

УДК 53.088.3; 53.088.7

МОДЕЛЬ АДАПТИВНОГО ФИЛЬТРА КАЛМАНА

Д.т.н. Ю.К. Тараненко, к.т.н. О.Ю. Олейник, Украинский химико-технологический университет, г. Днепр.

В работе представлена модель адаптивного фильтра Калмана, реализованная средствами Python. Результаты имитационного моделирования демонстрируют применение разработанной модели адаптивного фильтра для компенсации эффектов амплитудных и фазовых искажений в канале передачи данных.

У роботі представлена модель адаптивного фільтра Калмана, реалізована засобами Python. Результати імітаційного моделювання демонструють застосування розробленої моделі адаптивного фільтра для компенсації ефектів амплітудних і фазових спотворень в каналі передачі даних.

The article presents the Kalman adaptive filter model implemented by Python tools. The results of simulation simulations demonstrate the application of the developed adaptive filter model to compensate for the effects of amplitude and phase distortion in the data transmission channel.

Ключевые слова: фильтр Калмана, точность, шум, погрешность измерения.

Постановка проблемы. Основная задача, с которой сталкиваются проектировщики измерительной техники, это повышение точности измерительной схемы. Сегодня известно два способа решения данной задачи: первый – повышение чувствительности датчика к измеряемой величине, что сопровождается и повышением чувствительности к возмущающим воздействиям, и это, в свою очередь, требует применения дополнительных схем компенсации этого воздействия, снижающих точность прибора [1].

Основных способов повышения точности два, первый состоит в повышении чувствительности к измеряемой величине, однако при этом как правило растёт чувствительность и к неинформативным параметрам, что требует принятия дополнительных мер по их компенсации.

Второй способ решения лежит в плоскости методов статистической обработки результатов измерений [2], суть которых сводится к тому, что среднеквадратичное отклонение результата измерения обратно пропорционально корню квадратному из числа измерений. Статистические методы повышения точности разнообразны и многочисленны, но и они делятся на пассивные (для статических измерений) и активные (для динамических измерений). При этом сама измеряемая величина так же, как и помеха являются случайными величинами с изменяющимися дисперсиями

Адаптивность методов повышения точности динамических измерений заключается в использовании прогнозирования значений дисперсий и погрешности для каждого следующего цикла измерений. Поставленная задача решается применением цифровых фильтров,

которые представляют собой алгоритмы обработки данных с целью устранения шумов, искажения или избытка информации [3].

Анализ последних публикаций. В отличие от аналоговых фильтров, характеристики цифровых фильтров могут быть легко изменены путем изменения коэффициентов. Это делает цифровые фильтры привлекательными в коммуникационных приложениях, таких как адаптивный эквалайзинг, компенсация эхо, подавление шума, анализ и синтез речи и т.д. [4].

Основная идея адаптивной фильтрации представлена на рис.1 [5].

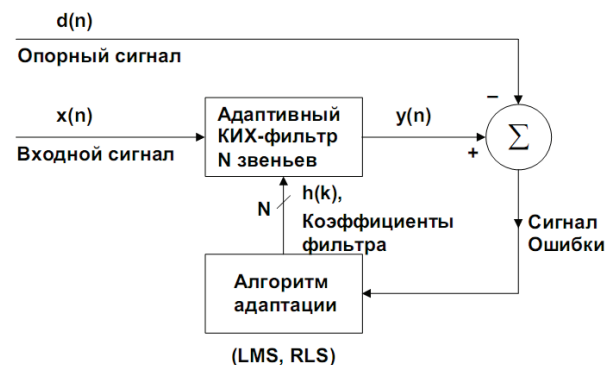


Рис.1. Схема адаптивной фильтрации

Основная цель адаптивной фильтрации состоит в том, чтобы отфильтровать входной сигнал $x(n)$ так, чтобы он соответствовал опорному сигналу $d(n)$. Для генерации сигнала ошибки опорный сигнал $d(n)$ вычитается из фильтруемого сигнала $y(n)$. Сигнал ошибки управляет алгоритмом адаптации, который генерирует коэффициенты фильтра, минимизирующие сигнал ошибки. Наиболее популярными алгоритмами являются метод наименьших квадратов и рекурсивный метод наименьших квадратов.

Сегодня известно много алгоритмов реализации цифровых фильтров. Винеровские фильтры, работающие в частотной области, лучше всего подходят для обработки процессов или отрезков процессов в целом (блочная обработка) [6]. Для последовательной обработки требуется текущая оценка сигнала на каждом такте с учетом информации, поступающей на вход фильтра в процессе наблюдения.

При винеровской фильтрации каждый новый отсчет сигнала потребовал бы пересчета всех весовых коэффициентов фильтра. В настоящее время широкое распространение получили адаптивные фильтры, в которых поступающая новая информация используется для непрерывной корректировки ранее сделанной оценки сигнала (сопровождение цели в радиолокации, системы автоматического регулирования в управлении и т.д.).

Особенный интерес представляют адаптивные фильтры рекурсивного типа, известные как фильтр Калмана.

Эти фильтры широко используются в контурах управления в системах автоматического регулирования и управления. В отличие от фильтра Винера, фильтр Калмана работает во временной, а не в частотной, области. Существенным преимуществом фильтра Калмана есть возможность задать априорную информацию о характере системе, связи переменных и на основании этого строить более точную оценку, но даже в простейшем случае (без ввода априорной информации) он дает отличные результаты.

Фильтр Калмана был разработан для решения многомерных задач, формулировка которых осуществляется в матричной форме. Матричная форма достаточно подробно описана для реализации на Python в работах [7], [8]. Описание работы фильтра Калмана, приведенная в указанных статьях, рассчитана на специалистов в области цифровой фильтрации. Поэтому возникла необходимость рассмотреть реализацию фильтра Калмана в более простой скалярной форме, что позволит упростить и расширить его применение в приборостроении.

Цель работы. Демонстрация результатов имитационного моделирования с применением разработанной модели адаптивного фильтра Калмана для компенсации эффектов амплитудных и фазовых искажений в канале передачи данных.

Результаты исследования. Рассмотрим схему фильтра Калмана для его дискретной формы.

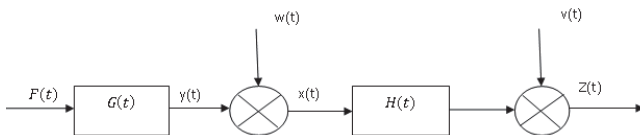


Рис.2. Схема дискретной формы фильтра Калмана.

$G(t)$ блок, работа которого описывается линейными соотношениями. На выходе блока вырабатывается неслучайный сигнал $y(t)$. Этот сигнал суммируется с шумом $w(t)$, который возникает внутри контролируемого объекта. В результате такого сложения, получаем новый сигнал $x(t)$. Этот сигнал представляет сумму неслучайного сигнала и шума и является случайным сигналом. Далее сигнал $x(t)$ преобразуется линейным блоком $H(t)$, суммируясь с шумом $v(t)$, распределённым по отличному от $w(t)$ закону. На выходе линейного блока $H(t)$ получаем случайный сигнал $z(t)$, по которому и определяется неслучайный сигнал $y(t)$. Следует отметить, что линейные функции блоков $G(t)$ и $H(t)$ могут тоже зависеть от времени.

Будем считать, что случайные шумы $w(t)$ и $v(t)$ – это случайные процессы с дисперсиями Q , R и нулевыми математическими ожиданиями. Сигнал $x(t)$ после линейного преобразования в блоке $G(t)$ распределён во времени по нормальному закону. С учётом изложенного, соотношение для измеряемого сигнала примет вид:

$$z(t) = H(x(t)) + v(t) \quad (1)$$

После фильтра нужно получить максимально возможное приближение y к неслучайному сигналу $y(t)$. При непрерывном динамическом измерении каждое следующее состояние объекта, а, следовательно, и значение контролируемой величины отличается от предыдущего по экспоненциальному закону с постоянным времени T в текущем интервале времени $t_{i+1} - t_i$,

$$x_{i+1} = x_i \cdot e^{-\frac{\Delta t}{T}} \quad (2)$$

За основу разработки был взят известный алгоритм работы фильтра Калмана, реализованный для Mathcad [9], результат работы которого демонстрирует рис.3. Заметно, что в результате преждевременной смены состояния для сравниваемой переменной $x(t)$ возрастает погрешность на участке резких изменений:

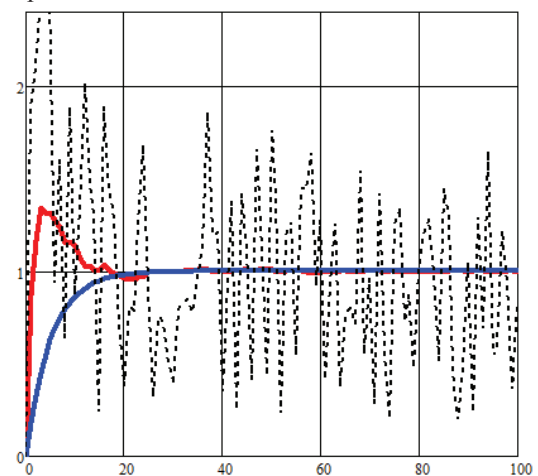


Рис. 3. Результат работы фильтра Калмана по алгоритму [3]:

- модельная функция незашумленного сигнала;
- модельная функция фильтра Калмана;
- измеряемая функция с шумом.

Для реализации модели цифрового фильтра Калмана для компенсации искажений в канале передачи данных была использована программная среда Python 3.4, которая является свободно доступным программным продуктом.

Процесс измерения рассматривается для суммы двух псевдослучайных величин, каждая из которых образуется как функция нормального распределения от равномерного распределения.

Листинг созданной модели простой скалярной реализации фильтра Калмана средствами программирования общего назначения Python имеет вид:

```
#!/usr/bin/env python
#coding=utf8
import matplotlib.pyplot as plt
import numpy as np
from numpy import exp,sqrt
from scipy.stats import norm
Q=0.8;R=0.2;y=0;x=0#начальные дисперсии шумов(выбраны произвольно) и нулевые значения переменных.
```

```

P=Q*R/(Q+R)# первая оценка дисперсий шумов.
T=5.0#постоянная времени.
n=[];X=[];Y=[];Z=[]#списки для переменных.
for i in np.arange(0,100,0.2):
    n.append(i)#переменная времени.
    x=1-exp(-1/T)+x*exp(-1/T)#модельная
функция для x.
    y=1-exp(-1/T)+y*exp(-1/T)# модельная
функция для y.
    Y.append(y)#накопление списка значений y.
    X.append(x)# накопление списка значений x.
    norm1 = norm(y, sqrt(Q))# нормальное
распределение с #математическим ожиданием – y.
    norm2 = norm(0, sqrt(R))# нормальное
распределение с #математическим ожиданием – 0.

ravn1=np.random.uniform(0,2*sqrt(Q))#равномерное
распределение #для шума с дисперсией Q.
    ravn2=np.random.uniform(0,2*sqrt(R))#
равномерное распределение #для шума с дисперсией R.
    z=norm1.pdf(          ravn1)+norm2.pdf(
ravn2)#измеряемая переменная z.
    Z.append(z)# накопление списка значений z.
    P=P-(P**2)/(P+Q+R) #переход в новое
состояние для x.
    x=(P*z+x*R)/(P+R)# новое состояние x.
    P=(P*R)/(P+R)# прогноз для нового
состояния x.

plt.plot(n, Y, color='g',linewidth=4, label='Y')
plt.plot(n, X, color='r',linewidth=4, label='X')
plt.plot(n, Z, color='b', linewidth=1, label='Z')
plt.legend(loc='best')
plt.grid(True)
plt.show()

```

Разработанный алгоритм обработки сигнала отличается использованием начальной прогнозной оценки влияния шумов. Это дало возможность уменьшить ошибку измерений $v(t)$. Алгоритм использует заданные модельные экспоненциальные функции, поэтому для наглядности приведём их отдельно на общем графике работы фильтра Калмана (рис.4).

Выводы

Сравнительный анализ результатов фильтрации с использованием разработанной модели подтверждает перспективность данной разработки и создаёт плодотворную базу для дальнейших исследований в области обработки сильно зашумленных результатов измерения. Одним из наиболее перспективных направлений развития Калмановских фильтров может быть работа над созданием модифицированного фильтра, погрешности которого будут представлены цветным шумом, что сделает его еще более ценным для решения реальных задач обработки измерительной информации.



Рис. 4. Результаты работы разработанной модели фильтра Калмана

СПИСОК ЛИТЕРАТУРЫ:

1. Основы теории точности измерительных приборов. [Текст] / С.В. Слепова // Челябинск. ЮУрГУ, 2008 г., 192 с.
2. Тараненко Ю.К. Применение байесовских методов при обработке сильно зашумленных результатов измерений [Текст] / Ю.К. Тараненко, О.Ю. Олейник // Вимірювальна та обчислювальна техніка в технологічних процесах. – 2017. – Т.№1. – С. 205-210.
3. Скляр Б. Цифровая связь. Теоретические основы и практическое применение: Пер. с англ. - М.: Издательский дом "Вильямс", 2003. - 1104 с
4. Семушин И.В., Цыганова Ю.В., Куликова М.В., Фатянова О.А. Адаптивные системы фильтрации, управления и обнаружения: коллективная монография под. ред. проф. И.В. Семушина. – Ульяновск: Изд-во УлГУ, 2011. – 298 с.
5. Уолт Кестер. Аналого-цифровое преобразование / М.: Техносфера, 2007. – 1016 с.
6. Шахтарин Б.И. Фильтры Винера и Калмана. Учебное пособие для вузов. 2014. – 396 с.
7. Дегтярёв, Ш. Тайль. Элементы теории адаптивного расширенного фильтра Калмана / Препринт ИПМ им. М.В. Келдыша РАН. – М., 2003. – №26. – 35с.
8. Гаврилов А.В. Использование фильтра Калмана для решения задач уточнения координат БПЛА // Современные проблемы науки и образования. – 2015. – № 1-1.; URL: <https://www.scienceeducation.ru/ru/article/view?id=19453> (дата обращения: 01.06.2017).
9. Яньков В.Ю. Лабораторный практикум по Маткаду. [Текст] / В.Ю. Яньков // М., МГУТУ, 2009. – 68с.