

Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное  
учреждение высшего образования  
«НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

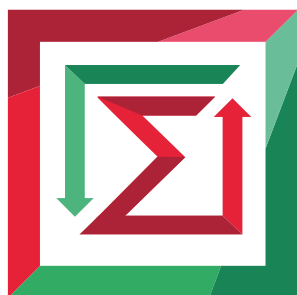


**НГТУ  
НЭТИ**

Кафедра теоретической и прикладной информатики

Лабораторная работа № 2  
по дисциплине «Методы активной идентификации динамических систем»

**ВЫЧИСЛЕНИЕ ИНФОРМАЦИОННОЙ МАТРИЦЫ ФИШЕРА**



ФАКУЛЬТЕТ:	ПМИ
Группа:	ПМИМ-01
Студенты:	Ершов П.К., Дорош А.Э.
ВАРИАНТ:	3
УРОВЕНЬ СЛОЖНОСТИ:	1
ПРЕПОДАВАТЕЛЬ:	Чубич В. М.

Новосибирск

2021

## 1. Цель работы

Научиться вычислять информационную матрицу Фишера (ИМФ).

## 2. ЗАДАНИЕ НА ЛАБОРАТОРНУЮ РАБОТУ

В соответствии с выбранным уровнем сложности разработать программу вычисления ИМФ.

1. Проверить правильность работы программы на специально подготовленном тестовом примере.
2. Вычислить ИМФ согласно варианту задания. Результаты привести в отчете.

## 3. Необходимые теоретические данные

**Алгоритм вычисления значения критерия идентификации:**

1. Определить  $F, \Psi, \Gamma, H, Q, R, \bar{x}(t_0), P(t_0)$  и

$$\left\{ \frac{\partial F}{\partial \theta_\alpha}, \frac{\partial \Psi}{\partial \theta_\alpha}, \frac{\partial \Gamma}{\partial \theta_\alpha}, \frac{\partial H}{\partial \theta_\alpha}, \frac{\partial Q}{\partial \theta_\alpha}, \frac{\partial R}{\partial \theta_\alpha}, \frac{\partial \bar{x}(t_0)}{\partial \theta_\alpha}, \frac{\partial P(t_0)}{\partial \theta_\alpha}, \alpha = 1, 2, \dots, s \right\}, \text{ где}$$

$F, \Psi, \Gamma, H, Q, R$  – матрицы модели, которые задаются в условиях задачи,  $\bar{x}(t_0)$  – начальный вектор состояний,  $P(t_0)$  – ковариационная матрица ошибок прогнозирования.

Сформировать матрицу  $\Psi_A$  по формуле (24) :  $\Psi_A = \begin{bmatrix} \Psi \\ \frac{\partial \Psi}{\partial \theta_1} \\ \dots \\ \frac{\partial \Psi}{\partial \theta_s} \end{bmatrix}$ .

2. Положить  $M(\theta) = 0$ ;  $k = 0$ ;  $P(t_k | t_k) = P(t_0)$ ,  
 $P(t_k | t_k) = P(t_0), \left\{ \frac{\partial P(t_k | t_k)}{\partial \theta_\alpha} = \frac{\partial P(t_0)}{\partial \theta_\alpha}, \alpha = 1, 2, \dots, s \right\}$
3. Определить  $u(t_k)$ .
4. Если  $k = 0$ , вычислить  $\bar{x}_A(t_{k+1})$  и  $\Sigma_A(t_{k+1})$  по формулам (21) и (22) соответственно, где

$$(21) \bar{x}_A(t_{k+1}) = \begin{cases} F_A(t_k) \bar{x}_A(t_k) + \Psi_A u(t_k), & \text{если } k = 1, 2, \dots, N-1; \\ \begin{bmatrix} F \bar{x}(t_0) + \Psi u(t_0) \\ \frac{\partial F}{\partial \theta_1} \bar{x}(t_0) + F \frac{\partial \bar{x}(t_0)}{\partial \theta_1} + \frac{\partial \Psi}{\partial \theta_1} u(t_0) \\ \dots \\ \frac{\partial F}{\partial \theta_s} \bar{x}(t_0) + F \frac{\partial \bar{x}(t_0)}{\partial \theta_s} + \frac{\partial \Psi}{\partial \theta_s} u(t_0) \end{bmatrix}, & \text{если } k = 0; \end{cases}$$

$$(22) \Sigma_A(t_{k+1}) = \begin{cases} F_A(t_k)\Sigma_A(t_k)F_A^T(t_k) + K_A(t_k)B(t_k)K_A^T(t_k), & \text{если } k = 1, \dots, N-1; \\ 0, & \text{если } k = 0; \end{cases}$$

тут

$$F_A(t_k) = \begin{bmatrix} F & 0 & \dots & 0 \\ \frac{\partial F}{\partial \theta_1} - \tilde{K}(t_k) \frac{\partial H}{\partial \theta_1} & F - \tilde{K}(t_k)H & \dots & 0 \\ \dots & \dots & \dots & \dots \\ \frac{\partial F}{\partial \theta_s} - \tilde{K}(t_k) \frac{\partial H}{\partial \theta_s} & 0 & \dots & F - \tilde{K}(t_k)H \end{bmatrix};$$

$$K_A(t_k) = \begin{bmatrix} \tilde{K}(t_k) \\ \frac{\partial \tilde{K}(t_k)}{\partial \theta_1} \\ \dots \\ \frac{\partial \tilde{K}(t_k)}{\partial \theta_s} \end{bmatrix};$$

$$\tilde{K}(t_k) = FK(t_k);$$

$$B(t_{k+1}) = HP(t_{k+1} | t_k) H^T + R;$$

$$K(t_{k+1}) = P(t_{k+1} | t_k) H^T B^{-1}(t_{k+1}),$$

после перейти на шаг 8.

5. Найти  $\tilde{K}(t_k)$  по формуле (26) и  $\left\{ \frac{\partial \tilde{K}(t_k)}{\partial \theta_i}, i = 1, 2, \dots, s \right\}$  по формуле

$$\frac{\partial \tilde{K}(t_k)}{\partial \theta_i} = \frac{\partial F}{\partial \theta_i} K(t_k) + F \frac{\partial K(t_k)}{\partial \theta_1}$$

$$(26) \tilde{K}(t_k) = FK(t_k)$$

6. Сформировать матрицы  $F_A(t_k)$  и  $K_A(t_k)$  по формулам (23) и (25) соответственно.

$$(23) F_A(t_k) = \begin{bmatrix} F & 0 & \dots & 0 \\ \frac{\partial F}{\partial \theta_1} - \tilde{K}(t_k) \frac{\partial H}{\partial \theta_1} & F - \tilde{K}(t_k)H & \dots & 0 \\ \dots & \dots & \dots & \dots \\ \frac{\partial F}{\partial \theta_s} - \tilde{K}(t_k) \frac{\partial H}{\partial \theta_s} & 0 & \dots & F - \tilde{K}(t_k)H \end{bmatrix};$$

$$(25) \quad K_A(t_k) = \begin{bmatrix} \tilde{K}(t_k) \\ \frac{\partial \tilde{K}(t_k)}{\partial \theta_1} \\ \dots \\ \frac{\partial \tilde{K}(t_k)}{\partial \theta_s} \end{bmatrix};$$

7. Вычислить  $\bar{X}_A(t_{k+1})$  и  $\Sigma_A(t_{k+1})$  по формулам (21) и (22) соответственно.

8. Найти  $P(t_{k+1} | t_k)$  по формуле (10) и  $\left\{ \frac{\partial P(t_{k+1} | t_k)}{\partial \theta_i}, i = 1, 2, \dots, s \right\}$  по формуле

$$\begin{aligned} \frac{\partial P(t_{k+1} | t_k)}{\partial \theta_i} = & \frac{\partial F}{\partial \theta_i} P(t_k | t_k) F^T + F \frac{\partial P(t_k | t_k)}{\partial \theta_i} F^T + F P(t_k | t_k) \frac{\partial F^T}{\partial \theta_i} \\ & + \frac{\partial \Gamma}{\partial \theta_i} Q \Gamma^T + \Gamma \frac{\partial Q}{\partial \theta_i} \Gamma^T + \Gamma Q \frac{\partial \Gamma^T}{\partial \theta_i} \end{aligned}$$

$$(10) \quad P(t_{k+1} | t_k) = F P(t_k | t_k) F^T + \Gamma Q \Gamma^T$$

Вычислить  $B(t_{k+1})$  по формуле (12) и  $\left\{ \frac{\partial B(t_{k+1})}{\partial \theta_i}, i = 1, 2, \dots, s \right\}$  по формуле

$$\begin{aligned} \frac{\partial B(t_{k+1})}{\partial \theta_i} = & \frac{\partial H}{\partial \theta_i} P(t_{k+1} | t_k) H^T + H \frac{\partial P(t_{k+1} | t_k)}{\partial \theta_i} H^T + \\ & + H P(t_{k+1} | t_k) \frac{\partial H^T}{\partial \theta_i} + \frac{\partial R}{\partial \theta_i} \end{aligned}$$

$$(12) \quad B(t_{k+1}) = H P(t_{k+1} | t_k) H^T + R$$

Определить  $K(t_{k+1})$  по формуле (13) и  $\left\{ \frac{\partial K(t_{k+1})}{\partial \theta_i}, i = 1, 2, \dots, s \right\}$  по формуле

$$\begin{aligned} \frac{\partial K(t_{k+1})}{\partial \theta_i} = & \left[ \frac{\partial P(t_{k+1} | t_k)}{\partial \theta_i} H^T + P(t_{k+1} | t_k) \frac{\partial H^T}{\partial \theta_i} \right. \\ & \left. - P(t_{k+1} | t_k) H^T B^{-1}(t_{k+1}) \frac{\partial B^{-1}(t_{k+1})}{\partial \theta_i} \right] B^{-1}(t_{k+1}) \end{aligned}$$

$$(13) \quad K(t_{k+1}) = P(t_{k+1} | t_k) H^T B^{-1}(t_{k+1})$$

Найти  $P(t_{k+1} | t_{k+1})$  по формуле (15) и  $\left\{ \frac{\partial P(t_{k+1} | t_{k+1})}{\partial \theta_i}, i = 1, 2, \dots, s \right\}$  по формуле

$$\begin{aligned} \frac{\partial P(t_{k+1} | t_{k+1})}{\partial \theta_i} = [I - K(t_{k+1})H] \frac{\partial P(t_{k+1} | t_k)}{\partial \theta_i} - \\ - \left[ \frac{\partial K(t_{k+1})}{\partial \theta_i} H + K(t_{k+1}) \frac{\partial H}{\partial \theta_i} \right] P(t_{k+1} | t_k) \end{aligned}$$

$$(15) P(t_{k+1} | t_{k+1}) = [I - K(t_{k+1})H] P(t_{k+1} | t_k).$$

9. Используя выражение (20), получить  $\Delta M(\theta)$ , отвечающее текущему значению  $k$ .

$$(20) M_{ij}(U, \theta) =$$

$$\begin{aligned} \sum_{k=0}^{N-1} \left\{ sp \left[ \frac{\partial H}{\partial \theta_i} C_0 (\bar{x}_A(t_k) \bar{x}_A^T(t_k) + \Sigma_A(t_k)) C_0^T \frac{\partial H^T}{\partial \theta_j} B^{-1}(t_{k+1}) \right] \right. \\ + sp \left[ \frac{\partial H}{\partial \theta_i} C_0 (\bar{x}_A(t_k) \bar{x}_A^T(t_k) + \Sigma_A(t_k)) C_j^T H^T B^{-1}(t_{k+1}) \right] \\ + sp \left[ H C_j (\bar{x}_A(t_k) \bar{x}_A^T(t_k) + \Sigma_A(t_k)) C_0^T \frac{\partial H^T}{\partial \theta_j} B^{-1}(t_{k+1}) \right] \\ + sp \left[ H C_j (\bar{x}_A(t_k) \bar{x}_A^T(t_k) + \Sigma_A(t_k)) C_j^T H^T B^{-1}(t_{k+1}) \right] \\ + sp \left[ \frac{\partial B(t_{k+1})}{\partial \theta_i} B^{-1}(t_{k+1}) \frac{\partial B(t_{k+1})}{\partial \theta_j} B^{-1}(t_{k+1}) \right], i, j \\ \left. = 1, \dots, s \right\} \end{aligned}$$

где матрица  $C_i = [O, \dots, O, I, O, \dots, O]$ ,

где блок  $O, \dots, O$  имеет длину  $i, i = 0, \dots, s$  и имеют размерность  $n \times n(s+1)$ , а вектор  $\bar{x}_A(t_k)$  вычисляется по формуле (21).

10. Положить  $M = M + \Delta M(\theta)$ .

11. Увеличить  $k$  на единицу. Если  $k \leq N - 1$ , перейти на шаг 3, иначе закончить процесс.

#### 4. Полученные результаты

Входные данные тестовой модели.

Исходя из указанной (для первого уровня сложности) модели:

$$\begin{aligned}x(t_{k+1}) &= \theta_1 x(t_k) + \theta_2 u(t_k) + w(t_k); \\ y(t_{k+1}) &= x(t_{k+1}) + v(t_{k+1}), k = 0, 1.\end{aligned}$$

$$\begin{aligned}E[w(t_i)w(t_k)] &= 0.1\delta_{ik} = Q\delta_{ik} \\ E[v(t_{i+1})v(t_{k+1})] &= 0.3\delta_{ik} = R\delta_{ik} \\ E[x(t_0)] &= 0 = \bar{x}(t_0) \\ E[x^2(t_0)] &= 0.1 = P(t_0). \\ U^T[u(t_0)u(t_1)] &= [2, 2] \\ \theta &= [\theta_1, \theta_2] = [1, 1].\end{aligned}$$

Можно получить следующие входные данные:

Матрицы для моделей состояния и измерения	Ковариационные матрицы шумов и начальные условия	$\theta^*, \Omega_\theta$	$U, \Omega_U$
$F = \theta_1,$ $\Psi = \theta_2,$ $\Gamma = 1,$ $H = 1$	$Q = 0.1, R = 0.3,$ $\bar{x}_0 = 0,$ $P_0 = 0.1$	$\theta^* = \begin{bmatrix} 1 \\ 1 \end{bmatrix},$	$U^T = [2, 2],$

*Результаты работы алгоритма:*

```
>> main
      8.3462   12.3077
      12.3077   27.6923
```

*Требуемое значение:*

$$M(\theta) = \begin{pmatrix} 8.35 & 12.31 \\ 12.31 & 27.70 \end{pmatrix}.$$

Входные данные для заданного варианта:

Матрицы для моделей состояния и измерения	Ковариационные матрицы шумов и начальные условия	$\theta^*, \Omega_\theta$	$U, \Omega_U$
$F = \begin{bmatrix} -0.8 & 1 \\ \theta_1 & 0 \end{bmatrix},$ $\Psi = \begin{bmatrix} 1 \\ 1 \end{bmatrix},$ $\Gamma = \begin{bmatrix} 1 \\ 1 \end{bmatrix},$ $H = [1 \quad 0]$	$Q = \theta_2, R = 0.1,$ $\bar{x}_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix},$ $P_0 = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix}$	$\theta^* = \begin{bmatrix} -1.5 \\ 0.5 \end{bmatrix},$ $\theta_1 \in [-2; -0.05],$ $\theta_2 \in [0.01; 0.8]$	$U^T = [3, \dots, 3],$ $u(t_k) \in [0; 5]$

Результаты работы алгоритма:

```
>> main
      0.0323   -0.1499
      -0.1499    1.7760
```

## 5. Текст программы

### Файл `init_data.m`

```
function [F, Psi, Gamma, H, Q, R, x0, P0, u] = init_data(theta)

    F = [-0.8, 1;
          theta(1), 0];
    Psi = [1; 1];
    Gamma = [1; 1];
    H = [1, 0];

    Q = theta(2);
    R = 0.1;
    x0 = [0; 0];
    P0 = [0.1, 0; 0, 0.1];
    u = 3;
end
```

### Файл `init_grad.m`

```
function [F_grad, Psi_grad, Gamma_grad, H_grad, Q_grad, R_grad, x0_grad,
P0_grad] = init_grad()

    F_grad = cell(2,1);
    F_grad{1} = [0, 0; 1, 0];
    F_grad{2} = [0, 0; 0, 0];

    Psi_grad = cell(2,1);
    Psi_grad{1} = [0;0];
    Psi_grad{2} = [0;0];

    Gamma_grad = cell(2,1);
    Gamma_grad{1} = [0;0];
    Gamma_grad{2} = [0;0];

    H_grad = cell(2,1);
    H_grad{1} = [0,0];
    H_grad{2} = [0,0];

    Q_grad = cell(2,1);
    Q_grad{1} = 0;
    Q_grad{2} = 1;

    R_grad = cell(2,1);
    R_grad{1} = 0;
    R_grad{2} = 0;

    x0_grad = cell(2,1);
    x0_grad{1} = [0;0];
    x0_grad{2} = [0;0];

    P0_grad = cell(2,1);
    P0_grad{1} = [0, 0; 0, 0];
    P0_grad{2} = [0, 0; 0, 0];
```



```
end
```

## Файл Get\_C.m

```
function [C] = Get_C(n, alpha, i)

    n2 = (alpha+1) * n;
    C = zeros(n, n2);

    C(:, n*i + 1 : n*(i+1)) = eye(n);

end
```

## Файл calculate\_Fisher\_Matrix.m

```
function [M] = calculate_Fisher_Matrix(theta, N)
    [F, Psi, Gamma, H, Q, R, x0, P0, u] = init_data(theta);
    [F_grad, Psi_grad, Gamma_grad, H_grad, Q_grad, R_grad, x0_grad, P0_grad]
= init_grad();

    n_1 = size(x0,1);
    n_2 = size(x0,2);
    alpha = length(theta);

    % выделение памяти
    M = zeros(alpha, alpha);
    dM = zeros(alpha, alpha);
    x_a = zeros(n_1 * (alpha + 1), n_2);
    sigma_a = zeros(alpha, alpha);
    P_kl_k_grad = cell(alpha,1);
    P_k_k_grad = cell(alpha, 1);
    B_grad = cell(alpha,1);
    K_grad = cell(alpha,1);
    K_grad_ = cell(alpha,1);
    P_kl_kl_grad = cell(alpha,1);
    F_a = zeros(n_1 * (alpha + 1), n_1 * (alpha + 1));
    K_a = zeros(n_1 * (alpha + 1), 1);

    % сформировать матрицу пси(a) по формуле 24

    Psi_a = zeros(n_1 * (alpha + 1), n_2);

    Psi_a(1:n_1, :) = Psi;

    for i = 1 : alpha
        Psi_a(n_1*i+1 : n_1*(i+1), :) = Psi_grad{i};
    end

    % задание начальных условий 2 шаг
    P_k = P0;
```

```

for i = 1 : alpha
    P_k_k_grad(i) = {P0_grad{i}};
end

for k = 0 : N - 1
    u = u;
    if (k == 0) % 4 шаг, затем 8

        x_a(1 : n_1*1, :) = F*x0 + Psi * u; % 21 u0

        for i = 1 : alpha
            x_a(n_1*i + 1 : n_1*(i+1), :) = F_grad{i} * x0 + F *
x0_grad{i} + Psi_grad{i} * u; % u0
        end

        sigma_a = zeros(n_1 * (alpha + 1), n_1 * (alpha + 1)); % 22
    end

    % 5-7 шаги
    if (k ~= 0)
        % 5 шаг
        K_ = F * K;
        for i = 1 : alpha
            K_grad_(i) = {F_grad{i} * K + F * K_grad{i}};
        end

        % 6 шаг

        F_a(1:n_1, 1:n_1) = F;
        for i = 1 : alpha
            F_a(n_1*i+1 : n_1*(i+1), 1:n_1) = F_grad{i} - K_ * H_grad{i};

            F_a(n_1*i+1 : n_1*(i+1), n_1*i+1 : n_1*(i+1)) = F - K_ * H;
        end

        K_a(1:n_1) = K;
        for i = 1 : alpha
            K_a(n_1*i+1 : n_1*(i+1)) = K_grad_{i};
        end

        % 7 шаг

        x_a = F_a * x_a + Psi_a * u;

        sigma_a = F_a * sigma_a * F_a' + K_a * B * K_a';

    end

    % 8 шаг

    P_k1_k = F * P_k * F' + Gamma * Q * Gamma'; % 10 формула

    for i = 1 : alpha
        P_k1_k_grad(i) = {F_grad{i} * P_k * F' + F * P_k_k_grad{i} * F'
...
            + F * P_k * F_grad{i}' + Gamma_grad{i} * Q * Gamma' + Gamma *
Q_grad{i} * Gamma' ...
            + Gamma * Q * Gamma_grad{i}'};
    end

```

```

end

B = H * P_kl_k * H' + R; % 12

for i = 1 : alpha
    B_grad(i) = {H_grad{i} * P_kl_k * H' + H * P_kl_k_grad{i} * H'
...
                + H * P_kl_k * H_grad{i}' + R_grad{i}};

end

K = P_kl_k * H' / B; % 13

for i = 1 : alpha
    K_grad(i) = {(P_kl_k_grad{i} * H' + P_kl_k * H_grad{i})' ...
                - P_kl_k * H' / B * B_grad{i}} / B;
end

P_kl = (eye(n_1) - K * H) * P_kl_k; % 15

for i = 1 : alpha
    P_kl_kl_grad(i) = {(eye(n_1) - K * H) * P_kl_k_grad{i} ...
                - (K_grad{i} * H + K * H_grad{i}) * P_kl_k};
end

% шаг 9
for i = 1 : alpha
    for j = 1 : alpha

        dM(i,j) = trace(H_grad{i} * Get_C(n_1, alpha, 0) * (x_a *
x_a' + sigma_a) * Get_C(n_1, alpha, 0)' * H_grad{j}' / B) ...
                + trace(H_grad{i} * Get_C(n_1, alpha, 0) * (x_a * x_a' +
sigma_a) * Get_C(n_1, alpha, j)' * H' / B) ...
                + trace(H * Get_C(n_1, alpha, i) * (x_a * x_a' + sigma_a)
* Get_C(n_1, alpha, 0)' * H_grad{j}' / B) ...
                + trace(H * Get_C(n_1, alpha, i) * (x_a * x_a' + sigma_a)
* Get_C(n_1, alpha, j)' * H' / B) ...
                + 0.5 * trace(B_grad{i} / B * B_grad{j} / B);

        end
    end

    M = M + dM;
    P_k = P_kl;
    P_k_k_grad = P_kl_kl_grad;
end
end

```

## Файл main.m

```

N = 2;
% истинная тета
theta_ist = [-1.5; 0.5];

M = calculate_Fisher_Matrix(theta_ist, N);
disp(M);

```