

МЕТОДЫ ЦИФРОВОЙ ФИЛЬТРАЦИИ В ОБРАБОТКЕ СИГНАЛОВ ЗОНДОВОЙ ЭЛЕКТРОМЕТРИИ

Тявловский А.К.¹, Поводайко А.Д.¹, Жарин А.Л.¹, Воробей Р.И.¹, Гусев О.К.¹, Колтунович Т.², Пантелеев К.В.¹

¹Белорусский национальный технический университет
Минск, Республика Беларусь

²Люблинский технический университет
Люблин, Польша

Измерения контактной разности потенциалов (КРП) и работы выхода электрона (РВЭ) методами зондовой электрометрии характеризуются относительно низким отношением сигнал / шум, что остро ставит проблему фильтрации измерительного сигнала. Улучшить отношение сигнал / шум можно за счет использования в качестве измерительного преобразователя цифрового зонда Кельвина [1], работающего в режиме большого сигнала, однако такой режим практически исключает использование классических аналоговых фильтров, поскольку выходной сигнал измерительного преобразователя представляется в форме динамического числового массива, т.е. дискретной последовательности цифровых отсчетов, поступающих с постоянными интервалами в реальном масштабе времени.

Такой динамический массив может быть обработан в режиме реального времени средствами цифровой обработки микропроцессора для дополнительного подавления шумов и паразитных гармоник сигнала, а также для статистического вычисления амплитуды сигнала, что позволяет существенно повысить точность определения параметров компенсационной зависимости цифрового зонда Кельвина и, соответственно, точность измерения КРП или РВЭ

Поток информации с выхода измерительного преобразователя можно представить суммой детерминированной (или квазидетерминированной) $Y_d(t)$ и стохастической (случайной) $Y_c(t)$ составляющих:

$$Y(t) = Y_d(t) + Y_c(t) \quad (1)$$

Цифровая фильтрация направлена на устранение стохастической составляющей, что достигается путем выполнения математических операций с оцифрованными данными. Одним из наиболее простых примеров цифрового фильтра с конечной импульсной характеристикой (КИХ-фильтра) является фильтр скользящего среднего, в котором на отфильтрованное значение влияет только h последних отсчетов:

$$y_k = \frac{1}{h} \sum_{i=l}^k x_i \quad (2)$$

где x_i – исходные значения рассматриваемой функции; $h=k-l$ – сглаживающий интервал – ко-

личество значений исходной функции для расчета скользящего среднего.

Схематически КИХ-фильтр может быть представлен графом потока сигналов, вид которого показан на рисунке 1.

На рисунке 2 приведен результат применения данного фильтра к реальному измерительному сигналу. Чем шире сглаживающий интервал, тем более плавным будет график результирующей функции. С другой стороны, увеличение сглаживающего интервала приводит к временному сдвигу усредненной функции относительно исходной.

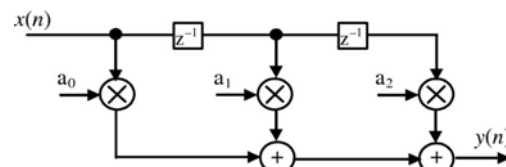


Рисунок 1 – Граф потока сигналов КИХ-фильтра

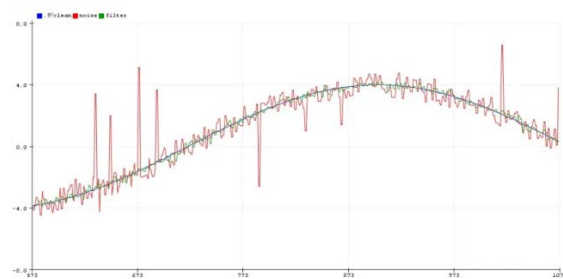


Рисунок 2 – Результат применения фильтра скользящего среднего к сигналу, содержащему шум и случайные выбросы

Основным недостатком КИХ-фильтров является большое количество отводов, необходимых для реализации некоторых аспектов частотной характеристики, а именно резкого обрезания [2]. Фильтры с бесконечной импульсной характеристикой (БИХ-фильтры) помимо входных используют также предыдущие выходные значения функции:

$$y(n) = \sum_{i=0}^{N-1} a_i x(n-i) + \sum_{j=1}^{M-1} b_j y(n-j) \quad (3)$$

Примером БИХ-фильтра первого порядка является экспоненциальный фильтр, описываемый выражением

$$y_k = y_{k-1} + \theta(x_k - y_{k-1}), \quad (4)$$

где x_k – измеренное значение в момент времени k ; y_k – фильтрованное значение в момент времени k ; θ – коэффициент фильтрации (от 0 до 1).

Результат применения экспоненциального фильтра представлен на рисунке 3. Данный фильтр схож с фильтром скользящего среднего, но имеет несколько преимуществ: 1) простота вычисления; 2) не требуется хранить в памяти микроконтроллера несколько предыдущих значений; 3) хорошо подавляет резкие скачки в измерениях; 4) довольно быстро приближается к значениям после скачка. С другой стороны, этот фильтр имеет довольно большую задержку.

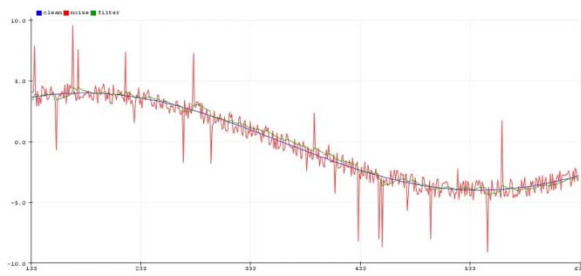


Рисунок 3 – Результат применения экспоненциального фильтра

Для фильтрации импульсных помех (выбросов) наиболее эффективен медианный фильтр. Данный фильтр использует массив из N последних значений, которые отсортированы по возрастанию. Значение, расположенное в середине отсортированного списка, присваивается выходному сигналу. Последний в этом случае также формируется с задержкой.

Практика проектирования показала, что наилучшего результата при обработке сигнала зондового электрометрического преобразователя, характеризующегося относительно малой шириной спектра полезного сигнала и низким отношением сигнал / шум, можно достигнуть при сочетании нескольких фильтров. Программная реализация описанных фильтров создавалась в среде разработки *Arduino IDE*. Для простого фильтра скользящего среднего использовалась временная переменная *sum*, предназначенная для накопления суммы отсчетов, с последующим ее делением на количество накопленных отсчетов. Размер выборки задается функцией *NUM_READ*. Увеличение размера выборки улучшает качество фильтрации. Практика показала, что такой фильтр целесообразно использовать в ситуации, при которой источник первичного сигнала опрашивается редко, время его опроса невелико и много меньше времени изменения сигнала, а результирующее время выборки не влияет на работу основной программы.

Более универсальным оказалось решение, основанное на формировании глобального или статического *static float valArray[NUM_READ]* массива. При поступлении нового отсчета сигнала весь массив сдвигается влево и в самую последнюю ячейку записывается новое значение *valArray[NUM_READ - 1] = newVal*. После этого массив суммируется, и сумма делится на количество ячеек. Данный способ одновременно хорош и плох тем, что создает некоторое запаздывание среднего значения, причем чем больше окно усреднения, т.е. размер выборки, тем эффективнее будет усредняться сигнал, но тем выше задержка относительно реального времени.

Данный недостаток может быть устранен введением кольцевого буфера *static int vals[NUM_READ]*, который может использоваться без сдвига. Для внесения нового отсчета в массив используется переменная, которая хранит текущее положение отсчета в массиве. Вместо полного суммирования массива используется переменная суммы, к которой при каждом вызове фильтра прибавляется новое значение *average += newVal* и вычитается самое старое *average -= vals[t]*. Благодаря этому количество операций сложения не зависит от размера буфера и всегда составляет 2. Этим обеспечивается высокое быстродействие при большой величине выборки.

Для фильтра экспоненциального среднего была создана оригинальная программная реализация, исключающая необходимость использования буфера. Выходное значение вычисляется на основе нового отсчета и предыдущего выходного значения: $filVal = newVal \cdot k + filVal(1 - k)$, где $k = 0.0 \dots 1.0$.

Коэффициент k определяет степень сглаживания сигнала. Для удобства программной реализации данное выражение было преобразовано путем раскрытия скобок и группирования слагаемых по k : $filVal = newVal \cdot k + filVal - filVal \cdot k$, $filVal = filVal + (newVal - filVal) \cdot k$, $filVal += (newVal - filVal) \cdot k$.

Использование составного оператора сложения позволяет уменьшить количество выполняемых математических операций и, соответственно, повысить быстродействие фильтра экспоненциального среднего.

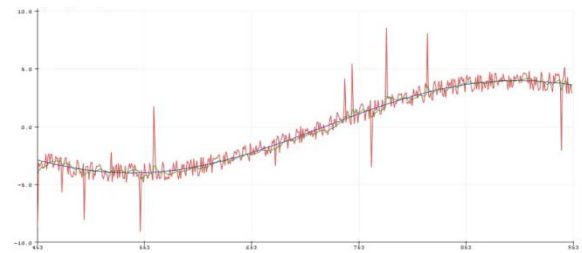


Рисунок 4 – Результат совместного использования фильтра скользящего среднего, экспоненциального и медианного фильтров

Медианный фильтр был реализован как фильтр третьего порядка. Подлежащие фильтрации данные помещались в трехэлементный буфер $float a = buf[0]; float b = buf[1]; float c = buf[2]$. Предложенная программная реализация медианного фильтра не требует вычислений, а только операций сравнения, благодаря чему его наличие не вносит дополнительных задержек при совместном использовании с фильтрами скользящего и экспоненциального среднего.

Пример результата, получаемого при совместном использовании описанных цифровых фильтров для сигнала, подверженного воздей-

ствию шумов и импульсных помех, показан на рисунке 4. Такое использование обеспечивает равно эффективное подавление как выбросов, так и шумовой составляющей сигнала.

Литература

1. Пантелеев, К.В. Цифровой измеритель контактной разности потенциалов / К.В. Пантелеев, А.И. Свистун, А.К. Тявловский, А.Л. Жарин // Приборы и методы измерений. 2016. – Т. 7, № 2. – С. 136–144.
2. Parker, M. Digital Signal Processing. 2nd Edition / M. Parker. – Elsevier Inc., 2017. – 432 p.

УДК 621.317.08

ТЕОРИЯ ДВУМЕРНОЙ ОБРАБОТКИ ФИНИТНЫХ СИГНАЛОВ В ПРОСТРАНСТВЕННО-ЧАСТОТНОЙ ОБЛАСТИ В БАЗИСАХ ФУРЬЕ С ВАРЬИРУЕМЫМИ ПАРАМЕТРАМИ Пономарев А.В.

*Ижевский государственный технический университет имени М.Т. Калашиникова
Ижевск, Российская Федерация*

Теория, методы и алгоритмы компьютерной Фурье-обработки двумерных (2-D) сигналов являются основой решения многих проблем в приборостроении: в области систем управления, обработки информации, разработки новых и совершенствовании существующих методов анализа и обработки информации, управления сложными системами, повышения их эффективности. В цифровой Фурье – обработке двумерное дискретное преобразование Фурье (2-D ДПФ) определяется следующим соотношением:

$$S_{N_1, N_2}(k_1, k_2) = \frac{1}{N_1 \cdot N_2} \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} x(n_1, n_2) W_{N_1}^{k_1 n_1} \cdot W_{N_2}^{k_2 n_2} \quad (1)$$

В теории цифровой Фурье – обработки одномерных (1-D) дискретных сигналов широкое применение находит дискретно-временное преобразование Фурье (ДВПФ). В одномерном случае ДВПФ – это z-преобразование, вычисленное в z-плоскости на единичной окружности.

Двумерное дискретно-пространственное преобразование Фурье (2-D ДППФ) – это двумерное z-преобразование, вычисленное в z-пространстве на единичной сфере:

$$X_{\text{ДППФ}}(f_1, f_2) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} x(m, n) \cdot z_1^{-m} z_2^{-n} \Big|_{\substack{z_1 = \exp(+j2\pi f_1) \\ z_2 = \exp(+j2\pi f_2)}}; \quad (2)$$

где f_1 и f_2 – непрерывные пространственные частоты.

Отметим, что как в одномерном случае ДВПФ, так и в двумерном случае ДППФ, представляют собой непрерывные преобразования дискретных 1-D и 2-D последовательностей соответственно и являются чисто теоретическими

понятиями 1-D теории и 2-D теории цифровой Фурье – обработки.

ДППФ двумерного сигнала $x(m, n)$ с фундаментальным периодом размерностью $N_1 \times N_2$ может быть аппроксимировано путем вычисления 2-D ДПФ сигнала $x_1(m, n)$ с фундаментальным периодом размерностью $L_1 \times L_2$:

$$x_1(m, n) = \begin{cases} x(m, n); & m = \overline{0, N_1-1}; n = \overline{0, N_2-1} \\ 0; & m = \overline{L_1 - N_1, L_1-1}; n = \overline{L_2 - N_2, L_2-1} \end{cases} \quad (3)$$

Сравнивая соотношения (1) и (2) несложно видеть, что 2-D ДПФ двумерного сигнала $x(n_1, n_2)$ равно 2-D ДППФ вычисленному на единичной сфере в точках $\exp(-j2\pi k_1 / N_1)$ и $\exp(-j2\pi k_2 / N_2)$. Анализ теоретических основ дискретной двумерной обработки сигналов на базе преобразований Фурье и соотношений (1 – 3) позволяет сделать следующие важные выводы.

Во-первых, 2-D ДПФ следует понимать как двумерный дискретный ряд Фурье, из свойств которого во многом вытекают свойства двумерного ДПФ.

Во-вторых, свойства 2-D ДПФ (как и свойства одномерного ДПФ (1-D ДПФ)) являются математически точными, и не могут трактоваться как аппроксимация свойств непрерывного преобразования Фурье (НПФ). В то же время одномерное и двумерное ДПФ конечной последовательности могут рассматриваться как аппроксимации соответствующих НПФ.

В-третьих, теория двумерной обработки дискретных сигналов на основе двумерного дискретного преобразования Фурье является теори-