

# **Обработка измерений типа детекций центра полосы в навигационном фильтре**

Николай Майоров

Формулируется математическая модель для измерений типа детекции центра полосы. Рассматривается два варианта:

1. Обрабатывается смещение относительно центра полосы и угол движения в полосе
2. Обрабатывается набор точек, принадлежащих центру полосы

## 1. Состояние ровера и определение ошибок

Для ровера оцениваются его географические координаты  $\varphi, \lambda, h$  (что эквивалентно ECEF декартовым координатам  $r^e$ ) и ориентация относительно ENU —  $C_b^n$ . Ошибка координат определена в осях ENU —  $\Delta r^n$ , ошибка ориентации  $\phi^n$  определена, как вектор малого поворота, согласно следующему линеаризованному соотношению между оценкой и истинной ориентацией  $\hat{C}_b^n \approx (I - (\phi^n \times))C_b^n$ .

## 2. Представление линии центра полосы и её проекция в связанные оси

Дорога в HD-картах представляется в виде кусочков «lanelet», каждый из которых имеет центр полосы, представляемый в виде последовательности точек, заданных координатами в ECEF. Для данных координат ровера можно получить ближайший отрезок линии центра полосы. Таким образом, каждое измерение производится для отрезка, заданного двумя ECEF координатами начала и конца —  $r_1^e$  и  $r_2^e$ .

Проекция полосы заключается в вычислении относительных координат начала и вектора направления отрезка в связанных осях:

$$\begin{aligned} p^b &\equiv [x_0 \ y_0 \ z_0]^T = C_e^b(r_1^e - r^e) \\ \tau^b &\equiv [\tau_x \ \tau_y \ \tau_z]^T = C_e^b(r_2^e - r_1^e) \end{aligned} \quad (1)$$

Для простоты предполагается, что система координат CV совпадает со связанной системой координат, для которой получается навигационное решение.

## 3. Обработка измерений типа смещения и угла

В этой модели предполагается, что система дает нам агрегированные измерения, которые описывают прямую центра полосы в виде смещения и угла. Вместо этих двух параметров можно было бы использовать и другие — например коэффициент и свободный член уравнения прямой, принципиально это не будет влиять на работу алгоритма. Также нужно отметить, что по смыслу эти параметры будут коррелированы, поэтому теоретически их нужно обрабатывать, как одно измерение (на практике однако можно от этого отклониться, вопрос во многом эмпирический).

### 3.1. Математическое определение измерений

Измерение концептуально производится следующим образом:

1. Отрезок центра полосы проецируется в связанные оси. Спроецированный отрезок задает луч, направление которого выбирается вдоль продольной оси ровера

2. Смещение в полосе вычисляется, как знаковое расстояние от центра связанной системы координат до проекции луча в плоскость  $xu$
3. Угол движения в полосе вычисляется, как угол между осью  $x$  и проекцией луча в плоскость  $xu$

Указанная проекция отрезка центра полосы неявно производится как раз системой компьютерного зрения. Эта система включает обработку двумерных изображений с камеры, а также переход от двумерного образа полосы в связанные оси в трехмерном пространстве. Вся эта процедура является нетривиальной, её точность и стабильность сильно зависят от конкретной реализации. От чего в свою очередь зависит эффективность использования и полезность таких измерений в навигационном фильтре.

### 3.2. Формулы для измерений

Смещение вычисляется, как знаковое расстояние от начала связанной системы координат до проекции луча в плоскость  $xu$ :

$$s = \frac{-\tau_y x_0 + \tau_x y_0}{\sqrt{\tau_x^2 + \tau_y^2}} = (\eta^b)^T p^b \quad (2)$$

Где введен единичный вектор нормали в плоскости  $xu$ :

$$\eta^b = \frac{1}{\|\tau_h^b\|} \begin{bmatrix} -\tau_y \\ \tau_x \\ 0 \end{bmatrix}, \text{ где } \|\tau_h^b\| = \sqrt{\tau_x^2 + \tau_y^2} \quad (3)$$

Угол движения в полосе вычисляется, как угол между осью  $x$  и проекцией  $\tau^b$  в плоскость  $xu$ :

$$\beta = \text{atan2}(\tau_y, \tau_x) \quad (4)$$

Здесь предполагалось, что вектор  $\tau^b$  имеет положительную  $x$ -компоненту, если в вычислениях это оказывается не так, то его надо взять с обратным знаком.

### 3.3. Связь ошибки предсказания измерений фильтром с ошибками оценок

Навигационный фильтр также получает на вход координаты концов отрезка центра полосы и вычисляет свои предсказания смещения и угла движения, используя текущие оценки координат и ориентации. Пока будем предполагать, что фильтр получает на вход правильные координаты концов отрезка центра полосы (или точнее, что они задают правильную прямую в пространстве). В простейшей логике выбора ближайшей полосы успех будет зависеть от величины ошибок координат, геометрии дороги и устройства полос в текущей окрестности.

Ошибки предсказаний в линейном приближении будут зависеть от ошибок вычис-

$$\Delta s = \left(\frac{\partial s}{\partial p}\right)^T \Delta p^b + \left(\frac{\partial s}{\partial \tau}\right)^T \Delta \tau^b$$

ления  $p^b$  и  $\tau^b$  следующим образом:  $\Delta \beta = \left(\frac{\partial \beta}{\partial \tau}\right)^T \Delta \tau^b$   
производных имеют вид:

Где векторы частных

$$\begin{aligned} \frac{\partial s}{\partial p} &= \eta^b \\ \frac{\partial s}{\partial \tau} &= -\frac{x_0 \tau_x + y_0 \tau_y}{\|\tau_h^b\|^2} \eta^b \\ \frac{\partial \beta}{\partial \tau} &= \frac{1}{\|\tau_h^b\|} \eta^b \end{aligned} \quad (5)$$

Ошибки же проекций связаны с ошибками состояния ровера следующим образом:

$$\begin{aligned} \Delta p^b &= -C_n^b \Delta r^n - (p^b \times) C_n^b \phi^n \\ \Delta \tau^b &= -(\tau^b \times) C_n^b \phi^n \end{aligned} \quad (6)$$

Скомбинировав эти формулы получим:

$$\begin{aligned} \Delta s &= -\left(\frac{\partial s}{\partial p}\right)^T C_n^b \Delta r^n - \left( \left(\frac{\partial s}{\partial \tau}\right)^T (\tau^b \times) + \left(\frac{\partial s}{\partial p}\right)^T (p^b \times) \right) C_n^b \phi^n \\ \Delta \beta &= -\left(\frac{\partial \beta}{\partial \tau}\right)^T (\tau^b \times) C_n^b \phi^n \end{aligned} \quad (7)$$

### 3.3.1. Проверка для простой геометрии

Предположим, что ровер движется строго горизонтально с нулевым курсом и полоса идет строго параллельно его продольной оси. В этом случае формулы для всех элементов упрощаются

$$\begin{aligned} C_b^n &= I \\ \tau^b &= [\tau_x \ 0 \ 0]^T \\ \eta^b &= [0 \ 1 \ 0]^T \\ \frac{\partial s}{\partial p} &= [0 \ 1 \ 0]^T \\ \frac{\partial s}{\partial \tau} &= [0 \ -\frac{x_0}{\tau_x} \ 0]^T \\ \frac{\partial \beta}{\partial \tau} &= [0 \ \frac{1}{\tau_x} \ 0]^T \end{aligned} \quad (8)$$

Подставляя в формулы для ошибок предсказания, получим

$$\begin{aligned} \Delta s &= -\Delta r_N - z_0 \phi_E = -\Delta r_Y + z_0 \Delta \gamma \\ \Delta \beta &= \phi_U = -\Delta \psi \end{aligned} \quad (9)$$

Где  $\Delta r_Y, \Delta \gamma, \Delta \psi$  обозначают ошибку позиции в боковом направлении, ошибку угла крена и курса соответственно.

Эти формулы согласуются с интуитивным представлением, как должны работать такие измерения.

Ошибка смещения напрямую связана с ошибкой координат в боковом направлении. А влияние ошибки угла крена пропорционально вертикальному разнесению центра измерительной системы и плоскости дороги. Геометрически можно удостовериться, что такой эффект действительно должен проявляться — при крене плоскости ху линия полосы начнет проецироваться со смещением вдоль оси у.

Ошибка угла движения в полосе напрямую связана с ошибкой угла курса.

## 4. Обработка «сырых» точек

В этом варианте предполагается, что CV доставляет набор двумерных координат точек, каждая из которых с точностью до ошибок принадлежит прямой центра полосы. Здесь естественно предполагать, что каждое такое измерение независимо, поэтому достаточно рассмотреть алгоритм обработки одной такой точки.

При идеальной оценке состояния ровера, точности HD карты и измерений, точки будут строго ложиться на прямую центра полосы. В качестве корректирующих сигналов (невязок) можно предложить два варианта:

1. Расстояние от точки до прямой. В этом случае естественно предполагать ошибку измерений в обеих координатах  $x$  и  $y$ , такой метод аналогичен задаче полных наименьших квадратов
2. Отклонение предсказанной координаты  $y$  от измеренной. В этом случае ошибка предполагается только в измерениях  $y$ , что соответствует обычной задаче наименьших квадратов

Первый подход выглядит более геометрически правильным и совершенным, но он более сложен в плане реализации, формировании матриц для измерения (возможно содержит нюансы, которые сразу не предусмотреть). Второй подход более простой и понятный, сначала будет разработан именно он.

### 4.1. Предсказание координаты $y$

Уравнение прямой центра полосы в связанных координатах в канонической форме имеет вид

$$\tau_y(x - x_0) - \tau_x(y - y_0) = 0 \quad (10)$$

Из него можно выразить  $y$  напрямую

$$y = kx + b, \text{ где } k = \frac{\tau_y}{\tau_x}, b = y_0 - kx_0 \quad (11)$$

Такой вид уравнения хорошо определен если  $\tau_x$  существенно отличается от нуля, при движении в полосе это будет выполняться.

Предсказание координаты  $y$  производится для измеренной координаты  $\tilde{x}$  с помощью текущих оценок:

$$\hat{y} = \hat{k}\tilde{x} + \hat{b} \quad (12)$$

Параметры  $\hat{k}$  и  $\hat{b}$  вычисляются по формулам, используя текущие оценки состояния ровера.

## 4.2. Связь невязки с ошибками состояния

Нужно записать связь ошибки предсказания  $\hat{y}$  с ошибками состояния:

$$\begin{aligned} \Delta y &= \tilde{x} \left( \frac{\partial k}{\partial \tau} \right)^T \Delta \tau + \left( \frac{\partial y_0}{\partial p} \right)^T \Delta p - x_0 \left( \frac{\partial k}{\partial \tau} \right)^T \Delta \tau - k \left( \frac{\partial x_0}{\partial p} \right)^T \Delta p \\ \Delta y &= (\tilde{x} - x_0) \left( \frac{\partial k}{\partial \tau} \right)^T \Delta \tau + \left( \frac{\partial b}{\partial p} \right)^T \Delta p \end{aligned} \quad (13)$$

Где векторы частных производных имеют вид:

$$\begin{aligned} \frac{\partial k}{\partial \tau} &= \frac{1}{\tau_x} \begin{bmatrix} -\frac{\tau_y}{\tau_x} \\ 1 \\ 0 \end{bmatrix} \\ \frac{\partial b}{\partial p} &\equiv \frac{\partial y_0}{\partial p} - k \frac{\partial x_0}{\partial p} = \begin{bmatrix} -k \\ 1 \\ 0 \end{bmatrix} \end{aligned} \quad (14)$$

Подставив выражения для ошибок  $\tau$  и  $p$  через ошибки состояния получим:

$$\Delta y = - \left( \frac{\partial b}{\partial p} \right)^T C_n^b \Delta r_n - \left( (\tilde{x} - x_0) \left( \frac{\partial k}{\partial \tau} \right)^T (\tau^b \times) + \left( \frac{\partial b}{\partial p} \right)^T (p^b \times) \right) C_n^b \phi^n \quad (15)$$

Если заметить соотношение

$$\frac{\partial k}{\partial \tau} = \frac{1}{\tau_x} \frac{\partial b}{\partial p} \quad (16)$$

, то формулу можно записать в более кратком виде:

$$\Delta y = - \left( \frac{\partial b}{\partial p} \right)^T C_n^b \Delta r_n - \left( \frac{\partial b}{\partial p} \right)^T \left( \frac{\tilde{x} - x_0}{\tau_x} (\tau^b \times) + (p^b \times) \right) C_n^b \phi^n \quad (17)$$

### 4.2.1. Проверка для простой геометрии

Рассмотрим такую же ситуацию, как в предыдущем разделе. В этом случае будет

$$\begin{aligned} k &= 0 \\ \frac{\partial b}{\partial p} &= \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \end{aligned} \quad (18)$$

Подставив все выражения в формулу найдем:

$$\Delta y = -\Delta r_N - z_0 \phi_E + \tilde{x} \phi_U = -\Delta r_Y + z_0 \Delta \gamma - \tilde{x} \Delta \psi \quad (19)$$

Где  $\Delta r_Y$ ,  $\Delta \gamma$ ,  $\Delta \psi$  обозначают ошибку позиции в боковом направлении, ошибку угла крена и курса соответственно.

Здесь мы видим три слагаемых:

1. Первое напрямую соответствует ошибке координат в боковом направлении
2. Второе связано с ошибкой крена, эффект пропорционален высоте центра системы компьютерного зрения
3. Третье связано с ошибкой угла курса, при этом коэффициентом пропорциональности выступает продольная координата наблюдаемой точки. Т. е. чем дальше точка, тем больше курсовая ошибка ровера влияет на предсказание её  $y$  координаты. Нужно также отметить, что в реализации СКО шума измерения имеет смысл также назначать пропорционально данному расстоянию, т. к. системы компьютерного зрения хуже различают объекты на большом удалении

Если сравнить с Уравнение 9, то мы увидим, что оба подхода в плане измерительной информации принципиально не различаются.