



Международный журнал информационных технологий и энергоэффективности

Сайт журнала:

<http://www.openaccessscience.ru/index.php/ijcse/>



УДК 004.912

ЭКСПЕРИМЕНТАЛЬНОЕ СРАВНЕНИЕ АЛГОРИТМОВ ОПРЕДЕЛЕНИЯ ОРИЕНТАЦИИ ДЛЯ МУЛЬТИКОПТЕРОВ

Сергеенков Д.Д.

Филиал федерального государственного бюджетного образовательного учреждения высшего образования «Национальный исследовательский университет МЭИ» в г. Смоленске, Россия (214013, г. Смоленск, Энергетический проезд, дом 1); e-mail: dumohsmol@yandex.ru

Статья посвящена экспериментальному сравнению трёх алгоритмов определения ориентации мультикоптера: двух вариантов комплементарного фильтра (фильтр Махони, фильтр Маджвика), а также фильтра Калмана. Все алгоритмы используют два датчика: трёхосевой датчик угловой скорости и трёхосевой датчик линейного ускорения. Представлено и описано испытательное устройство, а также план эксперимента. По результатам эксперимента, для каждого алгоритма произведена оценка отклонения расчетной ориентации от реального положения.

Ключевые слова: ориентация мультикоптеров, фильтр Махони, фильтр Маджвика, фильтр Калмана.

EXPERIMENTAL COMPARISON OF DEFINITION ALGORITHMS ORIENTATIONS FOR MULTICOPTERS

Sergeenkov D. D.

Smolensk Branch of Federal state budgetary educational institution of higher education "National research University Moscow power engineering Institute", Russia (214013, Smolensk, Energeticheski proezd, 1); e-mail: dumohsmol@yandex.ru

The article is devoted to an experimental comparison of three algorithms for determining the orientation of a multicopter: two variants of a complementary filter (Mahony filter, Madgwick filter), and Kalman filter. All algorithms utilize two sensors: a three-axis angular velocity sensor and a three-axis linear acceleration sensor. A test device was presented and described, with an experiment plan. According to the results of the experiment, deviation of orientation estimate from the actual position was calculated for each algorithm.

Keywords: multicopter orientation, Mahoney filter, Majvik filter, Kalman filter.

В настоящее время началось бурное развитие нового класса миниатюрных беспилотных летательных аппаратов – мультироторных БПЛА или мультикоптеров. Квадрокоптеры, гексакоптеры и прочие мультироторные аппараты отличаются высокой манёвренностью, лёгкостью в управлении, а также простой и надёжной конструкцией: единственными движущимися элементами являются моторы с пропеллерами. Маневрируют такие аппараты за счёт изменения скорости моторов.

Одним из недостатков мультироторных систем является отсутствие устойчивости – мультикоптеры не способны возвращаться в нейтральное положение после устранения возмущающих сил [1, 28]. Этот недостаток приводит к потребности в системе активной стабилизации – модулю, который будет создавать «искусственное» сопротивление против внешних возмущений.

Самый простой вариант такой системы – алгоритм стабилизации по угловой скорости. Для реализации этой системы необходим лишь один прибор – трехосевой датчик угловых скоростей. Однако при стабилизации исключительно по угловой скорости аппарат не будет обладать информацией о своей ориентации относительно поверхности Земли. То есть, автоматически принять горизонтальное положение мультикоптер не сможет – его должен выровнять человек.

Более продвинутым подходом является стабилизация по абсолютному углу – в этом случае аппарат получает способность автоматически удерживать заданную ориентацию относительно поверхности Земли. Очевидный вариант реализации такой системы – интегрировать показания датчика угловой скорости, а затем использовать полученные углы в системе автоматического регулирования. Однако на практике такая реализация работает неудовлетворительно: датчик угловой скорости обладает рядом погрешностей, из-за которых накапливается ошибка интегрирования. Ситуация усугубляется и тем, что в малых мультикоптерах используются дешевые и миниатюрные микроэлектромеханические датчики, которые не обладают высокой точностью – то есть ошибка интегрирования накапливается быстро (за минуты). Решение проблемы – дополнить датчик угловой скорости дополнительными приборами, и объединить их показания для надёжного определения ориентации, то есть решить задачу комплексирования показаний датчиков (в англоязычной литературе – Sensor Fusion).

Комплексирование показаний датчиков – нетривиальная задача, так как датчики измеряют разные физические величины и обладают различающимися особенностями и погрешностями.

В данной статье проведено сравнение трёх алгоритмов комплексирования датчиков, предназначенных для определения ориентации летательных аппаратов:

- фильтр Махони;
- фильтр Маджвика;
- фильтр Калмана.

Все алгоритмы рассмотрены в варианте для двух датчиков:

- трехосевого датчика угловой скорости;
- трехосевого датчика линейного ускорения.

Датчик угловой скорости используется в качестве основного – его показания наиболее точны, однако результаты их интегрирования подвержены дрейфу. Датчик линейного ускорения – шумный и крайне чувствительный к вибрациям, однако он способен определять ускорение свободного падения – его вектор всегда расположен перпендикулярно поверхности Земли и не подвержен дрейфу.

Фильтр Махони

Этот алгоритм предложен австралийским математиком Робертом Махони вместе с рядом его коллег [2]. Фильтр Махони представляет собой вариант комплементарного фильтра, адаптированный для решения задачи определения ориентации.

В оригинальной статье рассмотрена реализация фильтра с использованием матриц направляющих косинусов, однако наибольшее распространение получил модифицированный вариант с использованием алгебры кватернионов.

Алгоритм работы фильтра состоит из следующих этапов:

1. Сформировать кватернион Q , описывающий поворот системы координат аппарата относительно мировых координат. Изначально предполагается, что координаты аппарата совпадают с мировыми:

$$Q = \langle 1; 0; 0; 0 \rangle.$$

2. Получить показания датчика угловой скорости: вектор

$$\bar{\omega} = (\omega_\psi, \omega_\theta, \omega_\phi),$$

где $\omega_\psi, \omega_\theta, \omega_\phi$ – угловые скорости вокруг осей рыскания, крена и тангажа соответственно.

3. Получить показания датчика линейного ускорения

$$\bar{a} = (a_\psi, a_\theta, a_\phi),$$

где, a_ψ, a_θ, a_ϕ – проекции вектора кажущегося ускорения на оси рыскания, крена и тангажа аппарата.

4. Проверить компоненты a_ψ, a_θ, a_ϕ на корректность – хотя бы один из них не должен быть равен нулю.

5. Рассчитать модуль вектора \bar{a} :

$$|\bar{a}| = \sqrt{a_\psi^2, a_\theta^2, a_\phi^2}$$

6. Проверить модуль на корректность:

$$0.9 < |\bar{a}| < 1.1$$

Если модуль выходит за данный диапазон – показания датчика признаются некорректными, так как содержат кроме ускорения свободного падения ускорения иного рода.

7. Нормализовать вектор:

$$\bar{a}_n = \bar{a}/|\bar{a}|$$

8. Вектор ускорения свободного падения в мировых координатах всегда имеет вид $\bar{b}_w = (0; 0; 1)$. Совершим переход из мировых координат в координаты аппарата. Для этого используется умножение на кватернион Q и обратный ему:

$$\bar{b} = Q^{-1} \circ \bar{b}_w \circ Q$$

9. Если бы кватернион Q точно описывал поворот системы координат аппарата относительно мировых координат, то вектор \bar{b} совпал бы с \bar{a} . Однако, Q обновляется, в основном, за счёт интегрирования показаний датчика угловых скоростей – то есть в нём постепенно накапливается ошибка. По этой причине, векторы \bar{a} и \bar{b} могут не совпасть и будет существовать ненулевое векторное произведение:

$$\bar{c} = \bar{a} \times \bar{b}$$

10. Вектор \bar{c} описывает вращение, его можно использовать как поправку к вектору $\bar{\omega}$:

$$\bar{\omega}_c = \bar{\omega} + p\bar{c},$$

где p – пропорциональный коэффициент. От его величины зависит насколько «агрессивно» поправка от датчика ускорения будет применяться к показаниям датчика угловой скорости.

11. Сформировать вектор d , содержащий приращение угла относительно предыдущего опроса датчика.

$$\bar{d} = (d_\psi; d_\theta; d_\phi) = \Delta t \cdot \bar{\omega}_c,$$

где Δt – интервал времени между опросами датчика.

Это приращение включает в себя корректирующий элемент от датчика ускорения.

12. Сформировать кватернион

$$Q_g = EulerToQuat(d_\psi; d_\theta; d_\phi),$$

где *EulerToQuat* – функция перехода от углов Эйлера к кватерниону, описывающему поворот на эти углы, Δt – интервал времени между опросами датчика. Кватернион Q_g описывает поворот, который совершил аппарат относительно предыдущего опроса датчика.

13. Сформировать новое значение Q :

$$Q = Q \circ Q_g$$

14. Провести нормализацию нового значения Q :

$$Q = Q/|Q|$$

15. Извлечь результат: перейти от кватерниона к актуальным углам крена, тангажа, и рыскания:

$$\psi = QuatToEulerYaw(Q)$$

$$\theta = QuatToEulerPitch(Q)$$

$$\phi = QuatToEulerRoll(Q)$$

16. Перейти к шагу 2.

Результат работы алгоритма – углы ψ , θ , ϕ , которые содержит актуальный поворот аппарата относительно поверхности земли. При этом ошибка интегрирования корректируется с помощью вектора ускорения свободного падения.

Главными достоинствами алгоритма являются низкие требования к вычислительным ресурсам и ясная геометрическая трактовка.

В данной работе используется реализация фильтра на языке Си, доступная по адресу [7], со следующими поправками:

- реализована проверка из пункта 6;
- функция приближённого вычисления обратного квадратного корня заменена своим точным вариантом.

Фильтр Маджвика

Другой вариант комплементарного фильтра, разработанный британским математиком Себастьяном Маджвиком вместе с коллегами [3]. Основное отличие от фильтра Махони – в способе внесения поправки от датчика линейного ускорения. Коррекция производится как решение задачи оптимизации. Вводится критерий:

$$e = \frac{1}{2} |\bar{b} - \bar{a}|^2 \rightarrow \min$$

Поправка рассчитывается методом градиентного спуска с учётом критерия. На каждой итерации алгоритма выполняется шаг градиентного спуска.

Вторая особенность алгоритма – поправка применяется не к угловой скорости, а скорости изменения кватерниона.

Достоинством алгоритма заявлено повышенное качество определения ориентации при сохранении низкой вычислительно сложности.

В данной работе используется реализация фильтра на языке Си, доступная по адресу [7], со следующими поправками:

- Реализована проверка, аналогично пункту 6 для фильтра Махони.
- Функция приближённого вычисления обратного квадратного корня заменена своим точным вариантом.

Фильтр Калмана

Фильтр Калмана – это метод из теории оценивания, который объединяет информацию из различных неопределённых источников для получения интересующих значений вместе с неопределённостью в них. Этот фильтр был успешно применён во многих сферах, таких как миссии на Марс и автоматизированные системы наведения ракет [4]. В отличие от рассмотренных выше фильтров, варианты фильтра Калмана обычно описываются в матричной форме. Работа фильтра в самом общем виде заключается в рекурсивном повторении этапов [5].

1. Этап прогноза – использует предыдущий результат фильтра и информацию о законах движения аппарата.

2. Этап коррекции – корректирует показания, используя данные с датчиков.

Фильтр Калмана считается «классическим» решением задач ориентации и навигации, однако для него характерен ряд недостатков [6].

- Повышенная вычислительная сложность: алгоритм требует работы с многомерными (и часто разреженными) матрицами и векторами. Особую сложность фильтр Калмана представляет для низкопроизводительных микроконтроллеров, применяемых в малых мультикоптерах.

- Трудность конфигурирования – часть параметров фильтра не имеет ясного физического смысла, и его настройка зачастую превращается в сложный и неинтуитивный процесс.

В данной работе используется вариант фильтра Калмана от фирмы NXP, его исходный код доступен по адресу [8].

Испытательное устройство

Для экспериментального сравнения алгоритмов было разработано испытательное устройство (рисунок 1). Оно включает в себя:

1. Комбинированный цифровой микроэлектромеханический датчик MPU-6500, содержащий:

- трёхосевой датчик угловой скорости;
- трёхосевой датчик линейного ускорения.

2. Трёхосевой манипулятор. Он содержит три цифровых сервопривода LOBOT LX-16A, выполняющих повороты по осям рыскания, тангажа и крена ($\psi \rightarrow \theta \rightarrow \phi$). Сочленения манипулятора изготовлены при помощи технологии 3D-печати.

3. Плата для разработчика STM32F3DISCOVERY с микроконтроллером STM32F303VC. Микроконтроллер выполняет четыре задачи.

- Сбор показаний датчика и оценка ориентации с помощью выбранного алгоритма. Выполняется с частотой 1000 Гц.
- Опрос текущего положения сервоприводов манипулятора. Выполняется с частотой 200 Гц.
- Отправка полученных данных на компьютер. Выполняется с частотой 200 Гц.
- Перемещение манипулятора по запросу компьютера.

План эксперимента

Перед началом эксперимента:

- датчики настраиваются на максимальный диапазон измерений ($\pm 2000^\circ/\text{с}$, $\pm 8\text{g}$);
- датчики калибруются: выполняется 1000 замеров и их усреднённое значение в дальнейшем применяется как поправка для коррекции погрешности смещения нуля.

Эксперимент состоит из трёх этапов:

1. Отсутствие перемещения.
2. Медленное перемещение манипулятора.
3. Быстрое перемещение манипулятора.

Последовательность движения манипулятора выбрана такой, чтобы включить все возможные комбинации крайних положений осей ($\pm 45^\circ$), движение от -45° до $+45^\circ$ по всем осям одновременно, а также нейтральную точку 0° (рисунок 2).

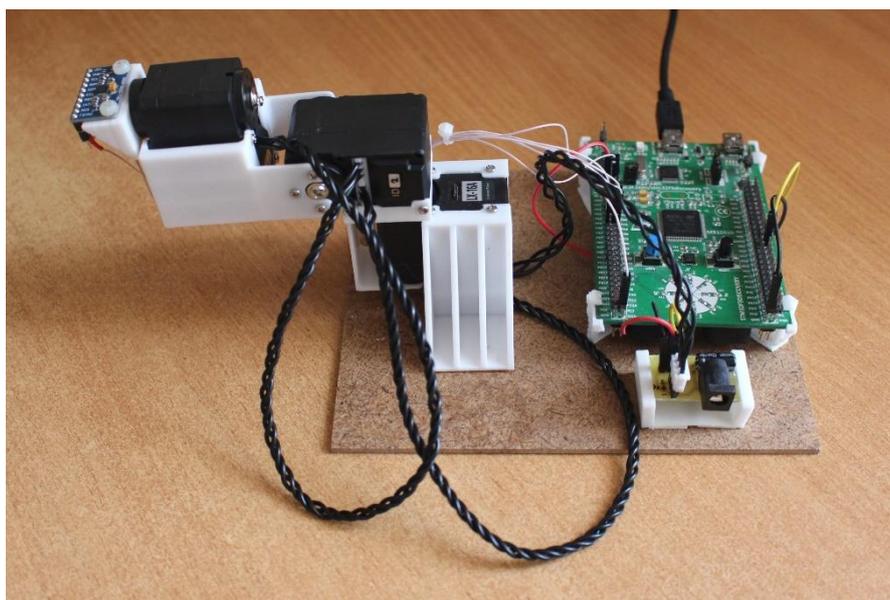


Рисунок 1 – Испытательное устройство. Слева направо: датчик MPU-6500, трёхосевой манипулятор, плата STM32F3DISCOVERY

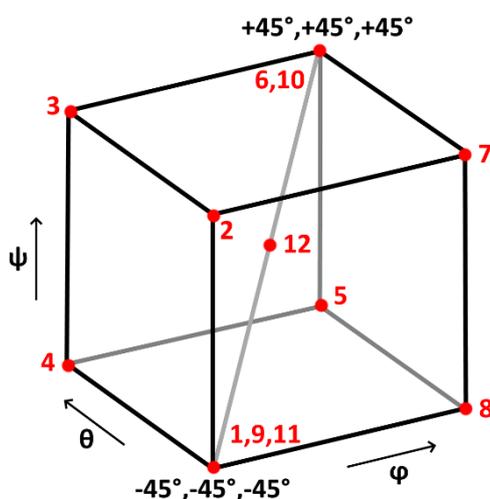


Рисунок 2 – куб в пространстве варьируемых параметров. Углы ψ , θ , ϕ изменяются в соответствии с пронумерованной последовательностью точек.

Для каждого этапа по каждой из осей строится график ошибки между реальным положением сервоприводов x_r и оценкой ориентации от рассматриваемого алгоритма x_p . Рассчитывается значение корня среднеквадратичного отклонения:

$$RMSE = \sqrt{\frac{\sum_{i=0}^N (x_r - x_p)^2}{N}}$$

Кроме того, измеряется вычислительная сложность алгоритма – время выполнения одного цикла оценки ориентации на микроконтроллере STM32F303VC.

Результаты экспериментов

Результаты экспериментов представлены в таблицах 1, 2, и на рисунке 3. Оба варианта комплементарного фильтра (Махони и Маджвика) показали схожие результаты, с незначительным преимуществом в сторону второго. Фильтр Калмана оказался лучшим при медленных или отсутствующих движениях, однако при быстрых перемещениях он оказался менее эффективен.

Кроме того, с фильтром Калмана обнаружилась дополнительная проблема: сильный дрейф по оси рыскания. Этот дрейф свойственен всем алгоритмам – вектор ускорения свободного падения не содержит информацию о вращении вокруг себя, следовательно, корректировка с помощью датчика линейного ускорения возможна лишь для оси тангажа и крена. Однако, в более простых комплементарных алгоритмах дрейф малозаметен. Это не означает, что невозможно сконструировать фильтр Калмана, который был бы этой проблеме не подвержен, однако подчёркивает всю сложность проектирования и настройки фильтров семейства.

В результате, предпочтительными алгоритмами определения ориентации по двум датчикам следует назвать варианты комплементарного фильтра – они оказались точнее и более простыми в вычислительном плане. При этом на стороне фильтра Маджвика – повышенная точность, а на стороне фильтра Махони – простота и наименьшая вычислительная сложность.

Таблица 1 – Корень среднеквадратичного отклонения ошибки оценки ориентации

Углы Эйлера (°)	Фильтр Махони	Фильтр Маджвика	Фильтр Калмана
Рыскание (без движения)	0.935	0.833	0.621
Тангаж (без движения)	0.454	0.290	0.303
Крен (без движения)	0.550	0.694	0.318
Рыскание (медленно)	3.287	3.794	7.712
Тангаж (медленно)	2.084	2.145	1.668
Крен (медленно)	3.785	4.13	3.178
Рыскание (быстро)	2.873	3.236	14.622
Тангаж (быстро)	2.726	2.613	3.189
Крен (быстро)	3.192	3.697	3.651
Среднее по всем строкам	2,210	2.163	3.918

Таблица 2 – Время выполнения одного цикла работы различных алгоритмов

Время (мкс)	Фильтр Махони	Фильтр Маджвика	Фильтр Калмана
Время выполнения цикла	297	300	445

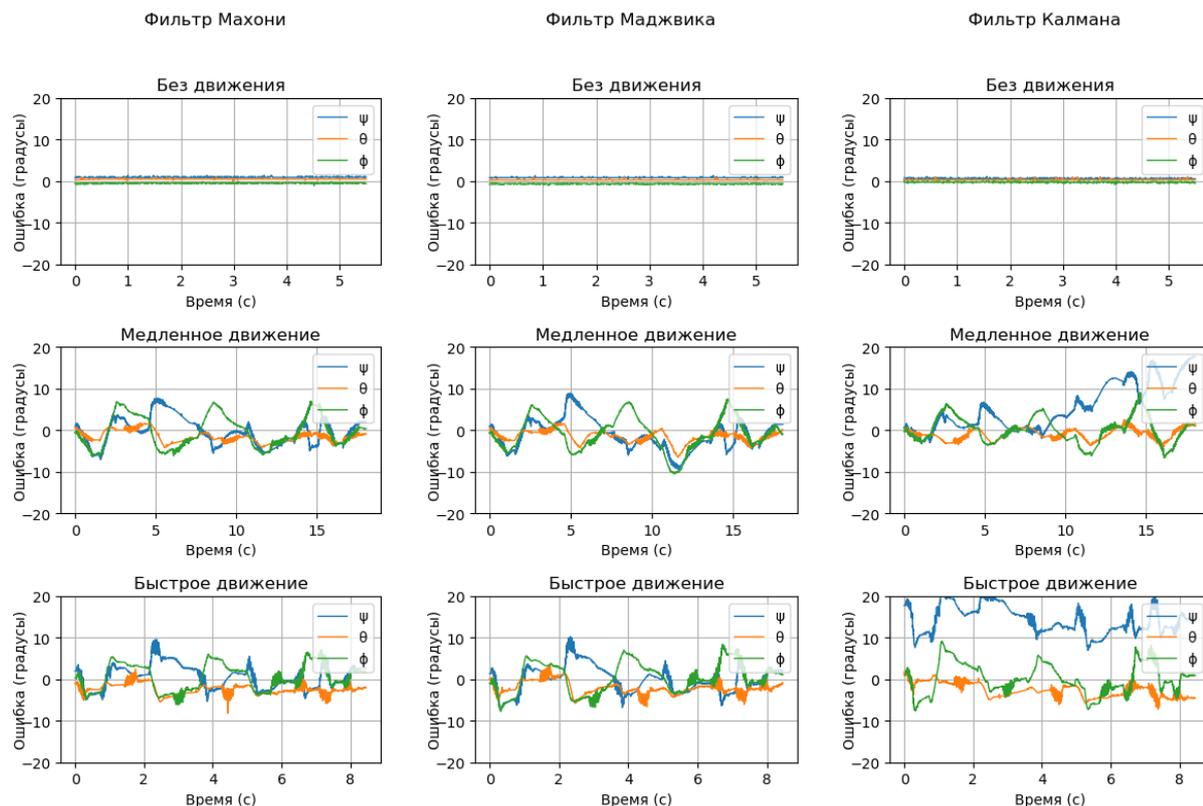


Рисунок 3 – Графики отклонения оценки положения от реального положения для различных алгоритмов.

В данной работе было проведено экспериментальное сравнение трех алгоритмов определения ориентации для мультикоптеров: фильтра Махони, фильтра Маджвика, а также варианта фильтра Калмана. Все фильтры рассматривались в варианте для двух датчиков: трехосевого датчика линейного ускорения и трехосевого датчика линейного ускорения. Результаты исследования выявили преимущество фильтров Махони и Маджвика.

Список литературы

1. Исследование колебаний квадрокоптера при внешних периодических воздействиях Попов Н.И., Емельянова О.В., Яцун С.Ф., Савин А.И. // Фундаментальные исследования. – 2014. – № 1 – С. 28-32
2. R. Mahony, Tarek Hamel, Jean-Michel Pflimlin. Nonlinear Complementary Filters on the Special Orthogonal Group. IEEE Transactions on Automatic Control, Institute of Electrical and Electronics Engineers, 2008, 53 (5), pp.1203-1217
3. Sebastian O.H. Madgwick, Andrew J.L. Harrison, Ravi Vaidyanathan. Estimation of IMU and MARG orientation using a gradient descent algorithm, 2011 IEEE International Conference on Rehabilitation Robotics, pp.179-185
4. Rudy Negenborn, Robot Localization and Kalman Filters. On finding your position in a noisy world / Thesis, Utrecht University, 2003

5. А. Н. Забегаев, В. Е. Павловский, Адаптация фильтра Калмана для использования с локальной и глобальной системой навигации, Препринты ИПМ им. М. В. Келдыша, 2010, 082, 24 с.
6. Vasiliy M. Tereshkov. A Simple Observer for Gyro and Accelerometer Biases in Land Navigation Systems, Journal of Navigation (2015), 68, pp. 635-645
7. Open source IMU and AHRS algorithms [электронный ресурс]. – Режим доступа: <https://x-io.co.uk/open-source-imu-and-ahrs-algorithms/> (01.04.2019).
8. Open-Source-Sensor-Fusion [электронный ресурс]. – Режим доступа: <https://github.com/memsindustrygroup/Open-Source-Sensor-Fusion> (01.04.2019).

References

1. Issledovanie kolebanij kvadrokoptera pri vneshnih periodicheskikh vozdeystvijah Popov N.I., Emel'janova O.V., Jacun S.F., Savin A.I. // Fundamental'nye issledovaniya. – 2014. – № 1 – p. 28-32
 2. R. Mahony, Tarek Hamel, Jean-Michel Pflimlin. Nonlinear Complementary Filters on the Special Orthogonal Group. IEEE Transactions on Automatic Control, Institute of Electrical and Electronics Engineers, 2008, 53 (5), pp.1203-1217
 3. Sebastian O.H. Madgwick, Andrew J.L. Harrison, Ravi Vaidyanathan. Estimation of IMU and MARG orientation using a gradient descent algorithm, 2011 IEEE International Conference on Rehabilitation Robotics, pp.179-185
 4. Rudy Negenborn, Robot Localization and Kalman Filters. On finding your position in a noisy world / Thesis, Utrech University, 2003
 5. A. N. Zabegaev, V. E. Pavlovskij, Adaptacija fil'tra Kalmana dlja ispol'zovanija s lokal'noj i global'noj sistemoj navigacii, Preprinty IPM im. M. V. Keldysha, 2010, 082, 24 p.
 6. Vasiliy M. Tereshkov. A Simple Observer for Gyro and Accelerometer Biases in Land Navigation Systems, Journal of Navigation (2015), 68, pp. 635-645
 7. <https://x-io.co.uk/open-source-imu-and-ahrs-algorithms/> [web-resource]
 8. <https://github.com/memsindustrygroup/Open-Source-Sensor-Fusion> [web-resource].
-