

# Оглавление

<b>1</b>	<b>Одновременное оценивание движения ВС и систематических ошибок. Алгоритмы параллельной фильтрации процессов, связанных через измерения</b>	<b>2</b>
1.1	Описание задачи наблюдения за многими ВС . . . . .	3
1.2	Задача фильтрации . . . . .	6
1.3	Уравнения оптимальной фильтрации . . . . .	9
1.4	Упрощенные алгоритмы оценивания по Henk Blom . . . . .	12
1.4.1	Разделённые фильтры для фазового вектора и для систематической ошибки . . . . .	13
1.4.2	Фильтр Калмана для фазового вектора, макро-фильтр для систематической ошибки . . . . .	14
1.5	Алгоритм оценивания на основе CI . . . . .	15
<b>2</b>	<b>Алгоритм многогипотезного восстановления траектории</b>	<b>16</b>
2.1	Постановка задачи . . . . .	16
2.2	Схема первоначального построения и пересчёта пучка траекторий . . .	17
2.2.1	Построения в начальном окне . . . . .	17
2.2.2	Ветвление пучка при поступлении нового замера . . . . .	17
2.2.3	Группировка и прореживание треков . . . . .	17
2.2.4	Расчёт оценки текущего положения ВС . . . . .	18
2.3	Математическая модель траекторного движения ВС . . . . .	18
2.3.1	Аналитические расчёты при интегрировании уравнений движения	18
2.4	Структура данных программы, термины . . . . .	18
2.5	Вычисление веса треков и меры расхождения треков . . . . .	19
2.6	Алгоритм оптимизации . . . . .	19
2.7	Программа конвертации данных . . . . .	20
	<b>Литература</b>	<b>23</b>

# 1 Одновременное оценивание движения ВС и систематических ошибок. Алгоритмы параллельной фильтрации процессов, связанных через измерения

В настоящее время в системах УВД для определения параметров движения воздушных судов (координаты, скорости, ускорения и т.д.) используются алгоритмы линейного рекуррентного оценивания, близкие по используемой математической технике к фильтру Калмана. В качестве основного метода применяется алгоритм IMM. Главная особенность состоит в том, что задача оценки параметров движения для всех ВС, находящихся в зоне наблюдения, решается независимо для каждого ВС. Это полностью соответствует представлению о том, что движение каждого ВС никак не зависит от движения других ВС. Также это удобно с точки зрения архитектуры программы, реализующей систему мультитраекторной обработки — данные, описывающие каждое ВС, можно легко выделить в отдельный объект, который можно создавать, удалять и использовать, например, для сравнения со вновь поступающими не привязанными к конкретному ВС измерениями. С точки зрения математических алгоритмов, такое разделение также удобно, поскольку позволяет оставаться в рамках расчётов в пространстве достаточно низкой размерности (4–6 для фильтра Калмана, 15–30 для IMM).

Наблюдение за движением ВС производится с помощью радиотехнических средств: как правило это система из нескольких радиолокаторов и система АЗН-В. Реальные измерительные средства, помимо случайных ошибок измерений, имеют систематические ошибки. Случайные ошибки измерения изначально предусмотрены архитектурой алгоритмов рекуррентного оценивания, как фильтра Калмана, так и IMM. Систематические ошибки в случае не сложных вариантов их пространственной зависимости также легко могут быть включены в алгоритмы оценивания, но при их включении обнаруживается одно весьма существенное обстоятельство: систематические ошибки одного и того же измерительного средства присутствуют в уравнении наблюдения для разных воздушных судов. Так, в простом случае связи между неизвестными оцениваемыми состояниями и измерением РЛС возникает следующее линейное уравнение наблюдения:

$$z_{al}(t) = C^x(t)\chi_a(t) + C^s(t)\varsigma_l(t) + D(t)w_l(t). \quad (1.1)$$

Здесь  $t$  — момент времени;  $a$  — индекс, обозначающий номер воздушного судна (aircraft);  $l$  — индекс радиолокатора (locator);  $z_{al}$  — вектор измерения;  $\chi_a$  — вектор параметров движения ВС;  $\varsigma_l$  — вектор параметров, характеризующий состояние РЛС;  $w_l(t)$  — текущая реализация случайной ошибки РЛС;  $C^x(t)$ ,  $C^s(t)$ ,  $D(t)$  — матрицы, характеризующие влияние каждого параметра на измерение.

Из вида этого уравнения ясно, что систематическая ошибка локатора  $l$  может быть оценена только совместно с параметрами движения ВС  $a$ . Но этот радиолокатор наблюдает не только это движение, также верно и обратное — ВС  $a$  наблюдается не только радиолокатором  $l$ . Фазовые переменные для разных движений оказываются «сцепленными» между собой через параметры систематических ошибок. Таким образом, система всех движений и всех систематических ошибок нуждается в совместном

оценивании.

Как будет показано далее, даже в простом случае неуправляемых движений, стандартные процедуры оптимального совместного оценивания — фильтр Калмана, оценка Гаусса–Маркова — приводят к соотношениям, в которых переменные, относящиеся к разным движениям и систематическим ошибкам, существенно связаны друг с другом. Это приводит к следующим неприятным последствиям:

- нет возможности задать в программе отдельные объекты для движений разных ВС;
- затруднено создание и удаление движений;
- в вычислениях необходимо поддерживать большую матрицу ковариации ошибок оценивания, (в которую входят все кросс-ковариации для ошибок оценивания между различными ВС, между каждым ВС и каждым РЛС и т.д.) это выливается в большие затраты по времени вычисления и по памяти.

От требования, чтобы параметры оценивались оптимально, можно отказаться. При этом появляется возможность устранить нежелательные эффекты, указанные выше. Но в таком случае необходимо тщательно проектировать алгоритм оценивания, для того чтобы получаемые оценки были близки к неизвестным истинным параметрам.

Целью исследования, излагаемого ниже, является создание алгоритма лёгкого для параллельной реализации по отдельным воздушным судам и при этом обладающего низким уровнем погрешности оценивания. Исследование логически продолжает исследование, изложенное в отчёте [1].

## 1.1 Описание задачи наблюдения за многими ВС

Считаем, что каждое воздушное судно подчиняется независимому, но оди по структуре уравнению движения. Так движение ВС номер  $i$  имеет описание

$$d\chi_i(t) = f(t, \chi_i(t), u_i(t))dt + dv_i(t),$$

где  $\chi_i$  — вектор параметров движения ВС;  $f$  — функция, задающая скорости движения;  $u_i(t)$  — функция управления, специфичная для ВС  $i$ ;  $dv_i$  — приращение случайного возмущения для непрерывного варианта динамики; само дифференциальное уравнение сформулировано, например, в смысле Ито. В силу того, что наблюдение за ВС ведётся «в большом масштабе», вектор  $\chi_i$  может содержать не очень большое число параметров, а функция  $f$  может быть выбрана достаточно простой. Измерения при помощи РЛС производятся в дискретные моменты времени, поэтому далее удобно иметь дело с дискретизированным вариантом системы. При этом разумно ограничиться динамикой, близкой к линейной

$$\chi_i(t_k) = A_i(t_k, \chi_i(t_{k-1}), u_i(t_k))\chi_i(t_{k-1}) + B_i(t_k)v_i(t_k). \quad (1.2)$$

Здесь  $v_i$  — случайное возмущение;  $B_i$  — матричная функция, формирующая влияние случайного возмущения на движение;  $A_i$  — матрица, формирующая вид движения системы, зависящая от текущего значения управления  $u(t_k)$ . Моменты времени  $t_k$  принадлежат некоторому дискретному множеству  $\mathcal{T}$  и, на самом деле, определяются по ходу развития движения, т.е. не являются заданными заранее.

В программе мультирадарной обработки для метода ИММ уравнения движения используются именно в виде (1.2). Далее, будем рассматривать более простую линейную динамику без управления

$$\chi_i(t_k) = A_i(\mathcal{T}_k)\chi_i(t_{k-1}) + B_i(t_k)v_i(t_k). \quad (1.3)$$

Здесь  $\mathcal{T}_k = \{t_l \in \mathcal{T}: t_l \leq t_k\}$  — множество моментов времени до текущего включительно.

В качестве основного варианта при моделировании выбираем прямолинейное равномерное движение на плоскости

$$\chi_i(t_k) = \begin{bmatrix} x_i(t_k) \\ v_i(t_k) \end{bmatrix}, \quad x_i(t_k), v_i(t_k) \in \mathbb{R}^2, \quad A_i(\mathcal{T}_k) = \begin{bmatrix} I_{2 \times 2} & (t_k - t_{k-1})I_{2 \times 2} \\ 0_{2 \times 2} & I_{2 \times 2} \end{bmatrix}, \quad (1.4)$$

где  $x_i, v_i$  обозначают векторы координат и скорости на плоскости  $\mathbb{R}^2$ . Непосредственно в моделировании используется  $B_i \equiv 0, v_i \equiv 0$ .

Формирование наблюдений  $z_{ij}$  будем описывать следующим уравнением наблюдения, несколько более сложным, чем уравнение (1.1):

$$z_{ij}(t) = C_i^x(t_k)\chi_i(t_k) + C_j^s(t_k, \chi_i(t_k))\varsigma_j(t_k) + D_j(t_k, \chi_i(t_k))w_j(t_k). \quad (1.5)$$

Матрицы  $C_j^s, D_j$  для всех имеющих смысл случаев зависят от положения ВС, поэтому явно указывается зависимость от  $\chi_i$ . В качестве параметров  $\varsigma_j$  могут выступать постоянная систематическая ошибка по дальности и азимуту, коэффициент линейной зависимости для систематической ошибки по дальности и т.д. Матрица  $C_j^s$  описывает влияние этих неизвестных параметров на измерения.

Для параметров  $\varsigma_j$ , характеризующих систематические ошибки РЛС, также введём динамику

$$\varsigma_j(t_k) = A_j^s(t_k)\varsigma_j(t_{k-1}) + B_j^s(t_k)v_j^s(t_k). \quad (1.6)$$

Матрица  $B_i^s$  характеризует дрейф систематических ошибок со временем. Для моделирования будем принимать:

$$\varsigma_j = \begin{bmatrix} \Delta_j^r \\ \Delta_j^\alpha \end{bmatrix}, \quad A_j^s(t_k) \equiv I_{2 \times 2}, \quad B_i^s(t_k) \equiv 0_{2 \times 2}. \quad (1.7)$$

Здесь  $\Delta_j^r, \Delta_j^\alpha \in \mathbb{R}$  — значения постоянных систематических ошибок по дальности и азимуту, соответственно. Подробно понятия систематических ошибок по дальности и азимуту введены в отчёте ???.

Рассмотрим общий фазовый вектор

$$\xi(t) = \begin{bmatrix} \chi_1(t) \\ \chi_2(t) \\ \vdots \\ \chi_n(t) \\ \varsigma_1(t) \\ \varsigma_2(t) \\ \vdots \\ \varsigma_m(t) \end{bmatrix}. \quad (1.8)$$

Здесь  $n$  и  $m$  — количества наблюдаемых ВС и наблюдающих радиолокаторов. Уравнения (1.3), (1.6) можно переписать как

$$\begin{aligned} \xi(t_k) &= A(\mathcal{T}_k)\xi(t_{k-1}) + B(t_k)v(t_k) = \\ &= \begin{bmatrix} A_1(\mathcal{T}_k) & & & 0 \\ & \ddots & & \\ & & A_n(\mathcal{T}_k) & \\ & & & A_1^\varsigma(t_k) \\ & & & & \ddots \\ 0 & & & & & A_m^\varsigma(t_k) \end{bmatrix} \begin{bmatrix} \chi_1(t_{k-1}) \\ \vdots \\ \chi_n(t_{k-1}) \\ \varsigma_1(t_{k-1}) \\ \vdots \\ \varsigma_m(t_{k-1}) \end{bmatrix} + \\ &+ \begin{bmatrix} B_1(t_k) & & & 0 \\ & \ddots & & \\ & & B_n(t_k) & \\ & & & B_1^\varsigma(t_k) \\ & & & & \ddots \\ 0 & & & & & B_m^\varsigma(t_k) \end{bmatrix} \begin{bmatrix} v_1(t_k) \\ \vdots \\ v_n(t_k) \\ v_1^\varsigma(t_k) \\ \vdots \\ v_m^\varsigma(t_k) \end{bmatrix}, \quad (1.9) \end{aligned}$$

где матрицы  $A$  и  $B$  представляют собой блочно-диагональные матрицы, объединяющие все  $A_i$ ,  $A_i^\varsigma$  и  $B_i$ ,  $B_i^\varsigma$ . Также будем использовать символы  $A^\chi$ ,  $A^\varsigma$ ,  $B^\chi$ ,  $B^\varsigma$  для обозначения верхнего и нижнего блоков матриц  $A$  и  $B$ , соответствующих динамике переменных  $\chi$  и  $\varsigma$ .

Каждый момент времени  $t_k \in \mathcal{T}$  свяжем с некоторым измерением  $z_{ij}(t_k)$  положения ВС с номером  $i$  при помощи радиолокатора  $j$ . Одновременное наблюдение одного ВС несколькими радиолокаторами (как и одновременное наблюдение одним радиолокатором нескольких самолётов) будем считать пренебрежимо редким событием и не будем вводить его в модель наблюдения. Запишем уравнение наблюдения в том виде, как оно должно применяться ко всему большому фазовому вектору.

$$\begin{aligned} z(t_k) &= z_{ij}(t_k) = C(t_k, \xi(t_k))\xi(t_k) + D(t_k, \xi(t_k))w(t_k), \quad (1.10) \\ C(t_k, \xi(t_k)) &= \begin{bmatrix} C^\chi(t_k) & C^\varsigma(t_k, \chi(t_k)) \end{bmatrix}, \\ C^\chi(t_k) &= \begin{pmatrix} 0 & \cdots & 0 & C_i^\chi(t_k) & 0 & \cdots & 0 \end{pmatrix}, \\ C^\varsigma(t_k, \chi(t_k)) &= \begin{pmatrix} 0 & \cdots & 0 & C_j^\varsigma(t_k, \chi_i(t_k)) & 0 & \cdots & 0 \end{pmatrix}, \\ D(t_k, \xi(t_k)) &= D_j(t_k, \chi_i(t_k)), \quad w(t_k) = w_j(t_k). \end{aligned}$$

Как указывалось выше, для моделирования будем применять предположение постоянных систематических ошибок по дальности и азимуту. При этом будем использовать линеаризованную модель воздействия таких ошибок на измерения. Соответствующие матрицы  $C^\chi(t_k)$ ,  $C^\varsigma(t_k, \chi_i(t_k))$ ,  $D(t_k, \chi_i(t_k))$  имеют вид:

$$\begin{aligned} C_i^\chi(t_k) &\equiv \begin{bmatrix} I_{2 \times 2} & 0_{2 \times 2} \end{bmatrix}, \quad C_j^\varsigma(t_k, \chi_i(t_k)) = \begin{bmatrix} \frac{1}{\|x_i(t_k) - x_j^R\|} (x_i(t_k) - x_j^R) & \Omega_{2 \times 2}^{\pi/2} (x_i(t_k) - x_j^R) \end{bmatrix}, \\ D(t_k, \chi_i(t_k)) &= C_j^\varsigma(t_k, \chi_i(t_k)), \quad \Omega_{2 \times 2}^{\pi/2} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}. \quad (1.11) \end{aligned}$$

Здесь  $x_j^R$  — координаты точки стояния радиолокатора  $j$ ;  $\Omega_{2 \times 2}^{\pi/2}$  — матрица поворота на угол  $\frac{\pi}{2}$  против часовой стрелки на плоскости  $\mathbb{R}^2$  с учётом северо-восточной системы

координат. Случайные ошибки

$$w_j^s(t_k) = \begin{bmatrix} w_j^r(t_k) \\ w_j^\alpha(t_k) \end{bmatrix}$$

разделяются на случайные ошибки, действующие по дальности и азимуту.

Для всех случайных ошибок считаем справедливыми свойства:

$$\begin{aligned} \mathbb{E}\{v_i(t_k)\} &= \mathbb{E}\{v_i^s(t_k)\} = \mathbb{E}\{w_j(t_k)\} = 0, \\ \mathbf{Cov}\{v_{i_1}(t_k), v_{i_2}(t_l)\} &= \delta_{kl}\delta_{i_1 i_2} V_{i_1}^x, \quad \mathbf{Cov}\{w_{j_1}(t_k), w_{j_2}(t_l)\} = \delta_{kl}\delta_{j_1 j_2} W_{j_1}, \\ \mathbf{Cov}\{v_{j_1}^s(t_k), v_{j_2}^s(t_l)\} &= \delta_{kl}\delta_{j_1 j_2} V_{j_1}^s, \quad \mathbf{Cov}\{v_i(t_k), w_j(t_l)\} = \mathbf{Cov}\{v_{j_1}^s(t_k), w_{j_2}(t_l)\} = \\ &= \mathbf{Cov}\{v_j^s(t_k), v_i(t_l)\} = 0, \quad \forall i, i_1, i_2 \in 1, \dots, n, \quad \forall j, j_1, j_2 \in 1, \dots, m, \quad \forall t_k, t_l \in \mathcal{T}, \end{aligned} \quad (1.12)$$

где  $\delta_{pq}$  — символ Кронекера;  $V_i^x$  — постоянная матрица дисперсии случайных возмущений для уравнений движения;  $V_j^s$  — постоянная матрица дисперсии случайных возмущений для уравнения эволюции параметров радиолокатора  $j$ ;  $W_j$  — постоянная матрица дисперсии случайных ошибок наблюдения для радиолокатора  $j$ . Матрицы ковариаций для больших столбцов  $v$  и  $w$  будем обозначать

$$V^x = \begin{bmatrix} V_1^x & & 0 \\ & \ddots & \\ 0 & & V_n^x \end{bmatrix}, \quad V^s = \begin{bmatrix} V_1^s & & 0 \\ & \ddots & \\ 0 & & V_m^s \end{bmatrix},$$

$$V = \begin{bmatrix} V^x & 0 \\ 0 & V^s \end{bmatrix}, \quad W = \begin{bmatrix} W_1 & & 0 \\ & \ddots & \\ 0 & & W_m \end{bmatrix}.$$

Для моделирования будем применять  $W_j$  вида:

$$W_j = \begin{bmatrix} \sigma_{rj}^2 & 0 \\ 0 & \sigma_{\alpha j}^2 \end{bmatrix}, \quad (1.13)$$

где  $\sigma_{rj}$ ,  $\sigma_{\alpha j}$  — заданные среднеквадратичные отклонения для случайных ошибок наблюдения по дальности и азимуту, относящихся к радиолокатору  $j$ . Матрицы  $V_i^x$  будем брать одинаковыми диагональными, также будем поступать и с матрицами  $V_j^s$  (в данном случае не будем раскрывать подробный вид и смысл диагональных элементов, поскольку при моделировании эта матрица принималась равной нулю).

$$V_i^x = \begin{bmatrix} \sigma_{x1}^2 & & 0 \\ & \sigma_{x2}^2 & \\ 0 & & \sigma_{v1}^2 \\ & & & \sigma_{v2}^2 \end{bmatrix}, \quad V_j^s = \begin{bmatrix} \sigma_{\zeta 1}^2 & & 0 \\ & \ddots & \\ 0 & & \sigma_{\zeta q}^2 \end{bmatrix}. \quad (1.14)$$

## 1.2 Задача фильтрации

Целью фильтрации является получение оценки  $\hat{\xi}(t_k)$  фазового вектора  $\xi$  на момент  $t_k$  поступления последнего измерения. Предполагается, что оценка вычисляется как некоторая функция  $\Xi$  от информации обо всех измерениях до этого момента времени:

$$\hat{\xi}(t_k) = \Xi(\{z(t)\}_{t \in \mathcal{T}_k}),$$

а также от априорной информации. Также предполагается, что задан некоторый критерий, по которому будет определяться качество оценивания. Популярным выбором является:

$$J(t_k) = \mathbb{E} \left\{ \|h^\top (\hat{\xi}(t_k) - \xi(t_k))\|^2 \right\}, \quad (1.15)$$

где  $h^\top$  — некоторая заданная линейная функция, выделяющая, например, некоторую часть координат из всего вектора,  $\xi(t_k)$  — истинное значение фазового вектора  $\xi$  в момент времени  $t_k$ . Поскольку речь идёт об оценивании в присутствии случайных ошибок наблюдения, оценка  $\hat{\xi}(t_k)$  является случайной величиной, и в критерии присутствует символ математического ожидания  $\mathbb{E}\{\cdot\}$ .

Наиболее простыми и разумными с точки зрения оптимальности являются линейные рекуррентные оценки с линейным прогнозированием:

$$\bar{\xi}(t_k) = A(\mathcal{T}_k) \hat{\xi}(t_{k-1}), \quad (1.16)$$

$$\hat{\xi}(t_k) = L(t_k, R(t_k)) \bar{\xi}(t_k) + K(t_k, R(t_k)) z(t_k), \quad (1.17)$$

$$R(t_k) = \mathcal{F}(t_k, R(t_{k-1})). \quad (1.18)$$

Здесь  $L(t_k, R(t_k))$  и  $K(t_k, R(t_k))$  — матричные коэффициенты, выбираемые для каждого момента самостоятельно, и зависящие от параметров линейных уравнений (1.9), (1.10), а также от вектора дополнительных параметров  $R(t_k)$ , пересчитываемого отдельно по некоторому, уже в общем случае нелинейному, правилу (1.18). Уравнение прогноза (1.16) обеспечивает оптимальную по имеющейся информации  $\hat{\xi}(t_{k-1})$  оценку вектора  $\xi(t_k)$  среди всех возможных оценок вообще. Т.е. при оптимальном выборе  $\hat{\xi}(t_{k-1})$  оценка  $\bar{\xi}(t_k)$  является оптимальной среди всех оценок вектора  $\xi(t_k)$  по измерениям, предшествующим моменту  $t_k$ . Уравнение (1.17) называют уравнением коррекции. Его целью является получение новой оценки, учитывающий последнее измерение.

Далее в тексте, если рассматриваемые величины  $\hat{\xi}(t_k)$ ,  $\bar{\xi}(t_k)$ ,  $L(t_k, R(t_k))$ , и т. д. относятся к одному и тому же моменту времени  $t_k$ , скобки с аргументами в некоторых случаях будут опускаться, если это не будет создавать двусмысленности.

Популярным дополнительным условием является условие несмещённости оценки

$$\mathbb{E} \left\{ \hat{\xi}(t_k) \right\} = \mathbb{E} \left\{ \xi(t_k) \right\}, \quad (1.19)$$

которое в случае детерминированного фазового вектора  $\xi$ , например, в случае равенства нулю матрицы  $B(t_k)$  в уравнении (1.9), принимает вид

$$\mathbb{E} \left\{ \hat{\xi}(t_k) \right\} = \xi(t_k). \quad (1.20)$$

Если оценка  $\hat{\xi}(t_{k-1})$  удовлетворяет условию (1.19), легко видеть, что и оценка  $\bar{\xi}(t_k)$  ему удовлетворяет в силу уравнения (1.9). Для уравнения коррекции (1.17) условие несмещённости (1.19) приводит к следующему условию

$$\begin{aligned} \mathbb{E} \left\{ \hat{\xi}(t_k) \right\} &= L \mathbb{E} \left\{ \bar{\xi}(t_k) \right\} + K \mathbb{E} \left\{ z(t_k) \right\} = \\ &= L \mathbb{E} \left\{ \xi(t_k) \right\} + K \mathbb{E} \left\{ C(t_k, \xi(t_k)) \xi(t_k) \right\} + K \mathbb{E} \left\{ D(t_k, \xi(t_k)) w(t_k) \right\} = \\ &= L \mathbb{E} \left\{ \xi(t_k) \right\} + K \mathbb{E} \left\{ C(t_k, \xi(t_k)) \xi(t_k) \right\} + K \mathbb{E} \left\{ D(t_k, \xi(t_k)) \right\} \mathbb{E} \left\{ w(t_k) \right\} = \\ &= L \mathbb{E} \left\{ \xi(t_k) \right\} + K \mathbb{E} \left\{ C(t_k, \xi(t_k)) \xi(t_k) \right\}, \\ &\implies L \mathbb{E} \left\{ \xi(t_k) \right\} = I - K \mathbb{E} \left\{ C(t_k, \xi(t_k)) \xi(t_k) \right\}, \end{aligned}$$

которое для случая матрицы  $C$ , не зависящей от  $\xi$ , или для случая, когда вектор  $\xi$  является детерминированным, переходит в матричное условие

$$L(t_k, R(t_k)) = I - K(t_k, R(t_k)) C(t_k, \xi(t_k)). \quad (1.21)$$

В случае рассматриваемой нами модельной системы матрица  $C$  очень слабо зависит от  $\xi$ . Так, заменив в выражении (1.11) для матрицы  $C_j^s(t_k, \xi(t_k))$  вектор  $x_i(t_k)$  на  $z_{ij}(t_k)$  или  $\bar{x}_i(t_k)$  (часть прогнозной оценки  $\bar{\xi}(t_k)$ ), мы получим близкое выражение, пригодное для использования в линейных алгоритмах. Далее, все алгоритмы будут рассматриваться с условием (1.21) с приближенной заменой  $x_i(t_k)$  на  $\bar{x}_i(t_k)$  в матрице  $C$  — это соответствует варианту Enhanced Kalman Filter (EKF) для нелинейной системы. При его подстановке в уравнение коррекции получается

$$\hat{\xi}(t_k) = \bar{\xi}(t_k) + K(t_k, R(t_k)) (z(t_k) - C(t_k, \bar{\xi}(t_k)) \bar{\xi}(t_k))$$

или в упрощённой записи

$$\hat{\xi}(t_k) = \bar{\xi}(t_k) + K (z(t_k) - C \bar{\xi}(t_k)). \quad (1.22)$$

Слагаемое  $C(t_k, \bar{\xi}(t_k)) \bar{\xi}(t_k)$  можно проинтерпретировать как прогнозное измерение на момент  $t_k$ . Таким образом в выражении оценки (1.22) фигурирует разность между действительным и прогнозом измерениями.

Далее, в разделах посвящённых алгоритмам параллельной фильтрации, поскольку все рассматриваемые соотношения касаются шага между моментами  $t_{k-1}$  и  $t_k$ , аргументы будут опускаться. Т. е. будут приняты обозначения

$$A = A(\mathcal{T}_k), \quad B = B(t_k), \quad C = C(t_k, \bar{\xi}(t_k)), \quad D = D(t_k, \bar{\xi}(t_k)).$$

Введём обозначение для матрицы ковариаций ошибки оценивания для прогнозной оценки

$$\begin{aligned} \bar{P}(t_k) &= \mathbf{Cov}\{\bar{\xi}(t_k) - \xi(t_k)\} = \\ &= \begin{bmatrix} \mathbf{Cov}\{\bar{\chi}_1(t_k) - \chi_1(t_k), \bar{\chi}_1(t_k) - \chi_1(t_k)\} & \cdots & \mathbf{Cov}\{\bar{\chi}_1(t_k) - \chi_1(t_k), \bar{\varsigma}_m(t_k) - \varsigma_m(t_k)\} \\ \vdots & \ddots & \vdots \\ \mathbf{Cov}\{\bar{\varsigma}_m(t_k) - \varsigma_m(t_k), \bar{\chi}_1(t_k) - \chi_1(t_k)\} & \cdots & \mathbf{Cov}\{\bar{\varsigma}_m(t_k) - \varsigma_m(t_k), \bar{\varsigma}_m(t_k) - \varsigma_m(t_k)\} \end{bmatrix} = \\ &= \begin{bmatrix} \bar{P}_{\chi_1\chi_1}(t_k) & \cdots & \bar{P}_{\chi_1\varsigma_m}(t_k) \\ \vdots & \ddots & \vdots \\ \bar{P}_{\varsigma_m\chi_1}(t_k) & \cdots & \bar{P}_{\varsigma_m\varsigma_m}(t_k) \end{bmatrix} = \begin{bmatrix} \bar{P}_{\chi\chi} & \bar{P}_{\chi\varsigma} \\ \bar{P}_{\varsigma\chi} & \bar{P}_{\varsigma\varsigma} \end{bmatrix} \quad (1.23) \end{aligned}$$

и для матрицы ковариации ошибки основной оценки по измерениям до момента  $t_k$  включительно

$$\begin{aligned} \hat{P}(t_k) &= \mathbf{Cov}\{\hat{\xi}(t_k) - \xi(t_k)\} = \\ &= \begin{bmatrix} \mathbf{Cov}\{\hat{\chi}_1(t_k) - \chi_1(t_k), \hat{\chi}_1(t_k) - \chi_1(t_k)\} & \cdots & \mathbf{Cov}\{\hat{\chi}_1(t_k) - \chi_1(t_k), \hat{\varsigma}_m(t_k) - \varsigma_m(t_k)\} \\ \vdots & \ddots & \vdots \\ \mathbf{Cov}\{\hat{\varsigma}_m(t_k) - \varsigma_m(t_k), \hat{\chi}_1(t_k) - \chi_1(t_k)\} & \cdots & \mathbf{Cov}\{\hat{\varsigma}_m(t_k) - \varsigma_m(t_k), \hat{\varsigma}_m(t_k) - \varsigma_m(t_k)\} \end{bmatrix} = \\ &= \begin{bmatrix} \hat{P}_{\chi_1\chi_1}(t_k) & \cdots & \hat{P}_{\chi_1\varsigma_m}(t_k) \\ \vdots & \ddots & \vdots \\ \hat{P}_{\varsigma_m\chi_1}(t_k) & \cdots & \hat{P}_{\varsigma_m\varsigma_m}(t_k) \end{bmatrix} = \begin{bmatrix} \hat{P}_{\chi\chi} & \hat{P}_{\chi\varsigma} \\ \hat{P}_{\varsigma\chi} & \hat{P}_{\varsigma\varsigma} \end{bmatrix}. \quad (1.24) \end{aligned}$$



Здесь символами  $\bar{P}_{\chi\chi}$ ,  $\bar{P}_{\chi\varsigma}$ ,  $\bar{P}_{\varsigma\varsigma}$  и т. д. обозначены большие блоки матриц, включающие все ковариации между соответствующими частями фазового вектора. Так, например,

$$\hat{P}_{\chi\varsigma} = \mathbb{C}\text{ov}\{\hat{\chi}(t_k) - \chi(t_k), \hat{\varsigma}(t_k) - \varsigma(t_k)\}.$$

Приведём общие уравнения для эволюции этих матриц. В силу уравнения (1.9) и (1.16) справедливо

$$\bar{\xi}(t_k) - \xi(t_k) = A\hat{\xi}(t_{k-1}) - A\xi(t_{k-1}) - Bv(t_k) = A\left(\hat{\xi}(t_{k-1}) - \xi(t_{k-1})\right) - Bv(t_k).$$

Следовательно, в силу независимости случайной ошибки динамики  $v(t_k)$  и ошибок оценивания  $\hat{\xi}(t_{k-1}) - \xi(t_{k-1})$ , зависящих от случайных величин  $v(t)$ ,  $w(t)$  при  $t \in \mathcal{T}_{k-1}$ , верно соотношение

$$\bar{P}(t_k) = \mathbb{E}\left\{(\bar{\xi}(t_k) - \xi(t_k))(\bar{\xi}(t_k) - \xi(t_k))^{\top}\right\} = A\hat{P}(t_{k-1})A^{\top} + BV B^{\top}. \quad (1.25)$$

Пусть выполнено условие несмещённости и уравнение коррекции (1.17) переходит в (1.22), тогда для произвольного матричного коэффициента  $K$  (без разницы каким образом полученного) справедливо

$$\begin{aligned} \hat{\xi}(t_k) - \xi(t_k) &= \bar{\xi}(t_k) - \xi(t_k) + K\left(z(t_k) - C\hat{\xi}(t_k)\right) = \\ &= \bar{\xi}(t_k) - \xi(t_k) + K\left(C\hat{\xi}(t_k) + w(t_k) - C\hat{\xi}(t_k)\right) = (I - KC)(\bar{\xi}(t_k) - \xi(t_k)) + Kw(t_k). \end{aligned}$$

Так же как и при выводе соотношения для  $\bar{P}(t_k)$ , можно утверждать о независимости случайной ошибки наблюдения  $w(t_k)$  и ошибок оценивания  $\bar{\xi}(t_k) - \xi(t_k)$ , так как последние зависят от случайных величин  $v(t)$ ,  $w(t)$  при  $t \in \mathcal{T}_{k-1}$  и от  $v(t_k)$ . Следовательно, верно соотношение

$$\begin{aligned} \hat{P}(t_k) &= \mathbb{E}\left\{\left(\hat{\xi}(t_k) - \xi(t_k)\right)\left(\hat{\xi}(t_k) - \xi(t_k)\right)^{\top}\right\} = \\ &= (I - KC)\bar{P}(t_k)(I - KC)^{\top} + KWK^{\top} \end{aligned} \quad (1.26)$$

известное как *формула Иозефа*.

Для критерия (1.15) известна формула

$$J(t_k) = \text{tr}\left\{h^{\top}\hat{P}(t_k)h\right\}. \quad (1.27)$$

### 1.3 Уравнения оптимальной фильтрации

Уравнения оптимальной фильтрации можно получить, минимизируя след матрицы  $\hat{P}(t_k)$  в соотношении (1.26) варьированием различных матричных коэффициентов  $K$ . При этом выводится коэффициент  $K^*(t_k)$  минимизирующий критерий на каждом шаге работы алгоритма.

$$K^*(t_k) = \bar{P}(t_k)C^{\top}(C\bar{P}(t_k)C^{\top} + DW D^{\top})^{-1}. \quad (1.28)$$

Интересно, что оптимальное значение  $K^*$  подходит и для любого  $h$  в формуле (1.27), т. е. соответствует равномерной по  $h$  оценке.

Полностью, с подстановкой соотношения (1.28), уравнения рекуррентной фильтрации называются уравнениями фильтра Калмана (или рекуррентной оценки Гаусса-Маркова для случая  $B = 0$ ). Приведём их полностью:

$$\begin{aligned}
 \bar{\xi}(t_k) &= A\hat{\xi}(t_{k-1}), \\
 \bar{P}(t_k) &= A\hat{P}(t_{k-1})A^\top + BVB^\top, \\
 \Lambda &= C\bar{P}(t_k)C^\top + DWD^\top, \\
 K^* &= \bar{P}(t_k)C^\top\Lambda^{-1}, \\
 \hat{\xi}(t_k) &= \bar{\xi}(t_k) + K^*(z(t_k) - C\bar{\xi}(t_k)), \\
 \hat{P}(t_k) &= (I - K^*C)\bar{P}(t_k)(I - K^*C)^\top + K^*WK^{*\top} = \\
 &= (I - K^*C)\bar{P}(t_k) = \bar{P}(t_k) - K^*\Lambda K^{*\top}.
 \end{aligned} \tag{1.29}$$

Матрица  $\Lambda$  является матрицей ковариации отклонения прогнозного измерения  $C\bar{\xi}(t_k)$  от действительного измерения  $z(t_k)$ . Два последних равенства в формуле для  $\hat{P}(t_k)$  широко известны в литературе по фильтру Калмана. Однако следует отдавать себе отчёт, что эти соотношения ориентированы на специфический выбор  $K$ , и в общем случае не верны.

Отметим, что в качестве дополнительных параметров  $R(t_k)$ , по которым пересчитывается коэффициент  $K$ , в данном случае выступают прогнозная  $\bar{P}(t_k)$  и действительная  $\hat{P}(t_k)$  матрицы ковариаций ошибок оценивания.

Рассмотрим важную особенность фильтра Калмана. Даже для изучаемой нами системы с её специфическим видом уравнения наблюдения (1.10) и матрицы  $C$  будет справедливо:

$$\bar{P}(t_k)C^\top = [\bar{P}_{\chi_i}(t_k)(C_i^\chi)^\top \quad \bar{P}_{\varsigma_j}(t_k)(C_j^\varsigma)^\top],$$

где под символами  $\bar{P}_{\chi_i}$ ,  $\bar{P}_{\varsigma_j}$  понимаются столбцы матрицы  $\bar{P}$ , соответствующие переменным  $\chi_i$  и  $\varsigma_j$ . В коэффициенте  $K^*$  активными (не равными нулю) являются все строки для всех движений ВС и всех параметров радиолокаторов, несмотря на то, что текущее измерение связано с конкретным ВС и конкретным радиолокатором. Т. е. коррекция оценки  $\bar{\xi}$  затрагивает все переменные, и коррекция матрицы ковариации  $\bar{P}$  производится по всем строкам и столбцам. Это является общим свойством для фильтрации Калмана — этап коррекции затрагивает все переменные и всю матрицу ковариаций.

В отличие от этапа коррекции, на этапе прогноза (экстраполяции) в фильтре Калмана не происходит такого значительного «перемешивания» различных частей фазового вектора. Рассматриваемая нами система является системой с «разделённой» динамикой (1.9), в которой части фазового вектора  $\chi_i$ ,  $\varsigma_j$  эволюционируют независимо друг от друга. Легко проверить, что в силу специфического вида матриц  $A$  и  $B$  соотношения этапа прогноза (1.29) легко «распараллеливаются»:

$$\begin{aligned}
 \bar{\xi}(t_k) = A\hat{\xi}(t_{k-1}) \quad \Rightarrow \quad & \bar{\chi}_i(t_k) = A_i\hat{\chi}_i(t_{k-1}), \\
 & \bar{\varsigma}_j(t_k) = A_j^\varsigma\hat{\varsigma}_j(t_{k-1}), \\
 \bar{P}(t_k) = A\hat{P}(t_{k-1})A^\top + BVB^\top \quad \Rightarrow \quad & \bar{P}_{\chi_i\chi_i}(t_k) = A_i\hat{P}_{\chi_i\chi_i}(t_{k-1})A_i^\top + B_iV_i^\chi B_i^\top, \\
 & \bar{P}_{\varsigma_j\varsigma_j}(t_k) = A_j^\varsigma\hat{P}_{\varsigma_j\varsigma_j}(t_{k-1})A_j^{\varsigma\top} + B_j^\varsigma V_j^\varsigma B_j^{\varsigma\top}.
 \end{aligned} \tag{1.30}$$

Таким образом, в задаче одновременного оценивания положения нескольких ВС и систематических ошибок РЛС, как и в других задачах с «разделённой» динамикой,

главным препятствием к параллельному применению алгоритмов фильтрации является именно этап коррекции. Содержательно можно сказать, что коррекция систематических ошибок по итогам некоторого измерения вынуждает несколько изменить оценки положения всех ВС, которые были сделаны по предыдущей, до коррекции, оценке систематических ошибок.

Рассмотрим соотношения фильтра Калмана в варианте, разделённом по фазовым переменным. Здесь и далее матрицы ковариаций разделяются на блоки, соответствующие частям фазового вектора  $\chi$  и  $\varsigma$ : Уравнение наблюдения:

$$z(t_k) = C_i^x \chi_i(t_k) + C_j^s \varsigma_j(t_k) + D_j w_j.$$

Фильтр для переменных, описывающих движение.

Этап предсказания:

$$\begin{aligned}\bar{\chi}(t_k) &= A^x \hat{\chi}(t_{k-1}), \\ \bar{P}_{\chi\chi}(t_k) &= A^x \hat{P}_{\chi\chi}(t_{k-1}) A^{x\top} + B^x V^x B^{x\top}.\end{aligned}$$

Этап коррекции:

$$\begin{aligned}K_\chi &= \left( \bar{P}_{\chi\chi_i}(t_k) C_i^{x\top} + \bar{P}_{\chi\varsigma_j}(t_k) C_j^{s\top} \right) \Lambda^{-1}, \\ \hat{\chi}(t_k) &= \bar{\chi}(t_k) + K_\chi \left( z(t_k) - C_i^x \bar{\chi}_i(t_k) - C_j^s \bar{\varsigma}_j(t_k) \right), \\ \hat{P}_{\chi\chi}(t_k) &= \bar{P}_{\chi\chi}(t_k) - K_\chi \Lambda K_\chi^\top.\end{aligned}\tag{1.31}$$

Фильтр для систематической ошибки.

Этап предсказания:

$$\begin{aligned}\bar{\varsigma}(t_k) &= A^s \hat{\varsigma}(t_{k-1}), \\ \bar{P}_{\varsigma\varsigma}(t_k) &= A^s \hat{P}_{\varsigma\varsigma}(t_{k-1}) A^{s\top} + B^s V^s B^{s\top}.\end{aligned}$$

Этап коррекции:

$$\begin{aligned}K_\varsigma &= \left( \bar{P}_{\chi\varsigma_i}(t_k)^\top C_i^{x\top} + \bar{P}_{\varsigma\varsigma_j}(t_k) C_j^{s\top} \right) \Lambda^{-1}, \\ \hat{\varsigma}(t_k) &= \bar{\varsigma}(t_k) + K_\varsigma \left( z(t_k) - C_i^x \bar{\chi}_i(t_k) - C_j^s \bar{\varsigma}_j(t_k) \right), \\ \hat{P}_{\varsigma\varsigma}(t_k) &= \bar{P}_{\varsigma\varsigma}(t_k) - K_\varsigma \Lambda K_\varsigma^\top.\end{aligned}\tag{1.32}$$

Обновление блока кросс-ковариации:

$$\begin{aligned}\bar{P}_{\chi\varsigma}(t_k) &= A^x \hat{P}_{\chi\varsigma}(t_{k-1}) A^{s\top}, \\ \hat{P}_{\chi\varsigma}(t_k) &= \bar{P}_{\chi\varsigma}(t_k) - K_\chi \Lambda K_\varsigma^\top.\end{aligned}$$

Для обоих фильтров используется одна матрица  $\Lambda$ :

$$\begin{aligned}\Lambda &= C_i^x \bar{P}_{\chi\chi_i}(t_k) C_i^{x\top} + C_i^x \bar{P}_{\chi\varsigma_j}(t_k) C_j^{s\top} + \\ &\quad + C_j^s \bar{P}_{\chi\varsigma_j}(t_k)^\top C_i^{x\top} + C_j^s \bar{P}_{\varsigma\varsigma_j}(t_k) C_j^{s\top} + D_j W_j D_j^\top.\end{aligned}\tag{1.33}$$

## 1.4 Упрощенные алгоритмы оценивания по Henk Blom

В статье [2] также как и в данном отчёте рассматривается задача одновременного оценивания движения многих ВС и определения систематических ошибок. Приводятся варианты упрощения алгоритма фильтрации Калмана, показавшие хорошую работу на практике. В основе этих упрощений лежит простое предположение. Пусть мы рассматриваем уравнение коррекции (1.31). Давайте при этом считать, что неизвестное  $\varsigma$  на самом деле нам известно, т. е. оценка  $\bar{\varsigma}$  не является случайной величиной и совпадает с  $\varsigma$ . Это может быть дальнейшим образом обобщено тем, что ковариации  $\bar{P}_{\chi\varsigma}$  и  $\bar{P}_{\varsigma\chi}$  равны нулю при вычислении текущей оценки  $\hat{\chi}$  переменной  $\chi$  (т. е. как бы не существуют). Это приводит к следующим приближённым формулам:

$$\begin{aligned} \bar{P}_{\chi\varsigma}(t_k)C^\varsigma^\top &= \bar{P}_{\chi\varsigma_j}(t_k)C_j^\varsigma^\top = 0, \quad \bar{P}_{\varsigma\chi}(t_k) = 0, \quad \implies \\ K_\chi(t_k) &= \bar{P}_{\chi\chi}(t_k)C^\chi^\top\Lambda_\chi^{-1} = \bar{P}_{\chi\chi_i}(t_k)C_i^\chi^\top\Lambda_\chi^{-1}, \end{aligned} \quad (1.34)$$

$$\Lambda_\chi = C^\chi\bar{P}_{\chi\chi}(t_k)C^\chi^\top + DW D^\top = C_i^\chi\bar{P}_{\chi\chi_i}(t_k)C_i^\chi^\top + D_jW_jD_j^\top. \quad (1.35)$$

Рассмотрим формулу (1.34). Предположим, что все кросс-ковариации между параметрами движения  $i$  и параметрами других движений равны нулю на момент, предшествующий коррекции:

$$\bar{P}_{\chi_{i_1}\chi_i} = 0, \quad \forall i_1 \neq i.$$

Тогда при применении коэффициента  $K_\chi$  из формулы (1.34) справедливо

$$K_{\chi_{i_1}}(t_k) = 0, \quad K_{\chi_i}(t_k) = \bar{P}_{\chi\chi_i}(t_k)C_i^\chi^\top\Lambda_\chi^{-1}. \quad (1.36)$$

Подставляя это конкретное выражение для  $K_\chi$  в уравнения (1.31), получаем, что для всех  $i_1 \neq i$  и любых  $l = 1, \dots, n$  справедливо

$$\hat{\chi}_{i_1} = \bar{\chi}_{i_1}, \quad \hat{P}_{\chi_{i_1}\chi_l} = \bar{P}_{\chi_{i_1}\chi_l}.$$

Это означает, что параметры движений  $i_1 \neq i$  не изменяются при учёте измерения  $z(t_k)$ , изменению подвергается только параметры движения, которое непосредственно наблюдается. Кроме того, из этого следует, что кросс-ковариации между параметрами остальных движений и параметрами движения  $i$  остаются равными нулю и после коррекции. Следовательно, если в начальный момент обеспечить для всех  $i_1 \neq i_2$  нулевые кросс-ковариации

$$\bar{P}_{\chi_{i_1}\chi_{i_2}} = 0,$$

то и дальше они останутся нулевыми, а каждое движение можно будет обрабатывать отдельным фильтром в параллельном режиме.

Аналогично можно поступать и в ходе коррекции  $\varsigma$ , считая при этом не случайными величинами текущие оценки  $\bar{\chi}$  параметров движения ВС. Это приводит к приближённым соотношениям, аналогичным соотношениям (1.34), (1.35), записанным для параметров систематических ошибок  $\varsigma$ :

$$\begin{aligned} \bar{P}_{\varsigma\chi}(t_k)C^\chi^\top &= \bar{P}_{\varsigma\chi_i}(t_k)C_i^\chi^\top = 0, \quad \bar{P}_{\chi\chi}(t_k) = 0, \quad \implies \\ K_\varsigma(t_k) &= \bar{P}_{\varsigma\varsigma}(t_k)C^\varsigma^\top\Lambda_\varsigma^{-1} = \bar{P}_{\varsigma\varsigma_j}(t_k)C_j^\varsigma^\top\Lambda_\varsigma^{-1}, \end{aligned} \quad (1.37)$$

$$\Lambda_\varsigma = C^\varsigma\bar{P}_{\varsigma\varsigma}(t_k)C^\varsigma^\top + DW D^\top = C_j^\varsigma\bar{P}_{\varsigma\varsigma_j}(t_k)C_j^\varsigma^\top + D_jW_jD_j^\top. \quad (1.38)$$

В случае пересчёта коэффициента  $K_\varsigma$  по правилу (1.37) точно также как и для  $\chi$  появляется возможность параллельного пересчёта отдельных параметров  $\varsigma_j$ . Для этого нужно лишь обеспечить, чтобы в начальный момент времени для кросс-ковариаций выполнялось условие

$$\bar{P}_{\varsigma_1 \varsigma_2} = 0.$$

#### 1.4.1 Разделённые фильтры для фазового вектора и для систематической ошибки

Опишем подробно соотношения параллельной фильтрации, с учётом упрощённого вычисления коэффициента  $K_\chi$  по формулам (1.34), (1.35) и коэффициента  $K_\varsigma$  по формулам (1.37), (1.38). Напомним, что уравнение наблюдения для текущего измерения выглядит следующим образом:

$$z(t_k) = C_i^\chi \chi_i(t_k) + C_j^\varsigma \varsigma_j(t_k) + D_j w_j.$$

Фильтр для переменных  $\chi_i$ , описывающих движение.  $i = 1, \dots, n$ .

Этап предсказания:

$$\begin{aligned} \bar{\chi}_i(t_k) &= A_i \hat{\chi}_i(t_{k-1}), \\ \bar{P}_{\chi_i \chi_i}(t_k) &= A_i \hat{P}_{\chi_i \chi_i}(t_{k-1}) A_i^\top + B_i V_i^\chi B_i^\top. \end{aligned}$$

Этап коррекции:

$$\begin{aligned} \hat{\chi}_i(t_k) &= \bar{\chi}_i(t_k) + K_{\chi_i} (z(t_k) - C_i^\chi \bar{\chi}_i(t_k) - C_j^\varsigma \bar{\varsigma}_j(t_k)), \\ \hat{P}_{\chi_i \chi_i}(t_k) &= \bar{P}_{\chi_i \chi_i}(t_k) - K_{\chi_i} \Lambda_{\chi_i} K_{\chi_i}^\top. \end{aligned}$$

Аппроксимация:

$$\begin{aligned} K_{\chi_i} &= \bar{P}_{\chi_i \chi_i}(t_k) C_i^{\chi \top} \Lambda_{\chi_i}^{-1}, \\ \Lambda_{\chi_i} &= C_i^\chi \bar{P}_{\chi_i \chi_i}(t_k) C_i^{\chi \top} + D_j W_j D_j^\top. \end{aligned}$$

Фильтр для параметров систематической ошибки  $\varsigma_j$ .  $j = 1, \dots, m$ .

Этап предсказания:

$$\begin{aligned} \bar{\varsigma}_j(t_k) &= A_j^\varsigma \hat{\varsigma}_j(t_{k-1}), \\ \bar{P}_{\varsigma_j \varsigma_j}(t_k) &= A_j^\varsigma \hat{P}_{\varsigma_j \varsigma_j}(t_{k-1}) A_j^{\varsigma \top} + B_j^\varsigma V_j^\varsigma B_j^{\varsigma \top}. \end{aligned}$$

Этап коррекции:

$$\begin{aligned} \hat{\varsigma}_j(t_k) &= \bar{\varsigma}_j(t_k) + K_{\varsigma_j} (z(t_k) - C^\chi \bar{\chi}_i(t_k) - C_j^\varsigma \bar{\varsigma}_j(t_k)), \\ \hat{P}_{\varsigma_j \varsigma_j}(t_k) &= \bar{P}_{\varsigma_j \varsigma_j}(t_k) - K_{\varsigma_j} \Lambda_{\varsigma_j} K_{\varsigma_j}^\top. \end{aligned}$$

Аппроксимация:

$$\begin{aligned} K_{\varsigma_j} &= \bar{P}_{\varsigma_j \varsigma_j}(t_k) C_j^{\varsigma \top} \Lambda_{\varsigma_j}^{-1}, \\ \Lambda_{\varsigma_j} &= C_j^\varsigma \bar{P}_{\varsigma_j \varsigma_j}(t_k) C_j^{\varsigma \top} + D_j W_j D_j^\top. \end{aligned}$$

### 1.4.2 Фильтр Калмана для фазового вектора, макро-фильтр для систематической ошибки

Кроме одновременного применения упрощённых формул (1.34), (1.35), (1.37), (1.38) для коэффициентов  $K_\chi$  и  $K_\varsigma$  в статье [2] приводится ещё один «промежуточный» вариант упрощения. В этом варианте рассматривается ситуация, когда за движением  $i$  в некоторый момент  $t_k$  одновременно наблюдают несколько радиолокаторов. Будем считать для простоты, что это радиолокаторы с номерами от 1 до  $m' \leq m$ . При этом упрощение и, как следствие, «распараллеливание» выполняется только для переменных  $\chi_i$ . Для параметров систематических ошибок  $\varsigma_j$  принимаются дополнительные упрощающие формулы, позволяющие не пересчитывать кросс-ковариации  $\bar{P}_{\varsigma\chi}$ . Не останавливаясь на подробной интерпретации метода, приведём вычислительные соотношения.

Напоминаем, что уравнение наблюдения выглядит следующим образом:

$$z(t_k) = C_i^x \chi_i(t_k) + C_j^\varsigma \varsigma_j(t_k) + D_j w_j.$$

Фильтр для переменных, описывающих движение.

Этап предсказания:

$$\begin{aligned} \bar{\chi}_i(t_k) &= A_i \hat{\chi}_i(t_{k-1}), \\ \bar{P}_{\chi_i \chi_i}(t_k) &= A_i \hat{P}_{\chi_i \chi_i}(t_{k-1}) A_i^\top + B_i V_i^x B_i^\top. \end{aligned}$$

Этап коррекции:

$$\begin{aligned} \hat{\chi}_i(t_k) &= \bar{\chi}_i(t_k) + K_{\chi_i} (z(t_k) - C_i^x \bar{\chi}_i(t_k) - C_j^\varsigma \bar{\varsigma}_j(t_k)), \\ \hat{P}_{\chi_i \chi_i}(t_k) &= \bar{P}_{\chi_i \chi_i}(t_k) - K_{\chi_i} \Lambda_{\chi_i} K_{\chi_i}^\top. \end{aligned}$$

Аппроксимация:

$$\begin{aligned} K_{\chi_i} &= \bar{P}_{\chi_i \chi_i}(t_k) C_i^{x\top} \Lambda_{\chi_i}^{-1}, \\ \Lambda_{\chi_i} &= C_i^x \bar{P}_{\chi_i \chi_i}(t_k) C_i^{x\top} + D_j W_j D_j^\top. \end{aligned}$$

Фильтр для систематической ошибки.

Этап предсказания:

$$\begin{aligned} \bar{\varsigma}(t_k) &= A^\varsigma \hat{\varsigma}(t_{k-1}), \\ \bar{P}_{\varsigma\varsigma}(t_k) &= A^\varsigma \hat{P}_{\varsigma\varsigma}(t_{k-1}) A^{\varsigma\top} + B^\varsigma V^\varsigma B^{\varsigma\top}. \end{aligned}$$

Этап коррекции:

$$\begin{aligned} \hat{\varsigma}(t_k) &= \bar{\varsigma}(t_k) + K_\varsigma (z(t_k) - C_i^x \bar{\chi}_i(t_k) - C_j^\varsigma \bar{\varsigma}_j(t_k)), \\ \hat{P}_{\varsigma\varsigma}(t_k) &= \bar{P}_{\varsigma\varsigma}(t_k) - K_\varsigma \Lambda_\varsigma K_\varsigma^\top. \end{aligned}$$

В вычислении матриц  $K_\varsigma$  и  $\Lambda_\varsigma$  используются аппроксимация члена  $C^x \bar{P}_{\chi\varsigma}(t_k)$ :

$$\begin{aligned} K_\varsigma &= (\bar{P}_{\varsigma\varsigma}(t_k) C^{\varsigma\top} + H^\top) \Lambda_\varsigma^{-1}, \\ \Lambda_\varsigma &= m' C^x \bar{P}_{\chi\chi}(t_k) C^{x\top} + \sum_{j_1, j_2=1}^{m'} C_{j_1}^\varsigma \bar{P}_{\varsigma_{j_1} \varsigma_{j_2}}(t_k) C_{j_2}^{\varsigma\top} + \sum_{j=1}^{m'} (H C_j^{\varsigma\top} + C_j^\varsigma H^\top + D_j W_j D_j^\top), \end{aligned}$$

где  $H$  строится следующим образом:

$$\begin{aligned} F_\chi &= \sum_{j=1}^{m'} (D_j W_j D_j^\top)^{-1}, \\ F_\varsigma &= \sum_{j=1}^{m'} (D_j W_j D_j^\top)^{-1} C_j^\varsigma, \\ H &= -(F_\chi^\top F_\chi)^{-1} F_\chi^\top F_\varsigma \bar{P}_{\varsigma\varsigma}(t_k). \end{aligned}$$

## 1.5 Алгоритм оценивания на основе СИ

Ключевое отличие алгоритмов, предложенных Henk Blom в статье [2], от алгоритма фильтрации Калмана состоит в том, что матричный коэффициент фильтра  $K$  не равен нулю только для тех переменных, которые непосредственно участвуют в уравнении наблюдения для текущего измерения. Это приводит к идее, что следует пытаться найти оптимальный фильтр, заранее удовлетворяющий такому свойству, которое в таком случае выступает как некоторое ограничение в задаче оптимизации.

## 2 Алгоритм многогипотезного восстановления траектории

**Данный** раздел отчёта посвящен исследованию задачи получения оценок параметров движения воздушного судна по поступающим радиолокационным замерам. В рассматриваемом подходе наряду с текущими показателями движения формируются наиболее вероятные варианты предыстории движения в виде модельных траекторий с заданными начальной точкой и управлениями. Рассматривается движение в горизонтальной плоскости. Модель движения – система дифференциальных уравнений 4-го порядка с ограничением на продольное и боковое ускорения.

Формируемая совокупность возможных траекторий движения воздушного судна с кусочно-постоянными управлениями используются для получения текущей средневзвешенной оценки текущего положения ВС. В общем случае такая задача является многоэкстремальной, даже в случае, когда для её решения используются замеры одного и того же воздушного судна. Поэтому является естественным использование в качестве оценки движения ВС нескольких вариантов траектории.

Предлагаемый подход во многом пересекается с подходом, используемым в настоящее время при мультирадарной обработке данных в НИТА, где задействуется фильтр Калмана и метод IMM. В частности, как и в IMM, в каждый момент времени рассматриваются различные варианты движения. Расчёт на улучшение показателей (по сравнению с методом IMM) по оценке текущих параметров движения сделан в первую очередь на дополнительное использование оценок предыстории в виде набора наиболее вероятных характерных вариантов (гипотез) движения ВС.

### 2.1 Постановка задачи

Рассматривается движение воздушного судна в горизонтальной плоскости. Требуется для задачи мультирадарной обработки оценить возможности использования подхода, опирающегося на способ представления траекторного движения воздушного судна в виде набора наиболее вероятных движений. В качестве текущей оценки положения ВС рассмотреть варианты выбора по "наилучшей траектории" и выбора по средней траектории с использованием принципа максимальной достоверности.

Формирование "пучка" траекторий должна производиться с условием максимальной представительности вариантов движения. Расчёты с ветвлением "пучка" и последующее прореживание с целью ограничения общего количества вероятных треков выполнять для каждого вновь поступившего замера.

Модель траекторного движения ВС должна соответствовать [ссылка на ГОСТ (есть) и сборник Красова (нашел)]

Провести моделирование на реальных и модельных данных. Для обработки реальных данных выполнить преобразование в местную горизонталь для некоторой средней точки.



## 2.2 Схема первоначального построения и пересчёта пучка траекторий

### 2.2.1 Построения в начальном окне

Начальное окно по времени, используемое для завязки траектории, имеет особый смысл для алгоритма. Длительность начального окна в текущей версии программы и далее в этом подразделе подразумевается заранее выбранным и фиксированным. Формирование пучка траекторий в начальном окне осуществляется по различным тройкам РЛС-замеров, удовлетворяющих следующим условиям:

- замеры попарно разнесены по времени не менее, чем на 8 сек.
- расстояние по геометрическим координатам для всех пар замеров составляет не меньше 800 м.
- при движении с постоянной по величине скоростью по окружности, соединяющей три точки геометрического положения РЛС-замеров, отклонение от замеров в соответствующие моменты времени не превосходит 800 м.

Предполагается, что в начальном окне существует хотя-бы одна такая тройка РЛС-замеров. Ограничение по боковому ускорению в явной форме не задаётся, хотя следует отметить, что оно обеспечивается неявно при выполнении перечисленных выше условий (если рассматривать движения с постоянными ускорениями).

По каждой допустимой тройке РЛС-замеров формируется трек с движением по окружности, проходящей через замеры, и с условием минимальной суммы отклонений по времени от замеров РЛС. Для каждого замера отклонение по времени рассчитывается как модуль разности между моментом РЛС-замера и моментом на траектории, который соответствует прохождению траектории через геометрическую отметку замера (Федотов: я здесь лишний раз отметил, подбираемая траектория проходит через геометрическую отметку замера не в соответствующий для замера момент времени, т.е.  $\text{Замер} = \text{Геом.Отм.} + \text{ВремяЗамера}$ ). Для получения соответствующих параметров движения в виде начальных координат, направления и величины скорости, а также величины бокового ускорения используются явные аналитические формулы и метод наименьших квадратов.

### 2.2.2 Ветвление пучка при поступлении нового замера

Данная операция выполняется для каждого вновь поступившего замера РЛС и по сути представляет собой формирование ведущих к данному замеру ответвлений от всех имеющихся треков с целью обеспечения представительности возможных вариантов движения ВС посредством получаемого пучка треков. То есть, по каждому треку формируется (если это возможно с учётом принятой модели движения) ещё один дополнительный трек, содержащий ответвление, которое ведёт в геометрическую точку замера. Формально количество треков таким образом может удвоиться.

В качестве ответвления рассматривается траектория в виде дуги окружности с нулевым продольным ускорением (с постоянной по величине скоростью). Практически для построения требуемой дуги используется та-же процедура формирования трека по трём точкам, что и в предыдущем подразделе. Отличие заключается лишь в фиксированных параметрах движения (координаты, направление и величина скорости) на начальный момент времени. Дополнительные две точки берутся непосредственно

с ответвляемого трека (см. рис.), при этом выбирается вариант с наибольшим результирующим весом у траектории с ответвлением.

Тут будет поясняющий рисунок.

### 2.2.3 Группировка и прореживание треков

При пересчёте пучка траекторий необходимо выполнять их прореживание, ввиду экспоненциального (с основанием два) роста числа вариантов треков по ходу поступления замеров РЛС за счёт операции ветвления (Федотов: Нов.Зам.РЛС + Ветвление => Удвоение числа треков). Здесь требуется, с одной стороны, сохранить представительность среди сохраняемых вариантов движения (Федотов: думаю тут можно без комментария, **представительность** не означает большое отличие между треками), а, с другой стороны, необходимо оставить только "лучшие" треки. Среди прочих рассматривался вариант, когда треки просто упорядочиваются по весу, а затем остаётся только некоторое количество лучших по весу треков. На практике этот подход оказался неудовлетворительным, поскольку пучок начинает состоять из практически одинаковых треков, которые отражают лишь какой-то один вариант движения, более предпочтительный на текущий момент времени. В этом случае переход на другой тип движения (с движения по окружности на движение по прямой и т.п.) приводит к повышенным смещённым ошибкам (Федотов: мне кажется так лучше).

В настоящее время как наиболее предпочтительная и несложная в реализации рассматривается идея, предполагающая разбиение треков на группы по рассчитываемой степени близости (см. раздел 2.5) с дальнейшим выбором в каждой группе наилучшего представителя

### 2.2.4 Расчёт оценки текущего положения ВС

– по лучшему треку

– по совокупности треков с весами

## 2.3 Математическая модель траекторного движения ВС

Алгоритм использует следующее модельное описание динамики самолёта:

$$\begin{aligned}\dot{x} &= v \cos \varphi, \\ \dot{z} &= v \sin \varphi, \\ \dot{\varphi} &= u/v, \\ \dot{v} &= w.\end{aligned}\tag{2.1}$$

Здесь  $x, z$  — координаты положения на плоскости; путевой угол  $\varphi$  — угол на плоскости между вектором скорости от оси  $x$ ;  $v$  — величина скорости ( $v > 0$ );  $u$  — боковое ускорение;  $w$  — продольное ускорение.

Предполагаем, что управления  $u, w$  стеснены геометрическими ограничениями  $|u| \leq u_{\max}, |w| \leq w_{\max}$ .

При практическом использовании интегрирования с построением трека в геометрических координатах используется сетка по времени, включающая все секундные отметки (это позволяет корректно сравнивать треки с несовпадающими моментами переключения).

### 2.3.1 Аналитические расчёты при интегрировании уравнений движения

Система (2.1) в случае постоянных  $u$ ,  $w$  является интегрируемой аналитически [6, 7].

Примем за начальный момент времени  $t = 0$ . Предположим, что в этот момент выполняются следующие начальные условия на фазовые координаты:

$$v(0) = v_0 > 0, \quad \varphi(0) = 0, \quad x(0) = z(0) = 0. \quad (2.2)$$

Тогда для момента времени  $t > 0$  при условии, что  $v > 0$  на всем промежутке от 0 до  $t$ , будут выполнены соотношения

$$\begin{aligned} v(t) &= v_0 + wt, \\ \varphi(t) &= \frac{u}{w} \ln \left( 1 + \frac{wt}{v_0} \right) = \frac{u}{w} \ln \frac{v(t)}{v_0}, \\ x(t) &= \frac{u}{u^2 + 4w^2} v^2(t) \sin \varphi(t) + \frac{2w}{u^2 + 4w^2} (v^2(t) \cos \varphi(t) - v_0^2), \\ z(t) &= -\frac{u}{u^2 + 4w^2} (v^2(t) \cos \varphi(t) - v_0^2) + \frac{2w}{u^2 + 4w^2} v^2(t) \sin \varphi(t). \end{aligned} \quad (2.3)$$

Для случая  $w = 0$  соотношения выводятся из (2.3) предельным переходом.

Другие начальные условия приводятся к (2.2) путём замен

$$\varphi \rightarrow \varphi + \varphi_0, \quad x \rightarrow x + x_0, \quad z \rightarrow z + z_0.$$

## 2.4 Структура данных программы, термины

*Участок постоянного управления* — промежуток времени, характеризующийся постоянством ускорений ВС. На нём фиксируются:  $u_i$  (поле записи **.Upr**) — значение поперечного управления;  $w_i$  (поле **.Wpr**) — значение продольного управления;  $t_{ni}$  (**.UTn**) — время начала участка постоянного управления;  $t_{ki}$  (**.UTk**) — время конца участка постоянного управления.

*Трек управления* — двунаправленный список участков постоянного управления. Каждый участок постоянного управления в треке управления должен содержать ссылки на предыдущий участок (поле записи **.Prd**) и последующий участок (поле **.Sld**). Величины  $t_{ni}$  и  $t_{ki}$  последовательных элементов списка должны быть согласованы, т. е.  $t_{ki-1} = t_{ni}$ . Также участки постоянного управления могут содержать поля **.nTr** и **.kTr** — ссылки на участок геометрического трека (определяется ниже), который был порождён рассматриваемым участком постоянного управления.

*Замер РЛС* содержит поля: **.Xzr** — координата замера «на север» в плоскости Земли; **.Zzr** — координата замера «на восток» в плоскости Земли; **.Hzr** — высота замера над плоскостью Земли; **.Tzr** — время замера; **.Nr1s** — номер РЛС замера.

*Трек замеров* — двунаправленный список замеров РЛС. Каждый замер РЛС в треке замеров должен содержать ссылки на предыдущий замер (поле записи **.Prd**) и последующий замер (поле **.Sld**).

*Точка геометрического трека* описывает положение ВС и содержит поля: **.X** — координата ВС «на север» в плоскости Земли; **.Z** — координата ВС «на восток» в плоскости Земли; **.H** — высота ВС над плоскостью Земли; **.Phi** — направление скорости ВС (угол по часовой стрелке относительно направления на север); **.V** — величина скорости ВС. Время точки геометрического трека в необходимых случаях вычисляется используя её положение в геометрическом треке.

*Геометрический трек* — двунаправленный список точек геометрического трека. Каждая точка геометрического трека должна содержать ссылки на предыдущий замер (поле записи `.Prd`) и последующий замер (поле `.Sld`). Вычисление геометрического трека по треку управления возможно при помощи процедуры `PostTrTrekasTekUchUprPrymo()`. Аргументом этой процедуры является первый элемент списка трека управления, этот элемент должен иметь поле `.nTr`, указывающее на точку геометрического трека. Эта точка используется как начальная при интегрировании уравнений движения ВС.

Треки всех видов (трек управления, трек замеров, геометрический трек) в программе хранятся в одном массиве

```
var MetkaTrekas:array[0..RazmerMassivaMetok]of TMetkaTrekas;
```

Тип `TMetkaTrekas` описан как запись с вариантной частью. Варианты соответствуют виду трека. Такой подход позволяет динамически распределять память между треками разных видов, использовать одни и те же процедуры манипулирования двунаправленным списком (добавление/удаление элемента, очистка трека и т. п.). Недостатком такого подхода служит большая вероятность возникновения ошибок программирования и сложность отладки.

Нужно ли здесь ещё что-то писать? ... ???

## 2.5 Вычисление веса треков и меры расхождения треков

Вес траектории используется для сравнения текущих треков с целью отбора наилучшего при условии их «близости» (меры расхождения). Вес трека связан с вероятностью движения по заданной совокупности замеров при условии, что часть замеров может содержать выбросы и быть ошибочной. Численно мера расхождения треков рассчитывается как среднее расстояние между треками в геометрических координатах, рассчитываемое на равномерной сетке времени в текущем расчётном окне.

!!! Формула

## 2.6 Алгоритм оптимизации

Алгоритм оптимизации основывается на методе Хука – Дживса [??].

В качестве входной информации алгоритм получает ссылку на трек управления (последовательность участков постоянного управления).

Алгоритм варьирует: значения продольного управления  $w_i$ , значения поперечного управления  $u_i$ , время переключения между участками постоянного управления  $t_{ni}$ . При этом учитываются ограничения: на абсолютные значения управления, на минимальную продолжительность постоянного управления. Целью варьирования является построение последовательности управлений, которые бы определяли геометрический трек с минимальным весом.

Для вычисления веса трека алгоритм формирует временный трек управления. В процессе формирования временного трека происходит проверка нарушения ограничений на управление, если ограничения нарушаются, то происходит возврат в основной алгоритм Хука – Дживса, при этом в качестве значения минимизируемой функции возвращается штраф пропорциональный величине нарушения ограничения (при этом при подаче на вход алгоритма оптимизации трека управления с нарушением ограничений возможно «скатывание» алгоритма в область, где ограничения не нарушаются). Для построения геометрического трека по сформированному треку управ-

ления используется обращение к процедуре `PostrTrekaSTekUchUpPrPrymo()` (которая производит интегрирование), затем вес трека вычисляется обращением к функции `RaschetVesaTreka()` и происходит возврат в основной алгоритм алгоритм Хука – Дживса с возвратом веса трека в качестве значения минимизируемой функции.

Варьирование прекращается когда текущие шаги варьирования оказываются меньше заданных финальных шагов варьирования.

Алгоритм возвращает ссылку на новый трек управления, получившийся в результате варьирования. В свою очередь трек управления содержит ссылки на соответствующий геометрический трек.

### Константы-параметры алгоритма оптимизации

Алгоритм использует следующие постоянные параметры (константы):

`Du1` = 0.5 – начальный шаг варьирования поперечного управления;

`Dw1` = 0.25 – начальный шаг варьирования продольного управления;

`Dt1` = 32.0 – начальный шаг варьирования разбивки времени;

`DuFin` = 0.01 – финальный шаг варьирования поперечного управления;

`DwFin` = 0.01 – финальный шаг варьирования продольного управления;

`DtFin` = 0.02 – финальный шаг варьирования разбивки времени;

`MAXu` = 4 ( $u_{\max}$ ) – максимальное значение поперечного управления;

`MAXw` = 2 ( $w_{\max}$ ) – максимальное значение продольного управления;

`dtmin` = 10.0 – минимальный промежуток времени постоянного управления;

`dh` = 0.5 – множитель уменьшения шага при неудаче в «поиске вокруг базовой точки» метода Хука – Дживса.

## 2.7 Программа конвертации данных

Программа `tracks_plots_00` предназначена для фильтрации и конвертации данных (РЛС, АЗН-В, монорадарная обработка, мультирадарная обработка) из текстовых файлов `tracks.txt`, `plots.txt`, `plots_ads.txt`, получаемых при помощи программы `vidparser.exe` из файлов `.vid`. Цель конвертации — получить небольшие по объёму текстовые файлы данных для использования в программе многогипотезного восстановления траектории.

В связи с тем, что исходные файлы имеют очень большой объём (гигабайты), полная загрузка информации из файлов в оперативную память (при использовании 32-битной ОС и 32-битного компилятора) не представляется возможной. Используются неоднократное чтение файлов и приближённая оценка сверху количества замеров в РЛС-треках.

### Алгоритм

- Первое чтение файла `tracks.txt`, сбор общей статистики по источникам, глобальным идентификаторам трека и т.п.
- Выделение памяти для АЗН-треков.
- Чтение файла `plots_ads.txt`. Заполнение массивов замеров АЗН.
- Упорядочение замеров внутри АЗН-треков по времени, запись треков в файлы `ads.new4` и `ads.plt`.

- Освобождение памяти для АЗН-треков.
- Выделение памяти (на основе приближённой оценки) для монорадарных треков (моно-треков), треков мультирадарной обработки (мульти-треков), РЛС-треков.
- Второе чтение файла `tracks.txt`. Заполнение массивов моно-треков и мульти-треков, заполнение номеров замеров в РЛС-треках.
- Упорядочение замеров внутри моно-треков и мульти-треков по времени, запись треков в файлы `*_mr.new4` и `*_mr.plt`.
- Чтение файла `plots.txt`. Заполнение массивов замеров РЛС-треков.
- Упорядочение замеров внутри РЛС-треков по времени, запись треков в файлы `*_r.new4` и `*_r.plt`.

### Особенности текущей версии, параметры программы

Воздушные суда идентифицируются по параметру `global` (глобальный номер трека), при этом в обработку идёт не более `AZN_TRACKS_MAX` воздушных судов, АЗН-треки которых имеют не менее `AZN_TRACKS_MIN_MEASUR` замеров (при этом из подходящих выбираются АЗН-треки с наибольшим числом замеров).

Из файлов `tracks.txt`, `plots.txt` учитываются РЛС-замеры и моно-замеры, параметр `sensor` которых равен параметру `RADAR_SENSOR_ONLY`.

Если после всех фильтраций трек содержит меньше `TRACKS_MIN_MEASUR` замеров, то он не записывается.

Выходные файлы записываются в поддиректорию `new`, которая не должна существовать до запуска программы. В директории `new` создаются поддиректории, совпадающие с параметром `global` записываемых ВС. Записываются файлы `ads.new4` и `ads.plt` (АЗН-треки), `240_mr.new4` и `240_mr.plt` (мульти-треки), `*_mr.new4` и `*_mr.plt` (моно-треки), `*_r.new4` и `*_r.plt` (РЛС-треки). Здесь `*` — номер РЛС.

Файлы `*.plt` имеют формат треков программы Ozi Explorer и используются для визуализации треков при помощи программы GPSTMapEdit.

### Формат файлов `.new4`

Текстовые файлы, в которых построчно записаны замеры. Столбцы: время замера, широта замера, долгота замера, высота замера. Столбцы разделяются символом табуляции.

Для РЛС-треков в качестве координат замеров используются поля `lat` и `lon` файла `plots.txt`, т.е. широта и долгота, сформированные из дальности и азимута сырых РЛС-замеров, которые предварительно корректируются с учётом текущей оценки систематических ошибок в программе мультирадарной обработки.

## Литература

- [1] Бедин Д. А., Денисов А. П., Иванов А. Г., Федотов А. А., Черетаев И. В., Ганебный С. А., Васильев А. В. Одновременное определение координат движущегося ВС и коррекция систематических ошибок РЛС при помощи фильтра Калмана. 2015. Отчёт по Договору 610-15У. 33 с.
- [2] Blom, H. A. P. and Van Doorn, B. A. Systematic Error Estimation in Multisensor Fusion Systems, Proceedings of SPIE — The International Society for Optical Engineering Vol. 1954, Oct. 1993, pp. 450–461.
- [3] ГОСТ 20058-80. Динамика летательных аппаратов в атмосфере. Термины, определения и обозначения. Издание официальное. Москва, Госкомстандарт, 1980.
- [4] Пятко С.Г., Красов А.И. и др. Автоматизированные системы управления воздушным движением. Санкт-Петербург, Изд. Политехника, 2004.
- [5] Бедин Д.А., Беляков А.В., Кумков С.И., Пацко В.С., Пятко С.Г., Строков К.В., Федотов А.А. Отчёт по договору №61/2007. Разработка алгоритмов и программ обработки и анализа информации в системе УВД (Этап 4). Том 2. Пакет алгоритмов для задачи восстановления траектории самолёта. 2008. 51 с.
- [6] Бедин Д.А., Беляков А.В., Кумков С.И., Пацко В.С., Пятко С.Г., Федотов А.А. Методы и алгоритмы анализа, выделения и отсеивания недостоверных замеров радиолокационной информации. Восстановление траектории самолета по зашумленным замерам в движущемся окне. 2006, Отчёт по Договору 53/2005. 27 с.
- [7] Бедин Д.А., Беляков А.В., Кумков С.И., Пацко В.С., Пятко С.Г., Строков К.В., Федотов А.А. Методы и алгоритмы анализа, выделения и отсеивания недостоверных замеров радиолокационной информации. Программный пакет алгоритмов восстановления траектории самолета по замерам с ошибкой. 2007, Отчёт по Договору 53/2005. 41 с.
- [8] Банди Б. Методы оптимизации (вводный курс). – М.: Радио и связь, 1988.