычисления, то для его работы нам понадобится `SymPy`:

```
import sympy as sp
from galgebra.ga import Ga
```

Будем предполагать, что работа ведется в оболочке *Jupyter Notebook* (или *Lab*). В этой оболочке весь вывод команд представляется в виде формул в формате *LaTeX*, поэтому непосредственно после примеров кода мы будем размещать отображаемые оболочкой формулы.

Работа с *Galgebra* начинается с задания линейного пространства и метрики на нем:

```
xy = sp.symbols('1 2', real=True)
o2d = Ga('e', g= [1, 1], coords=xy)
```

Здесь сначала мы определяем символы для индексов базисных векторов. В данном случае задается двумерное пространство, поэтому используются символы 1 и 2. Далее создается объект класса *Ga*, конструктор которого принимает первым аргументом символ для обозначения базисных векторов, а вторым аргументом — диагональную метрику. Отметим, что мы задали евклидову ортонормированную метрику, но можно также задать и метрику пространства Минковского: `g=[1, -1]`.

После создания объекта *Ga* можно приступать к созданию объектов типа мультивектор, используя функцию `mv`, где первый аргумент указывает символ для компонент, а второй — ранг объекта:

```
a_scalar = o2d.mv('a', 0)
a_vect = o2d.mv('a', 1)
a_bivect = o2d.mv('a', 2)
a_scalar, a_vect, a_bivect
```

$$(a, a^1 e_1 + a^2 e_2, a^{12} e_1 \wedge e_2).$$

Определены операции скалярного `|`, внешнего `^` и геометрического `*` умножений:

```
a_scalar ^ a_vect, a_vect * a_bivect
```

$$(aa^1 e_1 + aa^2 e_2, a^{12} a^2 e_1 - a^1 a^{12} e_2).$$

Можно создать и сложный мультивектор общего вида:

```
u = o2d.mv('u', 0) + o2d.mv('u', 1) + o2d.mv('u', 2)
```

$$u + u^1 e_1 + u^2 e_2 + u^{12} e_1 \wedge e_2.$$

В данном модуле также определен ряд операций, относящихся к анализу в рамках геометрической алгебры, т.е. есть возможность задавать поля векторов, бивекторов и т.д.

4.4. Пример вращения с помощью мультивектора

Рассмотрим тот же самый пример, но осуществим вращение с помощью мультивектора **R**.

Прежде всего импортируем необходимые модули:

```
import sympy as sp
from galgebra.ga import Ga
```

и зададим базис евклидова пространства, создав объект класса GA:

```
xyz = sp.symbols('1 2 3', real=True)
o3d = Ga('e_1 e_2 e_3', g= [1, 1, 1], coords=xyz).
```

Сначала зададим символы, которые будут служить индексами базисных векторов, а затем зададим евклидово пространство, указывая в аргументе *g* диагональный метрический тензор, обозначения для базисных векторов и обозначения для индексов в аргументе *coords*. Эти же индексы будут использоваться в компонентах векторов, бивекторов и тривекторов.

Затем присвоим отдельным переменным значение базисных векторов и базисного тривектора e_{123} :

```
e1, e2, e3 = o3d.mv()
I = e1*e2*e3.
```

Далее зададим угол, равный половине желаемого угла вращения, ось вращения и вращаемый вектор:

```
p = sp.pi; q = p/6; u = e3; v = e1.
```

Так как ротор \mathbf{R} является бивектором, то для его вычисления находим дополнение к вектору \mathbf{u} и бивектор e_{12} , что согласуется с геометрическим представлением, так как e_{12} интерпретируется как плоскость вращения Oxy . Заметим, что в модуле *Galgebra* не реализована операция дополнения в явном виде, поэтому используем умножение на \mathbf{I} :

```
U = I*u
```

Осталось составить ротор \mathbf{R} , который будет иметь вид $\sqrt{3}/2 - e_{12}/2$:

```
R = sp.cos(q) - sp.sin(q)*U
```

и применить формулу (3.1):

```
R*v*R.inv()
```

$$\frac{1}{2}\mathbf{e}_1 + \frac{\sqrt{3}}{2}\mathbf{e}_2.$$

Мы также можем проверить нормированность мультивектора \mathbf{R} и вычислить обратный и сопряженный мультивекторы:

```
R.norm(), R.inv(), ~R
```

$$1, \frac{\sqrt{3}}{2} + \frac{1}{2}\mathbf{e}_{12}, \frac{\sqrt{3}}{2} + \frac{1}{2}\mathbf{e}_{12}.$$

Продемонстрированный пример реализации вращения является несколько упрощенным, но на наш взгляд он хорошо иллюстрирует особенности каждого формализма.

5. ЗАКЛЮЧЕНИЕ

В заключение укажем преимущества мультивекторного способа над кватернионным.

Во введении мы уже упомянули, что описание вращений с помощью матриц имеет ряд недостатков, которые как раз и устраняют кватернионы, а именно:

- нет явления складывания рамок и алгоритм численно стабилен;
- исходными данными для осуществления поворота являются ось вращения и угол вращения вокруг этой оси, что повышает наглядность;
- требуется меньше вычислений по сравнению с матричным подходом.

Недостатки кватернионного подхода касаются геометрической интерпретации и невозможности применять его к пространствам, отличным от трехмерного:

- кватернион, являясь четырехмерным объектом, используется для описания трехмерного вектора (с помощью чистого кватерниона);
- кватернион выступает в двух ипостасях — как вращаемый вектор и как “вращатель”;
- есть сложности в интерпретации угла вращения ϑ , так как происходит его удвоение — в формуле задается угол ϑ , а вращение происходит на угол 2ϑ .

При мультивекторном подходе сохраняются преимущества кватернионного подхода с одновременным устранением его недостатков:

- наглядная геометрическая интерпретация всех алгебраических формул:

- векторы \mathbf{a} и \mathbf{b} задают бивектор $\mathbf{a} \wedge \mathbf{b}$, который интерпретируется как параллелограмм (плоскость);
- вращение представляется как композиция двух отражений: сначала относительно прямой с направляющим вектором \mathbf{a} , а затем относительно прямой с направляющим вектором \mathbf{b} ;
- эта интерпретация позволяет легко объяснить удвоение угла поворота ϑ , так как это угол между векторами \mathbf{a} и \mathbf{b} ;
- исходно задается плоскость вращения, но не составляет труда найти ось вращения, применив операцию правого дополнения; можно также задать ось вращения и найти плоскость, также применив операцию дополнения; при использовании кватернионов нет способа моделировать плоскость вращения;
- нет необходимости представлять вектор объектами другого пространства; вращается вектор (элемент линейного пространства), а вращателем выступает мультивектор специального вида (изоморфный кватернионам).

Мы провели вычисления с помощью кватернионов и мультивекторов. Оба формализма являются реализациями одной и той же алгебры Клиффорда [31]. При этом мы не рассматривали еще одну популярную реализацию алгебры Клиффорда – спиноры [32]. Каждая из реализаций имеет свою специфику, в том числе и уровень сложности формализма для исследователя, где спинорный формализм является самым сложным, а формализм геометрической алгебры – самым простым.

Применение различных представлений алгебры Клиффорда к научно-техническим расчетам имеет давнюю историю, однако для стороннего наблюдателя эти манипуляции выглядят несколько экзотично. Мы надеемся, что наша работа послужит делу популяризации такого рода формализмов, особенно формализма геометрической алгебры.

СПИСОК ЛИТЕРАТУРЫ

1. *Kanatani K.* 3D Rotations: Parameter Computation and Lie Algebra based Optimization. Chapman and Hall/CRC, 2020.
2. *Vince J.* Rotation transforms for computer graphics. London : Springer-Verlag, 2011.
3. *Kuipers J.B.* Quaternions and rotation sequences. Princeton University Press, 1999.
4. *Grassmann H.G.* Die Mechanik nach den Principien der Ausdehnungslehre // *Mathematische Annalen.* 1877. Bd. 12. S. 222–240.
5. *Clifford W.K.* Applications of Grassmann’s Extensive Algebra // *American Journal of Mathematics.* 1878. V. 1. № 4. P. 350–358.
6. *Казанова Г.* Векторная алгебра / Под ред. М.К. Поливанов. Современная математика. М.: Мир, 1979. 120 с.
7. *Hestenes D., Sobczyk G.* Clifford Algebra to Geometric Calculus: A Unified Language for Mathematics and Physics. Fundamental Theories of Physics. Netherlands Springer, 1987.
8. *Delanghe R., Sommen F., Soucek V.* Clifford algebra and spinor-valued functions. Mathematics and Its Applications. Netherlands: Kluwer Academic Publishers, 1992.
9. *Doran C., Lasenby A.* Geometric Algebra for Physicists. Morgan Kaufmann Publishers, 2003.
10. *Dorst L., Fontijne D., Mann S.* Geometric algebra for computer science. The Morgan Kaufmann Series in Computer Graphics. Morgan Kaufmann, 2007.
11. *Vince J.* Geometric algebra for computer graphics. Springer-Verlag, 2008.
12. *Lengyel E.* Mathematics. Lincoln, California : Terathon Software LLC, 2016. Vol. 1. 195 p.
13. *Kanatani K.* Understanding Geometric Algebra. Taylor and Francis Group/CRC, 2015.
14. *ten Bosch M.* Let’s remove Quaternions from every 3D Engine. Access mode: <https://marctenbosch.com/quaternions/>.
15. *SymPy.* 2022. Access mode: <http://www.sympy.org/ru/index.html>.
16. *Van Rossum G., Drake F.L.* Python 3 Reference Manual. Scotts Valley, CA: CreateSpace, 2009.
17. GAlgebra – Symbolic Geometric Algebra/Calculus package for SymPy. 2022. Access mode: <https://galgebra.readthedocs.io/en/latest/index.html>.
18. *Dickson L.E.* On Quaternions and Their Generalization and the History of the Eight Square Theorem // *The Annals of Mathematics.* 1919. 3. V. 20. № 3. P. 155–171.
19. *Cayley A.* IV. A sixth memoir upon quantic // *Philosophical Transactions of the Royal Society of London.* 1859. 1. V. 149. P. 61–90.

20. *Klein F.C.* Ueber die sogenannte Nicht-Euklidische Geometrie // Gauß und die Anfänge der nicht-euklidischen Geometrie. Wien: Springer-Verlag Wien, 1985. Bd. 4 von Teubner-Archiv zur Mathematik. S. 224–238.
21. *Klein F.C.* A comparative review of recent researches in geometry // Bulletin of the American Mathematical Society. 1893. V. 2. № 10. P. 215–249. 0807.3161.
22. *Яглом И.М.* Комплексные числа и их применение в геометрии // Математика, ее преподавание, приложения и история. 1961. Т. 6 из Математическое просвещение, сер. 2. С. 61–106.
23. *Gevorkyan M.N., Korolkova A.V., Kulyabov D.S.* Approaches to the implementation of generalized complex numbers in the Julia language // Workshop on information technology and scientific computing in the framework of the X International Conference Information and Telecommunication Technologies and Mathematical Modeling of High-Tech Systems (ИТТММ-2020) / Ed. by D.S. Kulyabov, K.E. Samouylov, L.A. Sevastianov. Vol. 2639 of CEUR Workshop Proceedings. Aachen, 2020. 4. P. 141–157. arXiv: 2007.09737.
24. *Митюшов Е.А., Мисюра Н.Е., Берестова С.А.* К 175-летию открытия кватернионов // Вестник Удмуртского университета. Матем. Мех. Компьют. науки. Т. 28. С. 611–617.
25. *Hamilton W.R.* On a New Species of Imaginary Quantities, Connected with the Theory of Quaternions // Proceedings of the Royal Irish Academy (1836-1869). 1840. V. 2. P. 424–434.
26. *Конвей Д.Х., Смит Д.А.* О кватернионах и октавах / Под ред. В.В. Доценко. М.: МЦНМО, 2009. 184 с.
27. *Кострикин А.И.* Линейная алгебра. М. : МЦНМО, 2009. Т. 2. 368 с.
28. Quaternions in numpy. 2022. Access mode: <https://quaternion.readthedocs.io>.
29. Pyquaternion. 2022. Access mode: <https://kieranwynn.github.io/pyquaternion/>.
30. *Геворкян М.Н., Демидова А.В., Велиева Т.Р., Королькова А.В., Кулябов Д.С.* Аналитико-численная реализация алгебры поливекторов на языке Julia // Программирование. 2022. № 1. С. 54–64.
31. *Пенроуз Р., Риндлер В.* Спиноры и пространство-время. М. : Мир, 1988. Т. 2. 573 с.
32. *Пенроуз Р., Риндлер В.* Спиноры и пространство-время. М. : Мир, 1987. Т. 1. 527 с.