

## Тема 2. Последовательные алгоритмы обнаружения изменения свойств случайного процесса

Рассмотрим общую структуру системы диагностирования (рис. 1). На объект диагностирования (ОД) действуют внешние воздействия, приводящие к возникновению дефектов. Диагностические признаки (ДП), сформированные на основе модели ОД, подаются на блок, реализующий алгоритм принятия решения (АПР), который должен выдать сигнал о наличии или отсутствии дефекта.



Рис. 1. Общая структура системы диагностирования.

По способу получения информации об объекте диагноза различают апостериорный и последовательный анализ.



Один из возможных вариантов построения последовательных АПР состоит в формировании решающей функции ( $G(n)$ ) на основе ДП объекта ( $z(n)$ ) и сравнении ее с порогом  $h$ , что проиллюстрировано на рис. 2.

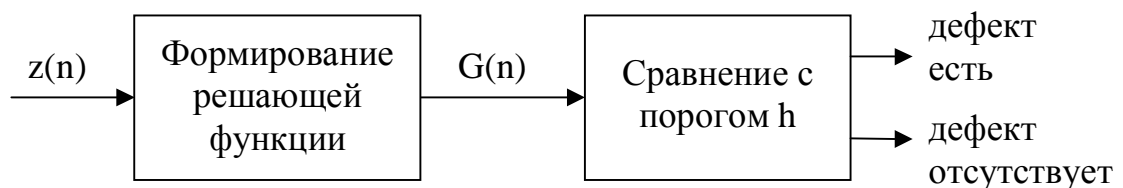


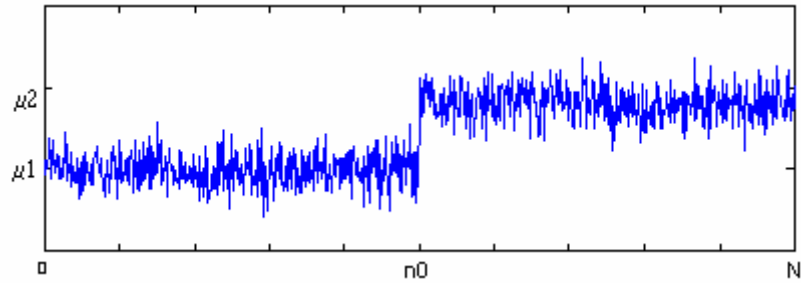
Рис. 2. Структура алгоритма принятия решения.

Результатом работы каждого алгоритма является решение о наличии или отсутствии дефекта в каждый момент времени. Для этого на каждом шаге по значениям сигнала  $z(n) \in N(m_1, b_1)$  вычисляется решающая функция  $G(n)$  и сравнивается с порогом (порогами).

## Типы дефектов

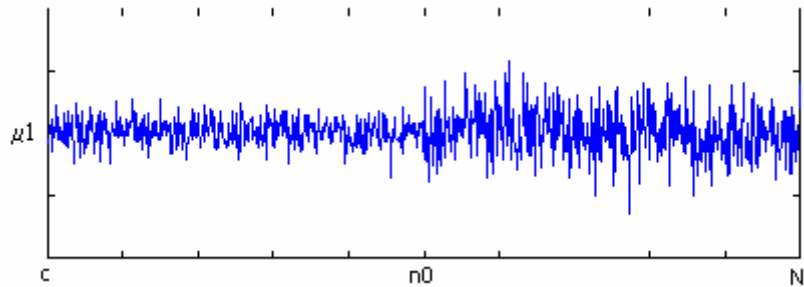
Дефект 1: однократное скачкообразное изменение математического ожидания.

$$\begin{cases} z(n) \in N(m_1, b_1), n \in [0, n_0] \\ z(n) \in N(m_2, b_1), n \in [n_0 + 1, N] \end{cases}$$



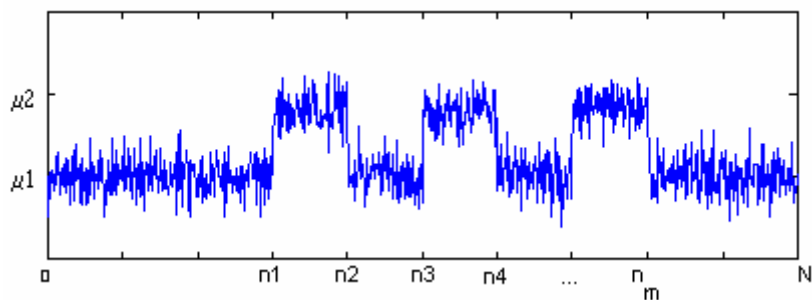
Дефект 2: однократное скачкообразное изменение дисперсии.

$$\begin{cases} z(n) \in N(m_1, b_1), n \in [0, n_0] \\ z(n) \in N(m_1, b_2), n \in [n_0 + 1, N] \end{cases}$$



Дефект 3: восстанавливаемое скачкообразное изменение математического ожидания.

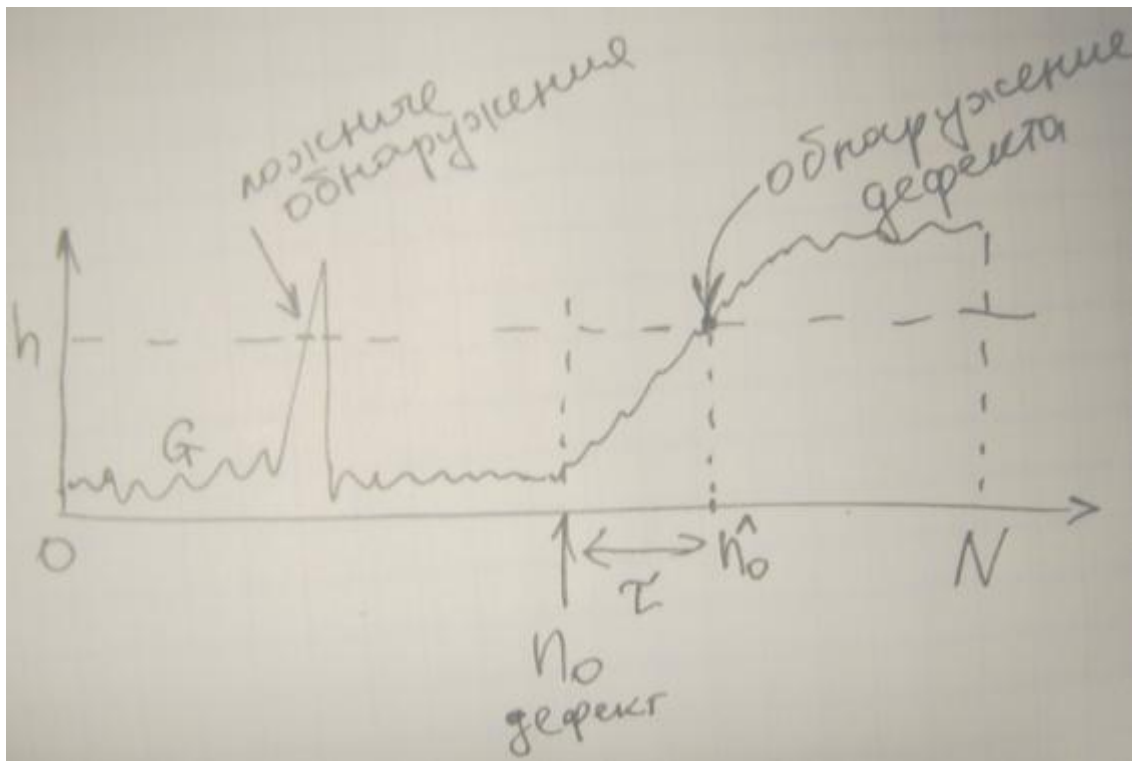
$$\begin{cases} z(n) \in N(m_1, b_1), n \in [0, n_1] \\ z(n) \in N(m_2, b_1), n \in [n_1 + 1, n_2] \\ z(n) \in N(m_1, b_1), n \in [n_2 + 1, n_3] \\ z(n) \in N(m_2, b_1), n \in [n_3 + 1, n_4] \\ \mathbf{L} \\ z(n) \in N(m_1, b_1), n \in [n_m + 1, N] \end{cases}$$



Дефект 4: восстанавливаемое скачкообразное изменение дисперсии.

$$\begin{aligned} z(n) &\in N(m_1, b_1), n \in [0, n_1] \\ z(n) &\in N(m_1, b_2), n \in [n_1 + 1, n_2] \\ z(n) &\in N(m_1, b_1), n \in [n_2 + 1, n_3] \\ z(n) &\in N(m_1, b_2), n \in [n_3 + 1, n_4] \\ \mathbf{L} \\ z(n) &\in N(m_1, b_1), n \in [n_m + 1, N] \end{aligned}$$

### Критерии качества обнаружения



Оценкой вероятности ложного обнаружения ( $P_{ло}$ ) является частота ложных обнаружений – отношение числа обнаружений к объему выборки при отсутствии дефекта:

$$P_{ло} = \frac{1}{N} \sum_{i=1}^N v_i,$$

где  $N$  – объем выборки,

$$v_i = \begin{cases} 1, G(i) \geq h \\ 0, G(i) < h \end{cases}$$

Среднее время обнаружения ( $\bar{t}$ ) – среднее смещение момента обнаружения относительно истинного момента дефекта:

$$\bar{t} = \frac{1}{K} \sum_{j=1}^K t_j,$$

где  $K$  – количество реализаций,

$$t_j = \hat{n}_{0j} - n_0 \text{ – время обнаружения одной реализации,}$$

$$\left| \begin{array}{l} n_0 - \text{время появления дефекта,} \\ \hat{n}_0 - \text{время обнаружения дефекта,} \\ \hat{n}_0 = \min_{n \geq n_0} \{n : G(n) \geq h\}. \end{array} \right.$$

## 1. Алгоритмы кумулятивных сумм

В алгоритмах данного класса на каждом шаге рассчитывается так называемая кумулятивная сумма.

### 1.1. Алгоритм Пейджа

Алгоритм Пейджа [15]:

$$G(n) = \begin{cases} S_n - \min_{1 \leq k < n} S_k, m_2 > m_1, S_n = \sum_{i=1}^n (z(i) - m_1 - v/2) \\ \max_{1 \leq k < n} S_k - S_n, m_2 < m_1, S_n = \sum_{i=1}^n (z(i) - m_1 + v/2) \end{cases}, \quad (1.1)$$

где  $m_2 = \bar{z}$ ,

$m_1$  – математическое ожидание выходного сигнала  $z$  до появления дефекта,

$n > 0$  – порог чувствительности.

Решение о наличии или отсутствии дефекта в каждый момент времени принимается на основе результатов сравнения:

$G < h \Rightarrow$  дефекта нет,

$G \geq h \Rightarrow$  есть дефект,

где  $h > 0$  – порог срабатывания.

*Модификация алгоритма (рекуррентный вид)*

Исходная формула (1.1) для случая  $m_2 > m_1$  преобразуется к виду:

$$G(n) = S_n - \hat{S}_n,$$

где  $\hat{S}_n = \min(\hat{S}_{n-1}, S_n)$ ,

$$S_n = S_{n-1} + z(n) - m_1 - v/2.$$

Начальные условия:  $S_1 = z(1) - m_1 - v/2$ ,  $\hat{S}_1 = S_1$ .

Вывод рекуррентной формулы:

$$\begin{aligned}
 G_n &= S_n - \min_{1 \leq k < n} S_k = S_n - \hat{S}_n \\
 S_n &= \sum_{i=1}^n (z_i - m_1 - v/2) = \\
 &= \sum_{i=1}^{n-1} (z_i - m_1 - v/2) + (z_n - m_1 - v/2) = \\
 &= S_{n-1} + (z_n - m_1 - v/2) \\
 \hat{S}_n &= \min_{1 \leq k < n} S_k = \\
 &= \min \left[ \min_{1 \leq k < n-1} S_k, S_n \right] = \\
 &= \min \left[ \hat{S}_{n-1}, S_n \right]
 \end{aligned}$$

Для случая  $m_2 < m_1$  преобразуется к виду:

$$\begin{aligned}
 G(n) &= \hat{S}_n - S_n, \\
 \text{где } \hat{S}_n &= \max(\hat{S}_{n-1}, S_n), \\
 S_n &= S_{n-1} + z(n) - m_1 + v/2.
 \end{aligned}$$

Начальные условия:  $S_1 = z(1) - m_1 + v/2$ ,  $\hat{S}_1 = S_1$ .

## 1.2. Алгоритм кумулятивных сумм с отражающим экраном

Алгоритм кумулятивных сумм с отражающим экраном (АКС) представляет собой модифицированную форму записи алгоритма Пейджа [15]:

$$\begin{aligned}
 G(n) &= \begin{cases} \max \{0, G(n-1) + z(n) - m_1 - v/2\}, & m_2 > m_1 \\ \max \{0, G(n-1) + z(n) - m_1 + v/2\}, & m_2 < m_1 \end{cases}, \\
 G(0) &= 0.
 \end{aligned}$$

*Аналитическая настройка параметров алгоритма*

При диагностировании изменений свойств процесса авторегрессии оценка вероятности ложного обнаружения определяется по формуле [16]:

$$P_{\text{ло}} = \frac{(m' - m)^2 \left( 1 - \sum_{i=1}^p a_i \right)^2}{2b(e^h - h - 1)},$$

оценка среднего времени обнаружения определяется по формуле:

$$\bar{t} = \frac{2b}{(m' - m)^2 \left( 1 - \sum_{i=1}^p a_i \right)^2} (e^{-h} + h - 1),$$

где  $a_i$  – коэффициенты авторегрессии,

$p$  – порядок авторегрессии,

$m$  – математическое ожидание процесса авторегрессии до разладки,

$m'$  – математическое ожидание процесса авторегрессии после разладки,

$b$  – дисперсия процесса авторегрессии,

$h$  – порог срабатывания алгоритма.

### *Модификация алгоритма кумулятивных сумм*

Если сигнал  $z(n)$  имеет составляющую типа редких больших выбросов, то алгоритмы Пейджа и АКС могут стать неприемлемыми, так как выброс резко увеличивает значение решающей функции и приводит к ложному обнаружению. Поэтому вводят две модификации алгоритма [8].

Первый вариант модифицированного алгоритма (АКС-м):

$$G(n) = \begin{cases} \max\{0, G(n-1) + \text{sign}(z(n) - m_1 - v/2)\}, & m_2 > m_1 \\ \max\{0, G(n-1) + \text{sign}(z(n) - m_1 + v/2)\}, & m_2 < m_1 \end{cases},$$

$$G(0) = 0.$$

Второй вариант модифицированного алгоритма (АКС-м2):

$$G(n) = \begin{cases} \max\{0, (G(n-1) + 1) \cdot \text{sign}(z(n) - m_1 - v/2)\}, & m_2 > m_1 \\ \max\{0, (G(n-1) + 1) \cdot \text{sign}(z(n) - m_1 + v/2)\}, & m_2 < m_1 \end{cases},$$

$$G(0) = 0.$$

### **1.3. Алгоритм Сегена-Сандерсона**

Алгоритм Сегена-Сандерсона [18]:

$$G(n) = \max_{i \leq n} \left\{ S_i - \min_{k \leq i} S_k \right\},$$

$$\text{где } S_i = \sum_{j=1}^i z(j).$$

Для получения возможности настройки алгоритма вводится модификация алгоритма:

$$G(n) = \frac{1}{\max(v, n)} \max_{i \leq n} \left\{ S_i - \min_{k \leq i} S_k \right\}, \quad (1.2)$$

где  $n > 0$  – порог чувствительности алгоритма.

*Модификация алгоритма (рекуррентный вид)*

Исходная формула (1.2) преобразуется к виду:

$$G(n) = \frac{\hat{G}_n}{\max(v, n)},$$

$$\text{где } \hat{G}_n = \max(\hat{G}_{n-1}, S_n - \hat{S}_n),$$

$$\hat{S}_n = \min(\hat{S}_{n-1}, S_n),$$

$$\left| \begin{array}{l} S_n = S_{n-1} + z(n). \\ \text{Начальные условия: } S_1 = z(1), \hat{S}_1 = S_1, \hat{G}_1 = 0. \end{array} \right.$$

#### 1.4. Алгоритм Надлера-Роббинза

Алгоритм Надлера-Роббинза [17]:

$$G(n) = \max_{0 \leq k \leq n} S_k - \min_{0 \leq k \leq n} S_k, \quad (1.3)$$

где  $S_0 = 0$ ,  $S_k = \sum_{j=1}^k (z(j) - m_1)$ .

Модификация алгоритма (рекуррентный вид)

Исходная формула (1.3) преобразуется к виду:

$$G(n) = S_n' - S_n'',$$

$$\text{где } S_n' = \max(S_{n-1}', S_n),$$

$$S_n'' = \min(S_{n-1}'', S_n),$$

$$S_n = S_{n-1} + z(n) - m_1.$$

$$\text{Начальные условия: } S_1 = z(1) - m_1, S_1' = S_1, S_1'' = S_1.$$

## 2. Алгоритмы со сглаживанием

### 2.1. Алгоритм сигнального отношения

Алгоритм сигнального отношения (ACO) [13]:

$$G(n) = \frac{S(n)}{R(n)}, \quad (2.1)$$

$$\text{где } S(n) = (1 - a)S(n-1) + ae(n), \quad (2.2)$$

$$R(n) = (1 - a)R(n-1) + a|e(n)|, \quad (2.3)$$

$$e(n) = z(n) - m_1. \quad (2.4)$$

Начальные значения:

$$S(0) = 0, R(0) = \sqrt{\frac{2}{p}}.$$

Параметры алгоритма:

$m_1$  – математическое ожидание выходного сигнала объекта до появления дефекта,

$0 < a < 1$  – коэффициент сглаживания.

Решение о наличии или отсутствии дефекта в каждый момент времени принимается на основе результатов сравнения:

$$h_1 < G < h_2 \Rightarrow \text{дефекта нет,}$$

$$G \geq h_2, G \leq h_1 \Rightarrow \text{есть дефект.}$$

где  $-1 < h_1 < h_2 < 1$  – нижний и верхний пороги срабатывания:

### *Аналитическая настройка параметров алгоритма*

Значения решающей функции, сформированной по данному алгоритму, распределены нормально с нулевым математическим ожиданием и дисперсией

$$q^2 = \frac{pa}{2\left(2 - \frac{2}{p}\right)^2(2-a)},$$

где  $a < 0.176$  – коэффициент сглаживания.

Оценка вероятности ложного обнаружения определяется по формуле [13]:

$$P_{lo} = 1 - \frac{1}{2} \operatorname{erf}\left(\frac{h_2}{q\sqrt{2}}\right) + \frac{1}{2} \operatorname{erf}\left(\frac{h_1}{q\sqrt{2}}\right),$$

$$\operatorname{erf}(x) = \frac{2}{\sqrt{p}} \int_0^x e^{-t^2} dt$$

где  $-1 < h_1 < h_2 < 1$  – нижний и верхний пороги срабатывания.

При диагностировании изменения математического ожидания случайного процесса от  $m_1 = 0$  до  $m_2 > 0$  и симметричных порогах  $h = h_2 = -h_1$  оценка среднего времени обнаружения определяется по формуле:

$$\bar{t} = \left\lceil \frac{\ln\left(\frac{r(1-h)}{r(1-h) + \frac{2h}{\sqrt{2p}}}\right)}{\ln(1-a)} \right\rceil - 1$$

где  $r = \frac{m_2}{b_1}$ ,

$\lceil x \rceil$  – округление сверху – ближайшее целое большее  $x$ ,

$b_1$  – дисперсия случайного процесса.

### **2.2. Алгоритм сигнального отношения с проверкой дисперсии**

В алгоритме сигнального отношения, описываемом формулами (2.1) – (2.4), можно ввести дополнительное сравнение  $R(n)$  с порогом  $h$ :

$R < h \Rightarrow$  дефекта нет,

$R \geq h \Rightarrow$  есть дефект.

Полученный алгоритм сигнального отношения с проверкой дисперсии (АСО-д) обладает возможностью обнаруживать изменение дисперсии.

### **2.3. Алгоритм экспоненциального сглаживания**

В качестве решающей функции можно использовать не отношение  $S(n)$  к  $R(n)$ , а только  $S(n)$ . В этом случае получим алгоритм экспоненциального сглаживания (АЭС) [6, 7]:

$$G(n) = (1-a)G(n-1) + ae(n),$$



$$G(0) = 0.$$

### 3. Алгоритмы с памятью

В алгоритмах данного класса при расчете решающей функции на каждом шаге используются значения сигнала в предыдущие  $M$  моментов времени. Параметр  $M > 0$  называется глубиной памяти алгоритма.

Решение о наличии или отсутствии дефекта в каждый момент времени принимается на основе результатов сравнения:

$$G < h \Rightarrow \text{дефекта нет,}$$

$$G \geq h \Rightarrow \text{есть дефект,}$$

где  $h > 0$  – порог срабатывания.

#### 3.1. Алгоритм Шьюхарта

Алгоритм Шьюхарта [7]:

$$G(n) = \begin{cases} 0, n < M \\ \frac{1}{M} \sum_{i=n-M+1}^n z(i), n \geq M \end{cases} \quad (3.1)$$

Модификация алгоритма (рекуррентный вид)

Исходная формула (3.1) преобразуется к виду:

$$G(n) = \begin{cases} 0, n < M \\ \frac{1}{M} \sum_{i=1}^M z(i), n = M \\ G(n-1) + \frac{z(n) - z(n-M)}{M}, n > M \end{cases}.$$

#### 3.2. Алгоритм Дарховского-Бродского

Алгоритм Дарховского-Бродского [5, 7, 12]:

$$G(n) = \begin{cases} 0, n < M \\ \max_{\lfloor aM \rfloor \leq k \leq \lfloor (1-a)M \rfloor} |Y_M(k, n)|, n \geq M, \end{cases}$$

$$\text{где } Y_M(k, n) = \frac{1}{k} \sum_{i=n-M+1}^{n-M+k} z(i) - \frac{1}{M-k} \sum_{i=n-M+k+1}^n z(i), n \geq M, \quad (3.2)$$

$\lfloor x \rfloor$  – округление снизу – ближайшее целое меньшее  $x$ ,

$0 < a < \frac{1}{2}$  – коэффициент глубины памяти.

Модификация алгоритма (рекуррентный вид)

Исходная формула (3.2) преобразуется к виду:

$$Y_M(k, n) = Y_M(k, n-1) + \frac{z(n-M+k) - z(n-M)}{k} - \frac{z(n) - z(n-M+k)}{M-k}, n > M$$

$$\text{Начальные условия: } Y_M(k, M) = \frac{1}{k} \sum_{i=1}^k z(i) - \frac{1}{M-k} \sum_{i=k+1}^M z(i).$$

### 3.3. Алгоритм Воробейчикова

Алгоритм Воробейчикова [9]:

$$G(n) = \begin{cases} h_0, n \leq M \\ \max\{h_0, G(n-1) + r * \text{sign}(z_n - z_{n-M}) - q\}, n > M \end{cases}$$

где  $r > q$  – натуральные несократимые числа,

$h_0 = r + q$  – минимальный порог срабатывания:  $h > h_0$ .

### 3.4. Алгоритм, основанный на принципе невязок

Решающая функция алгоритма, основанного на принципе невязок, формируется как невязка (расхождение) между наблюдаемой и принятой моделями случайного процесса  $z$  [2]:

$$G(n) = \frac{1}{\sqrt{2n}} \sum_{i=1}^n \left( \left( \mathcal{U}(i) \right)^2 - 1 \right).$$

Для получения возможности настройки алгоритма вводятся модификации алгоритма:

– вариант 1 – суммирование по окну с глубиной памяти  $M$

$$G(n) = \frac{1}{\sqrt{2n}} \sum_{i=n-M+1}^n \left( \left( \mathcal{U}(i) \right)^2 - 1 \right);$$

– вариант 2 – через каждые  $M$  отсчетов после каждого "обнаружения" дефекта обнулять  $G$ ,

где  $M > 0$  – глубина памяти,

$\mathcal{U}(n) \in N(0,1)$ :

1) если

$$z(n) \in N(m_1, b_1),$$

то

$$\mathcal{U}(n) = \frac{z(n) - m_1}{b_1};$$

2) если  $z$  – процесс авторегрессии

$$z(n) = m_1 + \sum_{i=1}^p a_i (z(n-i) - m_1) + b_1 y(n),$$

$$y(n) \in N(0,1),$$

то

$$\% (n) = \frac{z(n) - m_1 - \sum_{i=1}^p a_i (z(n-i) - m_1)}{b_1}.$$

*Аналитическая настройка параметров алгоритма*

Вероятность ложного обнаружения для этого алгоритма может быть определена на основе использования марковских процессов [14]. Такой подход позволяет получить следующее выражение для вероятности ложного обнаружения:

$$P_{ло} = \frac{1 - \prod_{i=1}^M p_i}{2 + \sum_{j=1}^{M-1} \prod_{i=1}^j p_i - \prod_{i=1}^M p_i},$$

$$\text{где } p_i = \begin{cases} F_i(h\sqrt{2i} + i) - F_i(-h\sqrt{2i} + i), & h \leq \sqrt{\frac{i}{2}}, \\ F_i(h\sqrt{2i} + i), & h > \sqrt{\frac{i}{2}} \end{cases},$$

$F_i$  – функция распределения  $\chi^2$  с  $i$  степенями свободы,

$M$  – глубина памяти,

$h$  – порог срабатывания алгоритма.

*Модификация алгоритма (рекуррентный вид)*

Исходная формула (2.39) преобразуется к виду:

$$G(n) = \sqrt{\frac{n-1}{n}} \cdot G(n-1) + \frac{1}{\sqrt{2n}} \left( (\% (n))^2 - 1 \right).$$

$$\text{Начальные условия: } G(1) = \frac{1}{\sqrt{2}} \left( (\% (1))^2 - 1 \right).$$

### 3.5. Алгоритм, основанный на проверке обобщенной дисперсии

Алгоритм, основанный на проверке обобщенной дисперсии (АОД) [10]:

$$G(n) = \begin{cases} 0, & n < M \\ \det(S_n), & n \geq M \end{cases},$$

$$\text{где } S_n = \frac{1}{M-1} \sum_{i=n-M+1}^n (z(i) - m_n)(z(i) - m_n)^T, n \geq M, \quad (3.3)$$

$$m_n = \frac{1}{M} \sum_{i=n-M+1}^n z(i), n \geq M. \quad (3.4)$$

Модификация алгоритма (рекуррентный вид)

Исходные формулы (3.3) – (3.4) преобразуются к виду (примем для простоты размерность вектора  $z$  равной 1):

$$S_n = S_{n-1} + \frac{z(n) - z(n-M)}{M-1} \left( z(n) + z(n-M) - 2m_{n-1} - \frac{1}{M} \right),$$

$$n > M$$

$$m_n = m_{n-1} + \frac{z(n) - z(n-M)}{M}, n > M,$$

Начальные условия:

$$S_M = \frac{1}{M-1} \sum_{i=1}^M (z(i) - m_M)^2,$$

$$m_M = \frac{1}{M} \sum_{i=1}^M z(i).$$

#### 4. Другие алгоритмы

##### 4.1. Алгоритм, основанный на интервальном подходе

Толерантный интервал – это интервал, который с заданной вероятностью  $g$  покрывает  $d$  случайных величин из всей выборки,  $0 < g < 1$ ,  $0 < d < 1$ . Доверительный интервал – это интервал, который с заданной вероятностью  $q$  покрывает величину,  $0 < q < 1$ .

Задавая доверительную вероятность  $q$ , строим доверительный интервал  $[u_1, u_2]$  для сигнала  $z(n) \in N(m_1, b_1)$  с помощью квантилей распределения Стьюдента  $t_{\frac{1+q}{2}}(n-1)$

[11]:

$$u_1(n) = m_n - t_{\frac{1+q}{2}}(n-1) \frac{S_n}{\sqrt{n}},$$

$$u_2(n) = m_n + t_{\frac{1+q}{2}}(n-1) \frac{S_n}{\sqrt{n}},$$

$$\text{где } m_n = \frac{1}{n} \sum_{i=1}^n z(i), \quad (4.1)$$

$$S_n^2 = \frac{1}{n-1} \sum_{i=1}^n (z(i) - m_n)^2, n > 1. \quad (4.2)$$

Для нахождения  $m_n$  и  $S_n^2$  можно использовать «принцип движущегося окна»:

$$m_n = \frac{1}{M} \sum_{i=n-M+1}^n z(i)$$

$$S_n^2 = \frac{1}{M-1} \sum_{i=n-M+1}^n (z(i) - m_n)^2$$

Задавая  $g$  и  $d$ , находим толерантный множитель  $K$ :

$$K_n = Z_\infty \left( 1 + \frac{Z_g}{\sqrt{2n}} + \frac{5Z_g^2 + 10}{12n} \right),$$

где  $Z_\infty, Z_g$  – абсциссы нормированной функции Лапласа  $\Phi_0(Z)$ :

$$d = 2\Phi_0(Z_\infty), \quad \Phi_0(Z_g) = g - 0.5,$$

$$\Phi_0(x) = \frac{1}{\sqrt{2p}} \int_0^x e^{-\frac{1}{2}z^2} dz = \frac{1}{2} \operatorname{erf} \left( \frac{x}{\sqrt{2}} \right).$$

Строим толерантный интервал  $[l_1, l_2]$ :

$$l_1(n) = m_n - K_n S_n,$$

$$l_2(n) = m_n + K_n S_n.$$

Решение принимается, исходя из анализа полученных интервалов:

$$u_2 \geq l_1, l_2 \geq u_1 \Rightarrow \text{дефекта нет,}$$

$$u_2 < l_1 \text{ или } l_2 < u_1 \Rightarrow \text{есть дефект.}$$

*Модификация алгоритма (рекуррентный вид)*

Исходные формулы (4.1) – (4.2) преобразуются к виду:

$$m_n = \frac{n-1}{n} m_{n-1} + \frac{1}{n} z(n), n > 1,$$

$$S_n^2 = \frac{n-2}{n-1} S_{n-1}^2 + \frac{1}{n} (z(n) - m_{n-1})^2, n > 1.$$

$$\text{Начальные условия: } m_2 = \frac{z(1) + z(2)}{2}, S_2^2 = \frac{(z(1) - z(2))^2}{2}.$$

#### 4.2. Допусковый алгоритм

Допусковый алгоритм состоит в проверке нахождения в допуске значений сигнала  $z$  [1]:

$$G(n) = z(n).$$

Решение о наличии или отсутствии дефекта в каждый момент времени принимается на основе результатов сравнения:

$$h_1 < G < h_2 \Rightarrow \text{дефекта нет,}$$

$$G \geq h_2, G \leq h_1 \Rightarrow \text{есть дефект,}$$

где  $h_1 < h_2$  – нижний и верхний пороги срабатывания:

#### *Аналитическая настройка параметров алгоритма*

Оценка вероятности ложного обнаружения определяется по формуле:

$$P_{\text{ло}} = 1 - \frac{1}{2} \operatorname{erf} \left( \frac{h_2 - m_1}{b_1 \sqrt{2}} \right) + \frac{1}{2} \operatorname{erf} \left( \frac{h_1 - m_1}{b_1 \sqrt{2}} \right), \quad (3.6)$$

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$$

где  $h_1 < h_2$  – нижний и верхний пороги срабатывания.

#### **4.3. Алгоритм Гиршика-Рубина-Ширяева**

Решающая функция данного алгоритма Гиршика-Рубина-Ширяева (ГРШ) рассчитывается по следующей формуле [3, 4]:

$$G(n) = e^{z(n)} (1 + G(n-1)),$$

$$G(0) = 0$$

Для получения возможности настройки алгоритма вводится модификация алгоритма:

$$G(n) = e^{z(n)-n} (1 + G(n-1)),$$

где  $n > 0$  – порог чувствительности алгоритма,

#### *Аналитическая настройка параметров алгоритма*

Оценка среднего времени обнаружения определяется по формуле [7]:

$$\bar{t} = \frac{\ln h}{m_2 - m_1 - |m_1|},$$

где  $m_1$  – математическое ожидание процесса до разладки,

$m_2$  – математическое ожидание процесса после разладки,

$h$  – порог срабатывания алгоритма.

### Свойства алгоритмов

Алгоритм	Обнаружение изменения среднего	Обнаружение изменения дисперсии	Восстанавли- ваемые дефекты	Трудоемкость	Память
Пейдж	+	—	— —	быстрый	мало
АКС	+	—	— —	быстрый	Мало
АКС-м	+	—	— —	быстрый	Мало
АКС-м2	+	—	— —	быстрый	Мало
СС	+	—	— —	быстрый	Мало
НР	+	—	— —	быстрый	Мало
АСО	+	—	++	Средний	Мало
АСО-д	+	+	++	Быстрый	Мало
АЭС	+	—	++	Быстрый	Мало
Шьюхарта	+	—	— —	Быстрый	много
Дарховского- Бродского	+	—	— —	Очень медленный	много
Воробейчикова	+	—	+ —	Быстрый	много
Невязок	+	+	++	Средний	много
АОД	+	+	— —	Средний	много
Интервальный	+	—	— —	Медленный	мало
Допусковый	+	+	++	Быстрый	Мало
ГРШ	+	+	± +	Быстрый	Мало

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Бендерская Е.Н., Колесников Д.Н., Пахомова В.И. Функциональная диагностика систем управления: Учебное пособие / СПбГТУ. – СПб.: Изд-во СПбГТУ, 2000. – 143 с.
2. Бородин Л.И., Моттль В.В. Алгоритм обнаружения моментов изменения параметров уравнения случайного процесса // Автоматика и телемеханика. – 1976. – №6. – с. 23-32.
3. Бродский Б.Е. Асимптотически оптимальные методы в задаче скорейшего обнаружения разладки. I. Характеристики методов скорейшего обнаружения разладки // Автоматика и телемеханика. – 1995. – №9. – с. 60-72.
4. Бродский Б.Е. Асимптотически оптимальные методы в задаче скорейшего обнаружения разладки. II. Исследование методов скорейшего обнаружения // Автоматика и телемеханика. – 1995. – №10. – с. 50-59.
5. Бродский Б.Е., Дарховский Б.С. О задаче скорейшего обнаружения момента изменения вероятностных характеристик случайной последовательности // Автоматика и телемеханика. – 1983. – №10. – с. 125-131.
6. Бродский Б.Е., Дарховский Б.С. Проблемы и методы вероятностной диагностики // Автоматика и телемеханика. – 1999. – №8. – с. 3-50.
7. Бродский Б.Е., Дарховский Б.С. Сравнительный анализ некоторых непараметрических методов скорейшего обнаружения момента “разладки” случайной последовательности // Теория вероятностей и ее применения. – 1990. – т.35, №4. – с. 655-668.
8. Бывайков М.Е., Ромащев А.А. О робастности в задаче обнаружения изменения параметра сдвига случайного процесса // Автоматика и телемеханика. – 1989. – №7. – с. 138-143.
9. Воробейчиков С.Э. Об обнаружении изменения среднего в последовательности случайных величин // Автоматика и телемеханика. – 1998. – №3. – с. 50-56.
10. Гаджиев Ч.М. Проверка обобщенной дисперсии обновляющей последовательности фильтра Калмана в задачах динамического диагностирования // Автоматика и телемеханика. – 1994. – №8. – с. 98-104.
11. Гаджиев Ч.М. Прогнозирование технического состояния динамических систем по обновляющей последовательности фильтра Калмана // Автоматика и телемеханика. – 1993. – №5. – с. 163-167.
12. Дарховский Б.С., Бродский Б.Е. Непараметрический метод скорейшего обнаружения изменения среднего случайной последовательности // Теория вероятностей и ее применения. – 1987. – т.32, №4. – с. 899-905.
13. Калишев О.Н. Метод диагностирования измерительных каналов с учетом предыстории // Автоматика и телемеханика. – 1988. – №6. – с. 135-143.
14. Кемени Дж., Снелл Дж. Конечные цепи Маркова. – М.: Наука, 1970. – 271 с.
15. Никифоров И.В. Последовательное обнаружение изменения свойств временных рядов / Академия наук СССР. Институт проблем управления. – М.: Наука, 1983. – 200 с.
16. Никифоров И.В. Применение кумулятивных сумм для обнаружения изменения характеристик случайного процесса // Автоматика и телемеханика. – 1979. – №2. – с. 48-58.
17. Nadler J., Robbins N.B. Some characteristics of page’s two-sided procedure for detecting a change in a location parameter. // The Annals of Mathematical Statistics. – 1971. – vol. 42, №2. – p. 231-238.
18. Segen J., Sanderson A.C. Detecting Change In a Time-Series // IEEE Transactions on Information Theory. – 1980. – vol. IT-26, №2. – p. 249-255.