

# Описание алгоритма ESKF\_SIMPLE

Николай Майоров

# Содержание

1. Принцип работы алгоритма .....	3
2. Обозначения .....	3
3. Вектор состояния .....	4
4. Вектор ошибок .....	4
5. Динамическая модель для вектора состояния .....	5
6. Динамическая модель для вектора ошибок .....	6
6.1. Модель для смещений ГНСС-лидар .....	8
7. Обработка измерений .....	9
7.1. Скорость спидометра и кинематические ограничения .....	9
7.2. Координаты ГНСС .....	10
7.3. Скорость ГНСС .....	10
7.4. Координаты и ориентация от лидарного мэтчинга .....	10
7.4.1. Координаты .....	11
7.4.2. Ориентация .....	11
7.4.3. Обработка измерения .....	11
8. Альтернативный режим работы фильтра в «лидарной карте» .....	11
9. Прочие важные алгоритмические аспекты .....	12
9.1. Учет несовпадения времени состояния и измерения .....	12
9.2. Ковариация выдаваемого решения .....	13
9.3. Измерения ИИБ в модели предсказания .....	13
9.4. Учет стоянки .....	13
9.5. Учет ошибки калибровки лидара к ИИБ .....	14
10. Рекомендации по улучшению алгоритма .....	14

Это краткое описание алгоритма, реализованного в ESKF\_SIMPLE. Не подразумевается, что оно будет написано в академическом выверенном стиле (возможно это будет сделано со временем в будущем).

## 1. Принцип работы алгоритма

Это алгоритм оценивания типа Error State Kalman Filter или Extended Kalman Filter. Для предсказания вектора состояния используется кинематическая модель с привлечением инерциального измерительного модуля (ИИБ), прочие датчики обрабатываются, как «измерения в фильтре Калмана» по мере их поступления. Можно также сказать, что это корректируемая инерциальная навигационная система (ИНС). В разработке автор опирался на статью «Quaternion kinematics for the error-state Kalman filter».

## 2. Обозначения

Введем нужные обозначения. Используемые системы координат:

- $e$  — система ECEF
- $i$  — инерциальная система соосная с ECEF в начальный момент времени
- $n$  — система координат ENU в текущем положении ровера
- $b$  — система связанная с ИИБ
- $l$  — система координат «base link», которая определяется фиксированной ориентацией относительно осей ИИБ и приблизительно совпадает с осями автомобиля, начало находится в центре задней оси автомобиля. Будем называть её базовой системой координат
- $v$  — транспортная системы координат, связанная с колесной базой автомобиля. Начало совпадает с началом базовой системы координат, однако направление осей может немного отличаться. Для этой системы выполняются ограничения на направление скорости (т. н. неголономные ограничения)
- $a$  — система координат, связанная с антенной ГНСС, в реальности важно только положение центра этой системы

Обозначение векторов и матриц:

- $v^a$  — трехмерный геометрический вектор  $v$ , заданный компонентами в системе координат  $a$
- $C_b^a$  — матрица ориентации между системами  $a$  и  $b$  (в реализации с тем же успехом могут быть использованы кватернионы). Проекция вектора с помощью матрицы выражается тождеством  $v^a \equiv C_b^a v^b$
- $(v \times)$  — кососимметричная матрица для трехмерного вектора  $v$ , определяется согласно действию векторного произведения  $(v \times)u \equiv v \times u$
- $l_{ab}^c$  — радиус-вектор от центра системы  $a$  к центру системы  $b$ , выраженный в осях системы  $c$
- $\omega_{ab}^c$  — угловая скорость системы  $a$  относительно системы  $b$ , выраженная в осях  $c$

### 3. Вектор состояния

Фильтр реализует оценивание декартовых координат и скорости в системе ECEF, ориентации связанной системы координат относительно ECEF, а также некоторых вспомогательных состояний. Он использует следующий вектор состояния:

- $r^e$  — радиус-вектор относительно центра Земли, выраженный в ECEF
- $V^e$  — вектор скорости относительно центра Земли, выраженный в ECEF
- $C_b^e$  — матрица (в реализации кватернион) ориентации связанной системы координат, относительно ECEF
- $b_a$  — смещение нуля триады акселерометров
- $b_g$  — смещение нуля триады гироскопов
- $k$  — масштабный коэффициент спидометра
- $C_e^m$  (опционально) — матрица ориентации между «системой координат лидарной карты» и ECEF. Т. е. это ошибка оценки ориентации с помощью лидарной регистрации, вызванная ошибками привязки облака точек карты к ECEF в окрестности текущего местоположения
- $l_{em}^m$  (опционально) — смещение между «системой координат лидарной карты» и ECEF, выраженное в осях лидарной карты. Это ошибка оценки координат с помощью лидарной регистрации, вызванная ошибками привязки облака точек карты к ECEF в окрестности текущего местоположения
- $C_{b'}^b$  (опционально) — матрица ориентации между связанной системой координат и её уточненным положением за счет учета кинематических ограничений. Наиболее коротко её смысле можно передать тождеством

$$C_b^l C_{b'}^b \equiv C_b^v \quad (1)$$

Более удачной реализацией будет оценивать ориентацию  $C_v^l$ , т.е. оценивать несоосность в осях автомобиля (проще интерпретировать и задать начальную ковариацию)

### 4. Вектор ошибок

Вектор ошибок описывает отклонение оценок фильтра от истины. На этапе прогноза фильтр обновляет матрицу ковариации для вектора ошибок, а на этапе коррекции получают оценки этого вектора, которые сразу же корректируют номинальный вектор состояния.

В фильтре имеется два типа переменных: вектор из евклидова пространства (координаты, смещения нуля и т. д.) и матрица (кватернион) ориентации. Для второго типа вектор ошибок трехмерный и имеет смысл вектора малого поворота. Оценка и истинное значение связано следующим соотношением (это можно понять исходя из знака коррекции в коде и определения невязки  $z$  в обработке измерения):

- для вектора:  $\hat{X} = X + x$
- для ориентации:  $\hat{C} = C \exp(\phi \times) \approx C(I + (\phi \times))$

Здесь  $\hat{\cdot}$  обозначает оценку,  $x$  и  $\phi$  ошибка векторной величины и ориентации соответственно,  $\exp$  обозначает операцию матричного экспоненцирования или получения матрицы ориентации по вектору малого поворота.

Перечислим обозначения для элементов вектора ошибок:

- $\Delta r^e$  — ошибка оценки координат
- $\Delta V^e$  — ошибка оценки скорости
- $\alpha$  — ошибка оценки матрицы  $C_b^e$
- $\Delta b_a$  — ошибка оценки смещения нуля акселерометров
- $\Delta b_g$  — ошибка оценки смещения нуля гироскопов
- $\Delta k$  — ошибка оценки масштабного коэффициента спидометра
- $\gamma$  — ошибка матрицы  $C_e^m$
- $\Delta l_{em}^m$  — ошибка оценки смещения лидарной карты
- $\beta$  — ошибка оценки матрицы  $C_{b'}^b$

## 5. Динамическая модель для вектора состояния

Под этим понимаются система дифференциальных уравнений, согласно которым оцениваемые величины изменяются во времени. Для основных кинематических величин используются следующим уравнения:

$$\begin{aligned}\dot{r}^e &= V^e \\ \dot{V}^e &= C_b^e f^b + g^e \\ \dot{C}_b^e &= C_b^e (\omega_{ib}^b \times)\end{aligned}\tag{2}$$

Здесь

- $f^b$  — кажущееся ускорение доставляемое акселерометрами
- $\omega_{ib}^b$  — угловая скорость относительно инерциального пространства доставляемая гироскопами
- $g_e$  — удельная сила тяжести, которая зависит от координат и вычисляется согласно модели

Все остальные переменные в векторе состояния номинально полагаются неизменными, т. е. оцениваются, как случайные константы (возможно подверженные случайным блужданиями).

В этих уравнениях сделаны следующие приближения:

- Не учитывается вращение Земли в уравнении для  $V^e$  и  $\dot{C}_b^e$ . В обновлении ориентации при достижении точности оценки смещения нуля гироскопов порядка или ниже угловой скорости вращения Земли (15 град/час, что возможно для используемых МЭМС датчиков) произойдет нарушение адекватности модели, т. к. ошибки гироскопов проявляются в связанных осях, а угловая скорость вращения Земли в осях ECEF. В обновлении скорости игнорируется сила Кориолиса, которая может также оказаться сопоставимой с точностью оценки смещений нуля акселерометров.

Общая рекомендация — учитывать вращения Земли, благо это делается достаточно просто

- Вычисление силы тяжести в осях ECEF реализовано следующим образом: сначала вычисляется модуль этого вектора для текущей широты и долготы, согласно модели рекомендуемой в WGS84, а далее этот вектор направляется в **центр** Земли. Таким образом не учитываются два важных эффекта: несферичность Земли и центробежная сила за счет вращения Земли (её по соглашению суммируют с силой гравитационного притяжения и говорят о силе тяжести). Чтобы определить направление силы тяжести, нужно воспользоваться фактом, что она ортогональна поверхности модельного эллипсоида (который строится, как аппроксимация эквипотенциальной поверхности поля силы тяжести Земли) или вертикальна в ENU. Для более детальной дискуссии обращайтесь например к книге «GNSS, Inertial and Multisensor Integrated Navigation Systems». Угловую разницу этих направлений можно оценить с помощью разности между геоцентрической и геодезической широтами, которая проиллюстрирована ниже.

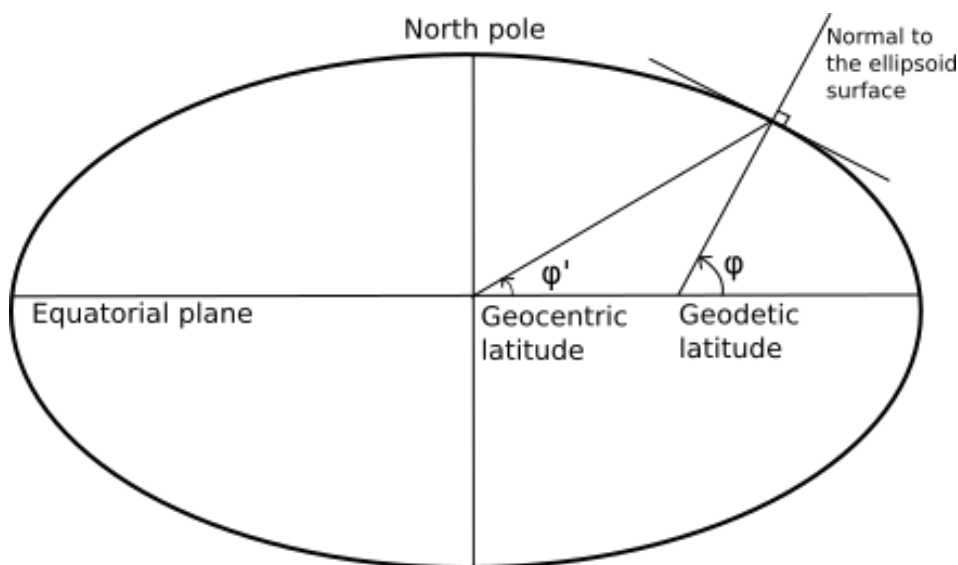


Рис. 1. Иллюстрация определений геоцентрической и геодезической широт

В радианах эта разность приближенно равна  $0.003 \sin(2\varphi)$ , где  $\varphi$  – широта места. Для широты Москвы это равно 0.16 градусам, что превосходит достижимую точность оценки углов горизонта с помощью МЭМС акселерометров. Предварительный анализ показывает, что это приведет к тому, что алгоритм будет оценивать углы крена и тангажа относительно «заваленного» горизонта, при этом указанная ошибка 0.16 градусов будет перераспределяться между ошибкой крена тангажа при поворотах ровера. Рекомендация — использовать правильный способ определения направления силы тяжести в ECEF.

## 6. Динамическая модель для вектора ошибок

Это линейная система дифференциальных уравнений для компонент вектора ошибок, иногда её также называют моделью или уравнениями ошибок. В фильтре она используется для прогноза ковариации, как таковой вектор ошибок на этапе прогноза отсутствует (он по смыслу неявно равен нулю, поэтому его обновление не требуется).

При анализе будем отталкиваться от используемой номинальной модели, игнорируя те ее неточности, которые были указаны выше.

Вычисляя разность между уравнениями для истинного состояния и уравнениями, используемые в фильтре, мы получим следующие уравнения ошибок:

$$\begin{aligned}\Delta \dot{r}^e &= \Delta V^e \\ \Delta \dot{V}^e &= -C_b^e(f^b \times) \alpha + C_b^e \Delta f^b + \Delta g^e \\ \dot{\alpha} &= \Delta \omega_{ib}^b - \omega_{ib}^b \times \alpha\end{aligned}\tag{3}$$

Из новых обозначений тут:

- $\Delta f^b$  — ошибки акселерометров
- $\Delta \omega_{ib}^b$  — ошибки гироскопов
- $\Delta g^e$  — ошибка вектора силы тяжести за счет ошибок координат. В текущей реализации модель силы тяжести неверная, а ошибки в модели не учитываются (это в принципе допустимо, но надо исправить саму модель)

Эта модель была проверена тремя способами:

- согласно коду
- моими собственными выкладками
- выкладками в статье Солы, на которую опирается код

Т. е. в рамках предположений она правильная и код реализует именно эту модель.

Ошибки гироскопов и акселерометров понимаются тут следующим образом — это разность между используемым измерением, равным физическому измерению датчика из которого вычитается оценка смещения нуля и истинным значением угловой скорости. Исходя из этого можно получить в рамках модели:

$$\begin{aligned}\Delta f^b &= -\Delta b_a + w_a \\ \Delta \omega_{ib}^b &= -\Delta b_g + w_g\end{aligned}\tag{4}$$

Где  $w_a$  и  $w_g$  стохастические ошибки акселерометров и гироскопов соответственно, которые в рамках модели считаются процессом типа белого шума.

Полная модель, которая реализуется в коде выглядит следующим образом:

$$\begin{aligned}\Delta \dot{r}^e &= \Delta V^e \\ \Delta \dot{V}^e &= -C_b^e(f^b \times) \alpha - C_b^e \Delta b_a + C_b^e w_a \\ \dot{\alpha} &= -\omega_{ib}^b \times \alpha - \Delta b_g + w_g \\ \Delta \dot{b}_g &= w_{gb} \\ \Delta \dot{b}_a &= w_{ab}\end{aligned}\tag{5}$$

Здесь  $w_{gb}$  и  $w_{ab}$  шумы, которые создают случайные блуждания смещения нуля гироскопов и акселерометров соответственно. Все остальные состояния, кроме «смещений ГНСС-лидар» (про них далее) моделируются, строго как константные.

Эта модель записывается в матричной форме вида  $\dot{x} = Fx + Gw$  и с помощью матриц  $F$  и  $G$  производится прогноз ковариации.

## 6.1. Модель для смещений ГНСС-лидар

Параметры, которые были обозначены как  $C_e^m$  и  $l_{em}^m$  введены для того, чтобы учесть локальные отклонения карты от идеального ECEF фрейма, к которому мы по концепции фильтра стремимся (и к которому можем привязываться с помощью ГНСС измерений). Моделирование изменчивости данных параметров сделано с помощью увеличения дисперсии соответствующих состояний в фильтре на величину пропорциональную **квадрату** пройденного расстояния на промежутке времени между которым делается обновление ковариации. Это осмысленно в том ключе, что структура лидарной карты меняется при движении, а не со временем. Однако предложенная модель с квадратичной пропорциональностью не соответствует никакому разумному случайному процессу, что можно пояснить следующим образом.

Предположим, что указанная константа пропорциональности равна  $(0.01 \text{ м} / 1 \text{ м})^2 = 0.01^2$  (т. е. на 1 метр пройденного пути, лидарная карта статически смещается на 1 см). Пусть ровер проехал 100 метров, тогда в зависимости от величин приращений мы получим разное увеличение ковариации состояния:

- Предположим приращения пути равны 1 м, тогда общее приращение ковариации будет  $0.01^2 \cdot 1^2 \cdot 100 = 0.01\text{м}^2$
- Предположим приращения пути равны 10 м, тогда общее приращение ковариации будет  $0.01^2 \cdot 10^2 \cdot 10 = 0.1\text{м}^2$

Т. е. для одного пройденного расстояния результат работы такого алгоритма будет зависеть от величины шагов. Скажем если шаг по времени определяется частотой ИИБ, то при смене частоты ИИБ фильтр существенно и неожиданно изменит свои свойства. Сам этот параметр пропорциональности не имеет того физического смысла, который в него закладывал автор, скорее ему трудно вообще придать какой-либо фундаментальный смысл.

Все встанет на свои места, если дисперсия состояния будет увеличиваться пропорционально пройденному расстоянию, что будет соответствовать случайному блужданию «по расстоянию». Т. е. дисперсия состояния должна увеличиваться на величину пропорциональную пройденному расстоянию, тогда увеличение дисперсии при прохождении какого-то расстояния не будет зависеть от того, какими шагами оно было пройдено. В виде непрерывной модели во времени это можно записать в виде:

$$\dot{l} = w\sqrt{V} \quad (6)$$

Т. е. шум должен масштабироваться на корень из модуля скорости. Единица измерения интенсивности таких блужданий будет м/ $\sqrt{\text{м}}$  для координат и  $^\circ/\sqrt{\text{м}}$  для ориентации (аналогично  $^\circ/\sqrt{\text{час}}$  для блужданий угла из-за шума гироскопов).



## 7. Обработка измерений

Фильтр поддерживает обработку следующих измерений:

- скорость спидометра и кинематические ограничения
- координаты ГНСС
- скорость ГНСС
- координаты и ориентация от лидарного мэтчинга

Далее будет разбор модели для каждого типа измерения.

### 7.1. Скорость спидометра и кинематические ограничения

В модели исходно предполагается, что измерения спидометра и кинематические ограничения относятся к транспортной системе координат. Для применения измерений метод `get_ref_velocity_and_H` вычисляет предсказание этой скорости а также связь ошибки предсказания с вектором ошибок (якобиан).

Вычисление оценки реализовано следующим образом:

- $\hat{V}_r^b = \hat{C}_e^b \hat{V}^e + \hat{\omega}_{ib}^b \times l_{bl}^b$  — оценка скорости в осях ИИБ с переносом в центр базовой (и транспортной) системы координат
- $\hat{V}_r^v = C_b^l \hat{C}_{b'}^b \hat{V}_r^b$  — проекция в транспортную систему координат
- $\hat{V}_{rs}^v = \hat{k} \hat{V}_r^v$  — только для обработки спидометра,

умножение на оценку масштабного коэффициента (более понятное решение — поделить измерение одометра на оценку масштабного коэффициента)

Связь ошибки такого предсказания с элементами вектора ошибок имеет вид:

$$\Delta V_{rs}^v = k C_b^v C_e^b \Delta V^e + k C_b^v (V^b \times) \alpha + k C_b^v (l_{bl}^b \times) \Delta b_g + V_r^v \Delta k - k C_b^v (V_r^b \times) \beta \quad (7)$$

Эта формула реализована в коде и она является правильной. Связь с ошибкой оценки смещения нуля ИИБ можно не рассматривать, но ошибки в её использовании нет.

Далее измерение обрабатывается стандартным образом:

- Для спидометра берется разность измерения и первой компоненты предсказания. Применяется небольшой трюк для определения знака скорости спидометра (если он не предоставляется самим спидометром) — знак берется по знаку предсказания фильтра. Этот трюк в целом разумный, однако требует расширения, чтобы определять случаи, когда достоверно определить знак невозможно и измерение должно отбрасываться, чтобы избежать возможной деградации оценки фильтра. Используется порог отбраковки «3-сигма»
- Для кинематических ограничений измерение берется равным 0. В ковариации измерения используется добавка пропорциональная значению скорости в центре базовой системы координат за счет вращения. Это какая-то эвристика, в целом разумная, однако какого-то сильного обоснования не имеет. Если все правильно реализовать, найти или оценить в транспортном средстве нужную точку (она может отличаться от центра задней оси), то модель нулевой боковой и вертикальной скорости при поворотах выполнется достаточно точно.

Отбраковка измерений выключена — это кажется неудачным решением, поскольку кратковременные нарушения (типа заносов транспортного средства) возможны.

## 7.2. Координаты ГНСС

Предсказание координат антенны делается стандартным образом

$$\hat{r}_a^e = \hat{r}^e + \hat{C}_b^e l_{ba}^b, \quad (8)$$

где  $l_{ba}^b$  — вектор от ИИБ до антенны в осях ИИБ. Ошибка предсказания

$$\Delta r_a^a = \Delta r^e - C_b^e (l_{ba}^b \times) \alpha \quad (9)$$

Возможная задержка ГНСС учитывается увеличением ковариации измерения пропорционально квадрату скорости движения. Это допустимое, но неоптимальное решение — лучше оценивать эту задержку, как элемент вектора состояния. А в адекватном ГНСС приемнике и при проставлении меток времени по общим часам, задержки позиционного решения относительно ИИБ не будет (это предпочтительный сценарий, который, насколько мне известно, выполняется в текущем сетапе).

Для отбрасывания измерения используется порог «3-сигма». Однако если измерение было отброшено, то ковариация измерения ГНСС увеличивается по геометрической прогрессии до тех пор, пока измерение не будет принято либо множитель ковариации не станет слишком большим. При этом порог квантиля хи-квадрат распределения увеличивается до 10%, т. е. измерение проверяется гораздо жестче. Это некоторая эвристика.

## 7.3. Скорость ГНСС

Предсказание скорости антенны делается в осях ENU:

$$\hat{V}_a^n = \hat{C}_e^n (\hat{V}^e + \hat{C}_b^e (\hat{\omega}_{ib}^b \times l_{ba}^b)) \quad (10)$$

Предсказание  $\hat{C}_e^n$  делается по ECEF координатам из фильтра, его ошибка игнорируется (это допустимо т. к. типичные ошибки координат «намного-намного» меньше, чем радиус Земли).

Связь ошибки предсказания с ошибками вектора состояния задается следующим образом

$$\Delta V_a^n = C_e^n \Delta V^e - C_e^n C_b^e ((\omega_{ib}^b \times l_{ba}^b) \times) \alpha + C_e^n C_b^e (l_{ba}^b \times) \Delta b_g \quad (11)$$

Измерение обрабатывается с порогом отбрасывания «3-сигма».

## 7.4. Координаты и ориентация от лидарного мэтчинга

Координаты и ориентация от лидарного мэтчинга по сути обрабатываются независимо, объединение происходит при формировании общего вектора невязок с 6 компонентами.

### 7.4.1. Координаты

Предсказание координат делается следующим образом:

$$\hat{r}_m^m = \hat{r}^e + \hat{l}_{em}^m + \hat{C}_e^m \hat{C}_b^e l_{bl}^b \quad (12)$$

Здесь нужно обратить внимание, что матрица  $\hat{C}_e^m$  не применяется к  $\hat{r}^e$ , что может казаться нужным с точки зрения геометрии и обозначений систем координат. Вопрос не столь простой, но кажется, что так делать не нужно и  $\hat{l}_{em}^m$  и  $\hat{C}_e^m$  имеют определенный смысл, как они используется сейчас. Если бы мы использовали  $\hat{C}_e^m$  для проекции  $\hat{r}^e$ , то оценки угловых ошибок умножались бы на большие компоненты вектора  $r^e$  (порядка радиуса Земли), что давало бы огромные смещения.

Ошибка предсказания

$$\Delta r_m^m = \Delta r_e + \Delta l_{lm}^m - C_e^m C_b^e (l_{bl}^l \times) \alpha - C_e^m (C_b^e l_{bl}^b \times) \gamma \quad (13)$$

### 7.4.2. Ориентация

Предсказание измерения делается следующим образом

$$\hat{C}_l^m = \hat{C}_e^m \hat{C}_b^e C_l^b \quad (14)$$

Невязка измерения вычисляется следующим образом

$$z_a = \log \left( \tilde{C}_l^m (\hat{C}_l^m)^T \right) \quad (15)$$

Где под  $\log$  подразумевается нахождение вектора поворота по матрице, а  $\sim$  — измерение. В коде реально используется обратный знак невязки, но и коррекция ошибки используется со знаком плюс, поэтому для записи якобианов я использую такое определение. Невязка связана с ошибками следующим образом

$$z_a = C_b^l \alpha + C_b^l C_e^b \gamma \quad (16)$$

### 7.4.3. Обработка измерения

Невязки и якобианы для координат и ориентации конкатенируются и обрабатываются стандартной процедурой коррекции в фильтре Калмана. Для порога обработки используется «3-сигма».

## 8. Альтернативный режим работы фильтра в «лидарной карте»

Режим активируется флагом `use_gnss_lidar_bias`. В этом режиме алгоритм оценивает координаты и скорость в «лидарной карте», т. е. в наших условных обозначениях  $r^m, V^m, C_b^m$ . Это приводит к тому, что в измерениях с лидаром не рассматриваются его смещения, а смещения возникают при обработке ГНСС данных. Выдача ковариации (которая делается для состояния `POS` без модификаций) здесь выглядит правильно.

Есть важный момент, что на этапе прогноза фильтра отклонение лидарной карты от ЕСЕФ не учитывается в вычислении вектора силы тяжести. Но, как мы выяснили, его направление там задается некорректно и в стандартном режиме.

Кроме выдачи ковариации, на уровне модели концептуально невидно, что в работе фильтра меняет переход к «лидарным координатам». Если все было бы корректно и консистентно реализовано (что не так), то оба подхода должны давать одинаковые оценки.

## 9. Прочие важные алгоритмические аспекты

Помимо используемых математических моделей также есть важные нюансы в реализации, которые существенно влияют на свойства, точность и консистентность выдаваемого решения.

### 9.1. Учет несовпадения времени состояния и измерения

При работе фильтра постоянно возникает следующая ситуация, что текущая оценка относится к моменту времени  $t$ , а пришедшее измерение к моменту времени  $t + \tau$ . Промежуток  $\tau > 0$  обычно меньше периода выдачи данных ИИБ, однако в случае потери данных ИИБ в системе может становится больше. Ясно, что рассогласование  $\tau$  нужно как-то учитывать и компенсировать.

В алгоритме применяется следующий способ: состояние фильтра буквально обновляется до времени  $t + \tau$ , с использованием последнего сохраненного сэмпла ИИБ (т. е. как будто бы в момент  $t + \tau$  пришел соответствующее измерение ИИБ). Данное решение привлекательно своей простотой, но в реальности является плохим и может либо ухудшать работу фильтра или вообще его ломать в зависимости от величины  $\tau$  и условий движения. Это можно объяснить разными способами, но ключевым является тот факт, что такая экстраполяция по ИИБ не может делаться «как будто это был реальный ИИБ» (по крайней мере в общем случае и полностью игнорируя все нюансы). Если кто-то работал с высокоточными ИИБ, то для них это было бы совершенно очевидно, т. к. такие датчики ловят миллионные доли угла поворота на малом промежутке времени и просто так подставлять туда значение с предыдущего сэмпла было бы грубой ошибкой. Для МЭМС датчиков это понимание может прийти с опытом работы с реальными данными и синтетическими тестами.

Решение, которое, строго говоря не оптимальное, но хорошо работает на практике заключается в том, что **измерение** переносится с момента времени  $t + \tau$  к моменту времени  $t$  (используя оценки фильтра). Но т. к. это эквивалентно тому, что мы перенесем состояния фильтра к моменту времени  $t + \tau$ , то проще так и делать (и это делается одним алгоритмическим блоком в коде). Большое и важное отличие, что эта экстраполяция не обновляет время фильтра, его состояние и ковариацию — это лишь способ убрать систематическое смещение между предсказанием и наблюдением за счет малого рассогласования времени.

Еще более оптимально — это учесть применяемую экстраполяцию в якобианах измерения, но я этим методом не пользовался (он требует дополнительной аналитической работы), но стоит попробовать.

## 9.2. Ковариация выдаваемого решения

В текущей версии выдаваемые величины вычисляются, как некоторая комбинация элементов вектора состояния, однако в выдаваемой ковариации это не учитывается. К примеру координаты в лидарной карте выдаются, как

$$\hat{r}_m^m = \hat{r}^e + \hat{l}_{me}^m, \quad (17)$$

однако ковариация выдается просто для  $\Delta r^e$ . Это очень грубый подход, который скрывает важные закономерности в работе фильтра ну и по сути выдает неправильную ковариацию (даже если бы все остальное работало идеально). К примеру для указанных «лидарных координат», если было получено точное и корректное лидарное измерение, то указанная комбинация будет иметь низкую ковариацию, однако отдельные блоки (ECEF координаты и «лидарное смещение») по отдельности могут иметь высокую ковариацию, если они не были оценены отдельно, когда в систему приходили точные координаты от ГНСС (условно «RTK\_FIXED»).

## 9.3. Измерения ИИБ в модели предсказания

В используемой модели ошибок на этапе предсказания нужно подставлять измерения и гироскопов и акселерометров. Это теоретически верно, но с практической точки зрения имеет минусы:

- Т. к. эти измерения отражают высокочастотные движения ровера, то встает вопрос о частоте реализации этапа прогноза. Нужно ли его делать на частоте ИИБ, можно ли делать его реже и осреднять измерения?
- Измерения ИИБ содержат шум и систематические ошибки, которые будут проникать в элементы якобианов
- Для контроля наблюдаемости на стоянке нужны дополнительные нетривиальные трюки (см. ниже)

Можно использовать другой выбор вектора ошибок и исключить измерения ИИБ из якобианов. Это является определенным «best practice», широко известным в узких кругах.

## 9.4. Учет стоянки

В алгоритмах такого типа стоянка является важным и вырожденным случаем, который требует отдельного рассмотрения. Рассмотрение заключается в том, что на стоянке во все якобианы нужно в качестве скорости подставлять строгий нуль, а в таком дизайне фильтра также нужно обратить внимание на измерения акселерометров и гироскопов (т. к. они также пролазят в якобианы). Если этого не делать, то шумные и меняющиеся около нуля числа в якобиане начнут создавать «ложную наблюдаемость» состояний (в основном связанных с курсовой ошибкой). При использовании лидарных мэтингов

эта проблема может носить менее выраженный характер и скорее относится к относительно неважным. Но для оптимального качества нужно обратить на это внимание.

## 9.5. Учет ошибки калибровки лидара к ИИБ

Эти параметры являются аналогией «смещений в лидарной карте», но выраженные в осях базовой системы. Они описывают ошибку определения осей и центра чувствительности лидара и имеют понятный и обоснованный физический смысл и модель (константу). Достаточно странно, что этот параметр не появился в модели оценивания. Он намного более на поверхности и проще для оценивания, чем упомянутые «смещения лидарной карты».

## 10. Рекомендации по улучшению алгоритма

- Использовать правильные геодезические модели (вращение Земли, направление силы тяжести)
- Исправить способ предсказания состояния ко времени измерения
- Устранить пропуски измерений ИИБ в рантайме и записанных rosbag
- Исправить модель изменения «лидарных байесов» со временем
- Подумать, как эти «лидарные байесы» лучше моделировать — разделить смещения на вертикальную/горизонтальные и курсовую/горизонтальные, исключить ненаблюдаемые или ненужные компоненты
- Добавить возможность оценки несоосности и смещения «ИИБ – лидар» в связанных осях (калибровка «ИИБ – лидар»)
- Как опция — смещение «ИИБ – антенна» в связанных осях
- Четко определить, какое состояние мы хотим выдавать, выдавать ковариацию согласованно
- Также важна полностью корректная инициализация начального состояния, включая учет корреляции между компонентами. Скажем если мы получаем начальную позу от лидаров, а используем как состояние координаты в ECEF и моделируем смещение между ECEF и «лидарной картой», то в начальной матрице ковариации должна возникнуть корреляция между ошибками координат и ошибками оценки смещения ГНСС-лидарная карта.
- Исключить измерения ИИБ из якобианов
- Улучшение работы на стоянке: зануление скорости в Якобианах, добавить коррекцию по нулевой скорости (Zero Velocity Update) и угловой скорости (Zero Integrated Rate Update)
- Продвинутая инициализация, объединение инициализации и рабочего режима в один непрерывный процесс (требует обсуждения и детального дизайна)

В итоге фильтр должен работать идеально в рамках заложенных моделей (что проверяется Монте-Карло моделированием) и включать нужные нам эффекты, далее настраивается под работу с реальными данными. Для оценки работы на реальных данных нужен набор тестовых логов («benchmark set») и количественные критерии оценки качества работы.

За основу стоит взять «ESKF\_PRECISE», в фильтре «ESKF\_SIMPLE» нет ничего особо ценного, чтобы продолжать его развивать. Существенная сложность заключается в упрощении организации кода, чтобы эти изменения воплотить каким-то разумным поступательным образом.