Оглавление

[1. Исходные данные 3](#_Toc27703226)

[2. Этапы моделирования каждого алгоритма с каждым типом дефекта 3](#_Toc27703227)

[3. Программа работы 4](#_Toc27703228)

[4. Выполнение работы 5](#_Toc27703229)

[4.1. Алгоритм, основанный на принципе невязок 5](#_Toc27703230)

[5. Выводы 7](#_Toc27703231)

[6. Листинг 8](#_Toc27703232)

# Исходные данные

Вариант №4.

|  |  |  |  |
| --- | --- | --- | --- |
| Система | Алгоритм 1 | Алгоритм 2 | Уровни вероятности ложного обнаружения |
| А | невязок | Экспоненциального сглаживания | 0.01 |

Используется построенная система диагностирования, для обнаружения изменения параметров объекта диагностирования используются заданные алгоритмы (лабораторные работы №3-4). Для каждого алгоритма имеются наборы параметров, обеспечивающие заданные уровни вероятности ложного обнаружения (курсовая работа).

# 2. Этапы моделирования каждого алгоритма с каждым типом дефекта

1. Составить план экспериментов, выбрать объем выборки и количество экспериментов, исходя из требуемого уровня доверительной вероятности для показателей качества обнаружения. Определить значения времени обнаружения, при которых работу алгоритма можно считать неудовлетворительной/удовлетворительной/хорошей/отличной.

2. Для базовых уровней дефектов (для каждого типа дефекта) определить среднее время обнаружения, рассмотреть все наборы параметров алгоритма. Построить графики зависимостей среднего времени обнаружения от параметров алгоритма. Среди наборов параметров, обеспечивающих заданный уровень вероятности ложного обнаружения, выбрать набор, обеспечивающий наименьшее среднее время обнаружения.

3. Для выбранного набора параметров алгоритма определить среднее время обнаружения, рассмотреть малый, средний и большой уровни дефектов (для каждого типа дефекта). Построить графики зависимостей среднего времени обнаружения от величин дефекта, проиллюстрировать при этом доверительные интервалы времени обнаружения. Оценить качество обнаружения в соответствии с введенными характеристиками (неудовлетворительное / удовлетворительное / хорошее / отличное).

# 3. Программа работы

1. Для каждого типа дефекта провести моделирование алгоритма 1, настроенного на уровень ложного обнаружения 0,01.

2. Для каждого типа дефекта провести моделирование алгоритма 1, настроенного на уровень ложного обнаружения 0,02.

3. Для каждого типа дефекта провести моделирование алгоритма 2, настроенного на уровень ложного обнаружения 0,01.

4. Для каждого типа дефекта провести моделирование алгоритма 2, настроенного на уровень ложного обнаружения 0,02.

Каждый пункт 1–4 предполагает выполнение моделирование по следующей схеме:

1) наборы параметров из курсовой работы –> имитационное моделирование с дефектом базового уровня –> средние времена обнаружения –> набор параметров, обеспечивающий наименьшее среднее время обнаружения

2) набор параметров, обеспечивающий наименьшее среднее время обнаружения –> имитационное моделирование с разными уровнями дефекта –> средние времена обнаружения

5. Предложить варианты комплексирования частных решений, принимаемых с помощью рассмотренных алгоритмов, для построения системы диагностирования в целом для всего ОД. Отметить области применения систем, на основе рассмотренных методов и возможные подходы по улучшению качества работы и расширению функциональных возможностей за счет возможных модификаций (алгоритмов, структуры, параметров функционирования и т.п.).

6. Сформулировать выводы о проделанной работе, оформить отчет.

# 4. Выполнение работы

## 4.1. Алгоритм, основанный на принципе невязок

Код программы представлен в приложении 1.

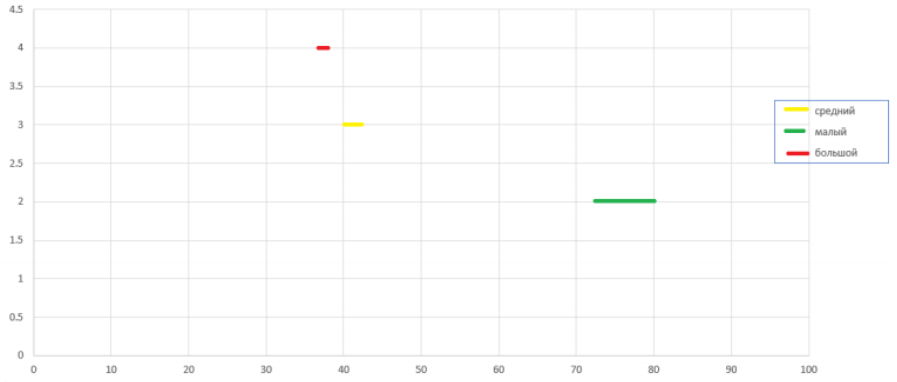
Таблица 1. Базовый дефект (М=2)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Pло | h | M |  |  |  |
| 0,0095 | 2,3 | 20 | 73,2606 | 76,7610 | 80,2614 |
| 0,0092 | 2,4 | 30 | 90,2476 | 93,9220 | 97,5964 |
| 0,0195 | 2,2 | 30 | 79,4500 | 82,7340 | 86,0180 |
| 0,0196 | 2,1 | 10 | 152,4370 | 160,7120 | 168,9870 |

Исходя из полученных результатов выбраны параметры h=2.3 M=20для Plo=0.01 и h=2.2 M=30 для Plo=0.02

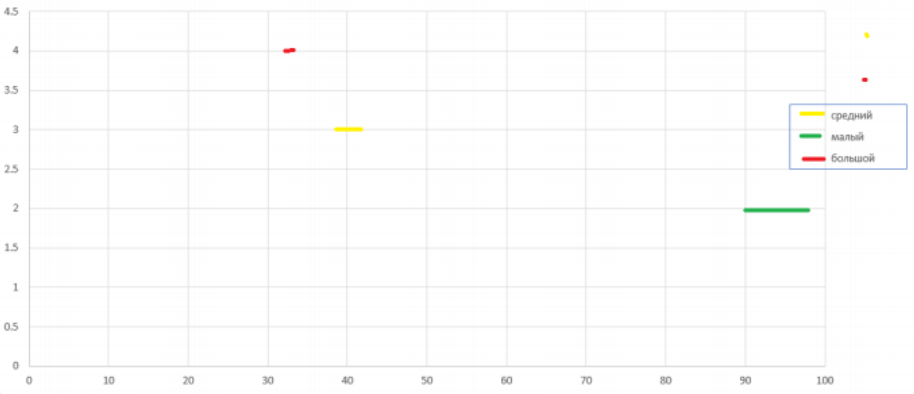
Для Plo=0.01:

|  |  |  |  |
| --- | --- | --- | --- |
| Уровень дефекта |  | Доверительный интервал | |
|  |  |
| Малый | 76,7610 | 73,2606 | 80,2614 |
| Средний | 41,8990 | 41,5226 | 42,2754 |
| Большой | 36,8360 | 36,0902 | 37,5818 |



Для Plo=0.02:

|  |  |  |  |
| --- | --- | --- | --- |
| Уровень дефекта |  | Доверительный интервал | |
|  |  |
| Малый | 93,9220 | 90,2476 | 97,5964 |
| Средний | 41,1910 | 39,9445 | 42,4375 |
| Большой | 33,6070 | 33,6940 | 33,9200 |



Тогда данный алгоритм хорошо обнаруживает большие и средниедефекты и удовлетворительно – малые.

# 5. Выводы

В ходе работы было выполнено моделирование следующего алгоритма, основанного на принципе невязок. С его помощью определялся дефект типа смещение уровня сигнала. Алгоритмневязок показал хорошие результаты по времени обнаружения дефекта

# 6. Листинг

Приложение 1

clear all;

%% Исходные данные

% Количество снятых точек

N = 1000;

% Момент возникновения неисправности

N\_def = N/2;

a1 = 0.1;

a2 = 0.5;

A = [a1 a2; 1 0];

F = [1;0];

C = [1 0];

Mi = 0;

Mi\_err=2;

Di = 1;

Mo = 0;

Do = 4;

diAuto = (1+a2)\*((1-a2)^2-a1^2)/(1-a2);

% Глубина памяти

memDepth = 20;

% Порог срабатывания

h = 2.3;

t\_obn=zeros(1,1000);

for i=1:1:1000

x = [0; 0];

P = Do\*eye(2);

G = zeros(1,N);

X1\_est = [0; 0];

zNew = zeros(1, N);

zNormal = zNew;

t\_cnt=0;

%% Цикл работы фильтра

for j = 1:N

if (j > N\_def)

x = A\*x+F\*diAuto\*normrnd(Mi\_err,Di^0.5,1,1);

Y = C\*x+normrnd(Mo,Do^0.5);

else

x = A\*x+F\*diAuto\*normrnd(Mi,Di^0.5,1,1);

Y = C\*x+normrnd(Mo, Do^0.5);

end

Q = A\*P\*A'+F\*Di\*F';

K = Q\*C'\*(C\*Q\*C'+Do)^-1;

P = Q-K\*C\*Q;

X1\_est = A\*X1\_est+F\*Mi+K\*(Y-C\*(A\*X1\_est+F\*Mi));

zNew(j) = Y-C\*(C\*X1\_est+F\*Mi);

S = Do+C\*P\*C'-Do\*K'\*C'-C\*K\*Do;

zNormal(1,j)= zNew(j)\*(S^-0.5);

%% Алгоритм обнаружения дефекта методом невязок

if(j <= memDepth)

if(j ~= 1)

G(j) = 1/sqrt(2\*j)\*sum(((zNormal(1,1:j)-Mi -

a1\*(zNormal(1,j-1)-Mi))./Di).^2-1);

else

G(j) = 1/sqrt(2\*j)\*sum(((zNormal(1,1:j)-Mi)./Di).^2-1);

end

else

G(j) = 1/sqrt(2\*memDepth)\*sum(((zNormal(1,jmemDepth:j)-Mia1\*(zNormal(1,j-1)-Mi))./Di).^2-1);

end

if(abs(G(j)) >= h)

t\_cnt=t\_cnt+1;

if(t\_cnt>30)

t\_obn(i)=j-500;

break;

end

else

t\_cnt=0;

end

end

end

q=0.95;

mean(t\_obn)

mean(t\_obn) - tinv( (1+q)/2, 1000-1 ) \* std(t\_obn) / sqrt(1000)

mean(t\_obn) + tinv( (1+q)/2, 1000-1 ) \* std(t\_obn) / sqrt(1000)