

**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«МЭИ»**

ИНСТИТУТ РАДИОТЕХНИКИ И ЭЛЕКТРОНИКИ

КАФЕДРА РАДИОТЕХНИЧЕСКИХ СИСТЕМ

**МЕТОДЫ ОПТИМАЛЬНОГО ПРИЕМА СИГНАЛОВ В АППАРАТУРЕ
ПОТРЕБИТЕЛЕЙ СРНС**

КОНТРОЛЬНАЯ РАБОТА №4

ФИО СТУДЕНТА: ЖЕРЕБИН В.Р.

ГРУППА: ЭР-15-15

ВАРИАНТ №: 3

ДАТА: 29.11.2019

ПОДПИСЬ: _____

ФИО ПРЕПОДАВАТЕЛЯ: ШАТИЛОВ А.Ю.

ОЦЕНКА: _____

МОСКВА, 2019 Г.

Дано

$\alpha = 1 \text{ с}^{-1}$ – ширина спектра флуктуаций радиального ускорения;

$T = 10 \text{ мс}$ – темп фильтрации, $T_d = 0,2 \text{ мкс}$ – темп работы АЦП;

$\omega_0 = 2\pi \cdot (1602 \text{ МГц})$ – несущая частота (для пересчетов);

$\omega_{\Pi} = 2\pi \cdot (2 \text{ МГц})$ – промежуточная частота;

$\sigma_{\alpha} = 10 \text{ м/с}^2$ – среднеквадратическое ускорение

$q_c/n_0 = 10^{0.1 \cdot (30 \text{ дБГц})} = 10^3 [\text{Гц}]$ – отношение мощности сигнала к спектральной плотности шума на входе приемника;

$c = 3 \cdot 10^8 \text{ м/с}^2$ – скорость света в свободном пространстве;

$S_{\xi} = 2\sigma_{\alpha}^2 \alpha \left(\frac{\omega_0}{c}\right)^2 = 2 \cdot 10^2 \cdot 1 \cdot \left(\frac{2\pi \cdot 1602 \cdot 10^6}{3 \cdot 10^8}\right)^2 = 2,25 \times 10^5$ – спектральная плотность мощности формирующего шума.

$\sigma_{\xi}^2 = \frac{S_{\xi}}{2T} = \frac{2,25 \times 10^5}{2 \cdot 10 \times 10^{-3}} = 1,126 \times 10^7$ – дисперсия формирующего шума.

$\sigma_n = \frac{a_0}{2\sqrt{q_c/n_0 T_d}} = \frac{1}{2\sqrt{10^3 \cdot 0,2 \times 10^{-6}}} = 35,4$ – СКО шума наблюдений.

Задание

1. Смоделировать входное воздействие и нелинейную систему фильтрации фазы и амплитуды в дискретном времени.

Входное воздействие:

$y_{k,i} = S_{k,i}(x_k) + n_{k,i}$ – в общем виде для нелинейного воздействия

$y_{k,i} = a_k \cos(\omega_{\Pi} t_{k,i} + \varphi_k) + n_{k,i}$ – частный случай нелинейного воздействия

Где $t_{k,i} = kT + iT_d$ – шкала времени с шагом дискретизации АЦП T_d

$n_{k,i}$ – ДБГШ с дисперсией σ_n^2

Нелинейную систему фильтрации фазы и частоты можно разбить на дискриминатор и фильтр. Дискриминатор осуществляет предварительную ЦОС на частоте дискретизации АЦП T_d и накапливает сигнал, то есть, в общем случае, выполняет роль коррелятора. Фильтр производит оценку информативных параметров с темпом изменения информационных параметров T .

$N = \frac{T}{T_d} = \frac{10 \times 10^{-3}}{0,2 \times 10^{-6}} = 50 \times 10^3$ – количество отсчетов дискриминатора на один отсчет фильтра

Уравнения динамики информационных параметров в дискретном виде:

$$a_k = a_{k-1} + \zeta_{k-1}T,$$

$$\varphi_k = \varphi_{k-1} + \Omega_{k-1}T,$$

$$\Omega_k = \Omega_{k-1} + \nu_{k-1}T,$$

$$\nu_k = \nu_{k-1} \cdot (1 - \alpha T) + \alpha T \cdot \xi_{k-1},$$

ζ_{k-1}, ξ_{k-1} , – ДБГШ с дисперсиями σ_{ζ}^2 и σ_{ξ}^2 соответственно

Введем вектор состояний информационных параметров:

$$\mathbf{x}_k = [a_k \quad \varphi_k \quad \Omega_k \quad \nu_k]^T$$

Фильтруемый процесс представим в виде многомерного марковского процесса:

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} + \mathbf{G}\boldsymbol{\xi}_{k-1}$$

С учетом уравнений динамики, выразим многомерные коэффициенты процесса:

$$\mathbf{F} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & T & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & (1 - \alpha T) \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} T & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & \alpha T \end{bmatrix}, \quad \boldsymbol{\xi}_{k-1} = \begin{bmatrix} \zeta_{k-1} \\ \xi_{k-1} \end{bmatrix}$$

Формирующие шумы амплитуды и фазы полагаем независимыми, следовательно:

$$\mathbf{D}_\xi = M[\boldsymbol{\xi}_k \boldsymbol{\xi}_{k-1}^T] = \begin{bmatrix} \sigma_\zeta^2 & 0 \\ 0 & \sigma_\xi^2 \end{bmatrix}$$

В наблюдениях используются следующие оцениваемые информационные параметры: амплитуда a_k и начальная фаза φ_k

$$y_{k,i} = a_k \cos(\omega_{\text{п}} t_{k,i} + \varphi_k) + n_{k,i}$$

Тогда связь информационных параметров с наблюдениями:

$$\boldsymbol{\lambda}_k = \begin{bmatrix} a_k \\ \varphi_k \end{bmatrix} = \mathbf{c}\mathbf{x}_k \rightarrow \mathbf{c} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

Основные уравнения фильтрации:

$$\mathbf{u}_{dk} = \begin{bmatrix} \frac{1}{\sigma_n^2} \sum_{i=0}^{N-1} y_{k,i} \cos(\omega_{\text{п}} t_{k,i} + \tilde{\varphi}_k) - \frac{\tilde{a}_k N}{2\sigma_n^2} \\ -\frac{\tilde{a}_k}{\sigma_n^2} \sum_{i=0}^{N-1} y_{k,i} \sin(\omega_{\text{п}} t_{k,i} + \tilde{\varphi}_k) \end{bmatrix} - \text{уравнение дискриминатора}$$

Шаг экстраполяции:

$$\tilde{\mathbf{x}}_k = \mathbf{F}\hat{\mathbf{x}}_{k-1}$$

$$\tilde{\mathbf{D}}_{x,k} = \mathbf{F}\mathbf{D}_{x,k}\mathbf{F}^T + \mathbf{G}\mathbf{D}_\xi\mathbf{G}^T$$

Шаг оценивания:

$$\mathbf{D}_{x,k} = (\tilde{\mathbf{D}}_{x,k}^{-1} + \mathbf{c}^T \mathbf{W}_k \mathbf{c})^{-1}$$

$$\hat{\mathbf{x}}_k = \tilde{\mathbf{x}}_k + \mathbf{D}_{x,k} \mathbf{c}^T \mathbf{u}_{dk}$$

Где \mathbf{W}_k – эквивалентная матрица весов наблюдений:

$$\mathbf{W}_k = \begin{bmatrix} W_{aa,k} & W_{a\varphi,k} \\ W_{\varphi a,k} & W_{\varphi\varphi,k} \end{bmatrix} \approx \begin{bmatrix} \frac{N}{2\sigma_n^2} & 0 \\ 0 & \frac{N\tilde{a}_k^2}{2\sigma_n^2} \end{bmatrix}$$

Начальные условия:

$$\mathbf{D}_0 = \begin{bmatrix} 0,3^2 & 0 & 0 & 0 \\ 0 & \pi^2 & 0 & 0 \\ 0 & 0 & (30 \text{ рад/с})^2 & 0 \\ 0 & 0 & 0 & (340 \text{ рад/с})^2 \end{bmatrix}, \quad \begin{bmatrix} a_0 \\ \varphi_0 \\ \Omega_0 \\ \nu_0 \end{bmatrix} = \begin{bmatrix} 1 \\ \pi/12 \\ 100 \\ 100 \end{bmatrix}, \quad \begin{bmatrix} \hat{a}_0 \\ \hat{\varphi}_0 \\ \hat{\Omega}_0 \\ \hat{\nu}_0 \end{bmatrix} = \begin{bmatrix} 0,5 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

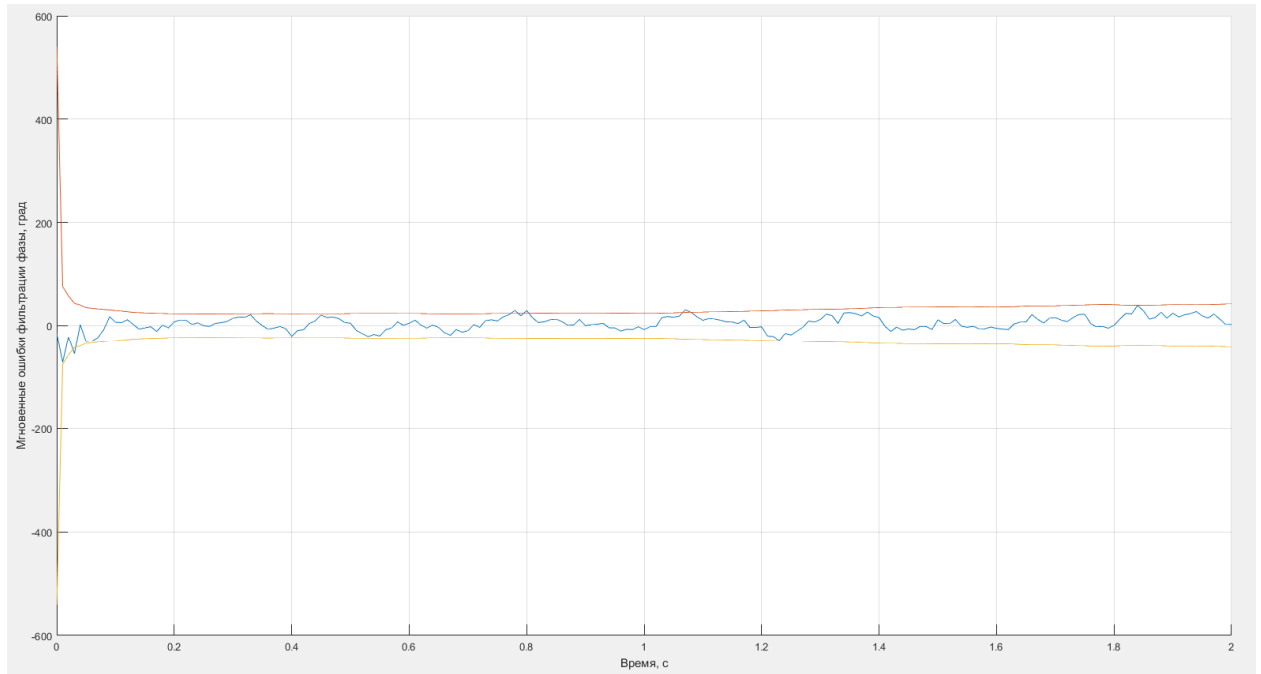
Амплитуда моделируется ступенькой:

$$a_k = \begin{cases} 1, & \text{при } t_k < 1\text{с} \\ 0,5, & \text{при } t_k \geq 1\text{с} \end{cases}$$

2. Графики временных зависимостей для ошибок фильтрации фазы

- мгновенной ошибки фильтрации фазы: $\varepsilon_\varphi(t_k) = \hat{\varphi}_k - \varphi_k$;
- предельные границы ошибок фильтрации фазы по уровню 3σ (по оценкам матрицы дисперсий фильтра $\mathbf{D}_{x,k}$):

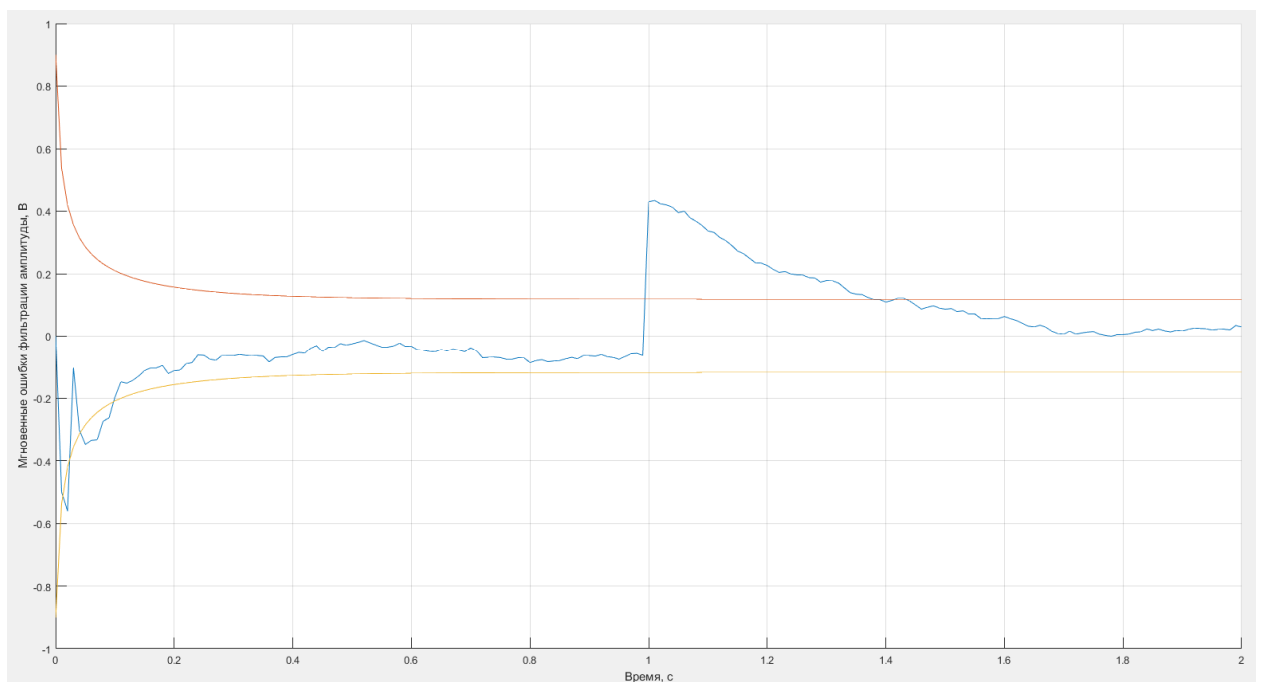
$$+3\sqrt{D_{22}(t_k)}, \quad -3\sqrt{D_{22}(t_k)}, \quad t = 0 \dots 2 \text{ с}$$



3. Графики временных зависимостей для ошибок фильтрации амплитуды

- мгновенной ошибки фильтрации амплитуды: $\varepsilon_a(t_k) = \hat{a}_k - a_k$;
- предельные границы ошибок фильтрации амплитуды по уровню 3σ (по оценкам матрицы дисперсий фильтра $\mathbf{D}_{x,k}$):

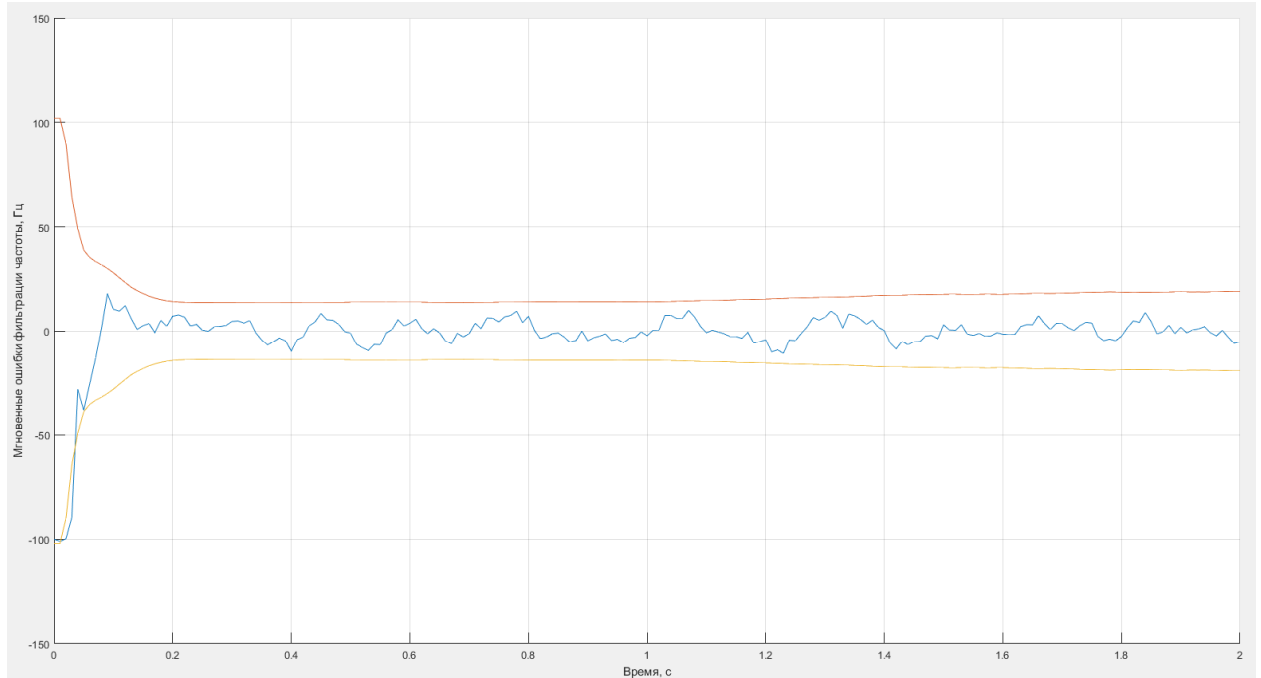
$$+3\sqrt{D_{11}(t_k)}, \quad -3\sqrt{D_{11}(t_k)}, \quad t = 0 \dots 2 \text{ с}$$



4. Графики временных зависимостей для ошибок фильтрации частоты

- мгновенной ошибки фильтрации частоты: $\varepsilon_{\Omega}(t_k) = \hat{\Omega}_k - \Omega_k$;
- предельные границы ошибок фильтрации частоты по уровню 3σ (по оценкам матрицы дисперсий фильтра $D_{x,k}$):

$$+3\sqrt{D_{33}(t_k)}, \quad -3\sqrt{D_{33}(t_k)}, \quad t = 0 \dots 2 \text{ с}$$



5. Общие выводы

Расширенный фильтр Калмана можно разбить на дискриминатор и фильтр, при условии, что информационные параметры сигнала меняются существенно медленнее, чем сам сигнал. В дискриминаторе происходит накопление сигнала и формирование корреляционных функций. В фильтре выполняются уравнения фильтрации по оптимальному алгоритму. Фильтр переходит в установившийся режим за $\sim 0,3$ с. При уменьшении амплитуды наблюдаемого сигнала, увеличиваются отношение сигнал/шум и, следовательно, предельные границы мгновенных ошибок фильтруемых параметров.

Приложение 1

Листинг программы MATLAB

```
close all; clear all; clc;
format long

%% Параметры
T = 10e-3;
Td = 0.2e-6;
N = T/Td;

t_start = 0;
t_stop = 2;
t = t_start:T:t_stop;
td = t_start:Td:t_stop;

q_dB = 30;
q = 10^(q_dB/10);

alpha = 1;
sigma_alpha = 10;
c = 3e8;

f0 = 1602e6;
omega0 = 2*pi*f0;
fp = 2e3;
omegap = 2*pi*fp;

%% Шум наблюдений
a0 = 1;
sigma_n = a0/(2*sqrt(q*Td));
D_n = sigma_n^2;

%% Формирующий шум
S = 2*sigma_alpha^2*alpha*(omega0/c)^2;
D_xi = S/(2*T);
D_zeta = 0.5;

%% Коэффициенты фильтра
F = [1 0 0 0;
      0 1 T 0;
      0 0 1 T;
      0 0 0 1-alpha*T];
G = [T 0;
      0 0;
      0 0;
      0 alpha*T];
C = [1 0 0 0;
      0 1 0 0];
D_f = [D_zeta 0;
        0 D_xi];

%% Начальные условия
x = [1; pi/12; 100; 100];
D = [.3^2 0 0 0;
      0 pi^2 0 0;
      0 0 34^2 0;
      0 0 0 340^2];
xf = [1.5; 0; 0; 0];

%% Выделение памяти и начальные приближения
```

```

a = nan(size(t));          a(1) = x(1);
phi = nan(size(t));        phi(1) = x(2);
OMEGA = nan(size(t));      OMEGA(1) = x(3);
a_extr = nan(size(t));     a_extr(1) = 0;
phi_extr = nan(size(t));   phi_extr(1) = 0;
OMEGA_extr = nan(size(t)); OMEGA_extr(1) = 0;
D11 = nan(size(t));        D11(1) = D(1,1);
D22 = nan(size(t));        D22(1) = D(2,2);
D33 = nan(size(t));        D33(1) = D(3,3);
D44 = nan(size(t));        D44(1) = D(4,4);

for k = 2:length(t)
    %% процесс
    x = F*x + G*randn(1,1)*sqrt(D_f);
    a(k) = x(1);
    phi(k) = x(2);
    OMEGA(k) = x(3);

    %% экстраполяция
    xf = F*xf;
    D = F*D*F' + G*D_f*G';
    W = N/(2*D_n)*[1 0;
                   0 xf(1)^2];

    a_extr(k) = xf(1);
    phi_extr(k) = xf(2);
    OMEGA_extr(k) = xf(3);

    %% Дискриминация
    for i = 1:N
        i_m = (k-2)*N+i;
        if i_m*Td <= 1
            ai = 1;
        else
            ai = 0.5;
        end
        y = ai*cos(omegap*i_m*Td + phi(k)) + randn(1,1)*sigma_n;
        I(i) = y*cos(omegap*i_m*Td + phi_extr(k));
        Q(i) = y*sin(omegap*i_m*Td + phi_extr(k));
    end
    U_d1 = sum(I) * (1/D_n) - (xf(1)*N)/(2*D_n);
    U_d2 = -sum(Q) * (xf(1)/D_n);
    u_d = [U_d1; U_d2];

    %% Оценка
    D = inv(inv(D) + C'*W*C);
    xf = xf + D*C'*u_d;

    D11(k) = D(1,1);
    D22(k) = D(2,2);
    D33(k) = D(3,3);
    D44(k) = D(4,4);

    if k*T <= 1
        ak = 1;
    else
        ak = 0.5;
    end

    epsilon_a(k) = (a_extr(k) - ak);
end

```

```

%epsilon_a = (a_extr - a);
epsilon_phi = (phi_extr - phi);
epsilon_OMEGA = (OMEGA_extr - OMEGA);

figure(1);
hold on, grid on;
plot(t, epsilon_a, t, [3*sqrt(D11); -3*sqrt(D11)])
xlabel("Время, с");
ylabel("Мгновенные ошибки фильтрации амплитуды, В");

figure(2);
hold on, grid on;
plot(t, epsilon_phi*180/pi, t, [3*sqrt(D22); -3*sqrt(D22)]*180/pi)
xlabel("Время, с");
ylabel("Мгновенные ошибки фильтрации фазы, град");

figure(3);
hold on, grid on;
plot(t, epsilon_OMEGA, t, [3*sqrt(D33); -3*sqrt(D33)])
xlabel("Время, с");
ylabel("Мгновенные ошибки фильтрации частоты, Гц");

```