

**Тема.** Спектральный анализ.

**Цель работы:** проверка инструмента обработки эксперимента с целью анализа данных и выявления характерных частей.

### *Теоретический минимум*

Мощным инструментом обработки данных, определенных дискретной зависимостью или непрерывной функцией (полученной, например посредством интерполяции или регрессии), является спектральный анализ, имеющий в своей основе различные интегральные преобразования. Спектральный анализ используется как в целях подавления шума, так и для решения других проблем обработки данных.

Задачами, непосредственно связанными со спектральным анализом, являются проблемы сглаживания и фильтрации данных. Они заключаются в построении для исходной экспериментальной зависимости  $y(x_i)$  некоторой (непрерывной или дискретной) зависимости  $f(x)$ , которая должна приближать ее, учитывая к тому же, что данные  $(x_i, y_i)$  получены с некоторой погрешностью, выражающей шумовую компоненту измерений. При этом функция  $f(x)$  с помощью того или иного алгоритма уменьшает погрешность, присутствующую в данных  $(x_i, y_i)$ . Такого типа задачи называют задачами фильтрации. Сглаживание путем построения регрессии данных — это частный случай фильтрации.

Спектром совокупности данных  $y(x)$  называют некоторую функцию другой координаты (или координат)  $F(\omega)$ , полученную в соответствии с определенным алгоритмом. Примерами спектров являются преобразование Фурье и вейвлет-преобразование. Каждое из интегральных преобразований эффективно для решения своего круга задач анализа данных.

Разработка преобразований Фурье сыграла огромную роль в появлении и развитии ряда новых областей науки и техники.

Достаточно отметить, что электротехника переменного тока, электрическая связь и радиосвязь базируются на спектральном представлении сигналов. Ряды Фурье также можно рассматривать как приближение произвольных функций (определенные ограничения в этом известны) тригонометрическими рядами бесконечной длины. При конечной длине рядов получаются наилучшие среднеквадратические приближения. **MATLAB** содержит функции для выполнения быстрого одномерного и двумерного быстрого дискретного преобразования Фурье.

Математический смысл преобразования Фурье (ПФ) состоит в представлении сигнала  $y(x)$  в виде бесконечной суммы синусоид вида

$$F(\omega)\sin(\omega - x),$$

где функция  $F(\omega)$  называется преобразованием Фурье или интегралом Фурье (Фурье-спектром сигнала). Аргумент  $\omega$  имеет смысл частоты соответствующей составляющей сигнала.

Прямое преобразование Фурье переводит описание сигнала (функции времени) из временной области в частотную, а обратное преобразование Фурье переводит описание сигнала из частотной области во временную. На этом основаны многочисленные методы фильтрации сигналов.

Для одномерного массива прямое преобразование Фурье реализуются по следующей формуле:

$$\hat{f}(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(x)e^{-ix\omega} dx.$$

Обратное преобразование Фурье реализуются по следующей формуле:

$$f(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \hat{f}(\omega)e^{ix\omega} d\omega.$$

В терминах обработки сигналов, преобразование берет представление функции сигнала в виде временных рядов и отображает его в частотный спектр, где  $\omega$  — угловая частота. То есть оно превращает функцию времени в функцию частоты; это разложение функции на гармонические составляющие на различных частотах.

Когда функция  $f$  является функцией времени и представляет физический сигнал, преобразование имеет стандартную интерпретацию как спектр сигнала. Абсолютная величина получающейся в результате

комплексной функции  $F$  представляет амплитуды соответствующих частот ( $\omega$ ), в то время как фазовые сдвиги получаются как аргумент этой комплексной функции.

Однако важно осознавать, что преобразования Фурье не ограничиваются функциями времени и временными частотами. Они могут в равной степени применяться для анализа пространственных частот, также как для практически любых других функций.

Для использования в компьютерах, как для научных расчетов, так и для цифровой обработки сигналов, необходимо иметь функции, которые определены на дискретном множестве точек вместо непрерывной области, снова периодическом или ограниченном. В этом случае используется дискретное преобразование Фурье.

Дискретное преобразование Фурье — это одно из преобразований Фурье, широко применяемых в алгоритмах цифровой обработки сигналов (его гомоморфизмы применяются в сжатии звука в MP3, сжатие изображений в jpg и др.), а также в других областях, связанных с анализом частот в дискретном (к примеру, оцифрованном аналоговом) сигнале. Также дискретные преобразования Фурье помогают решать частные дифференциальные уравнения и выполнять такие операции, как свёртки. Преобразования бывают одномерные, двумерные и даже трехмерные.

Последовательность  $N$  действительных чисел  $x_0, \dots, x_{n-1}$  преобразовывается в последовательность из  $N$  комплексных чисел  $X_0, \dots, X_{n-1}$  с помощью дискретного преобразования Фурье по формуле:

$$X_k = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi i}{N} kn} \quad k = 0, \dots, N-1$$

где  $i$  - это мнимая единица. Обратное дискретное преобразование Фурье задается формулой:

$$x_n = \frac{1}{N} \sum_{k=0}^{N-1} X_k e^{\frac{2\pi i}{N} kn} \quad n = 0, \dots, N-1.$$

## Упражнение 1. Функции одномерного преобразования Фурье.

Поскольку напрямую вычисления дискретного преобразования требует  $O(N^2)$  операций, то на практике используют более быстрый алгоритм Быстрого преобразования Фурье, которое требует  $O(N \log N)$  операций.

В описанных ниже функциях реализован особый метод быстрого преобразования Фурье — Fast Fourier Transform (FFT, или БПФ), позволяющий резко уменьшить число арифметических операций в ходе приведенных выше преобразований. Он особенно эффективен, если число обрабатываемых элементов (отсчетов) составляет  $2n$ , где  $n$  — целое положительное число. Используется следующая функция с различным набором аргументов:

- **fft(X)** — возвращает для вектора  $X$  дискретное преобразование Фурье, по возможности используя алгоритм быстрого преобразования Фурье. Если  $X$  — матрица, функция **fft** возвращает преобразование Фурье для каждого столбца матрицы;
- **fft(X,n)** — возвращает  $n$ -точечное преобразование Фурье. Если длина вектора  $X$  меньше  $n$ , то недостающие элементы заполняются нулями. Если длина  $X$  больше  $n$ , то лишние элементы удаляются. Когда  $X$  — матрица, длина столбцов корректируется аналогично;
- **fft(X,[ Ldirn])** и **fft(X,n,dim)** — применяют преобразование Фурье к одной из размерностей массива в зависимости от значения параметра **dim**.

Рассмотрим сначала наиболее типичную для физического эксперимента ситуацию расчета Фурье-спектра действительного сигнала, для которой реализован алгоритм БПФ.

Обратите внимание, что аргумент прямого Фурье-преобразования, т. е. вектор  $u$ , должен иметь ровно  $2^n$  элементов ( $n$  — целое число). Результатом является вектор с  $1 + 2^{n-1}$  элементами. Если число данных не совпадает со степенью 2, то необходимо дополнить недостающие элементы нулями, иначе вместо решения появится сообщение об ошибке.

Чтобы смысл преобразования Фурье был более понятен, используем в качестве модельных данных дискретизацию

детерминированного сигнала, равного сумме трех синусоид. Расчет Фурье-спектра производится по  $N = 128$  точкам, причем предполагается, что интервал дискретизации данных равен  $d$ . В середине листинга применяется встроенная функция **fft**, а его оставшаяся часть предназначена для корректного пересчета соответствующих значений частот ( $N0$ ,  $NN$ ).

Для иллюстрации применения преобразования Фурье создадим трехчастотный сигнал (см. рис. 1) для  $N = 128$ :

```
>> N=128; xMAX=100;d=xMAX/N;i=0:127;t=i*d;
>> x=sin(2*pi*200*t)+0.4*sin(2*pi*150*t)+0.4*sin(2*pi*250*t);
>> plot(x(1:100),'r')
```

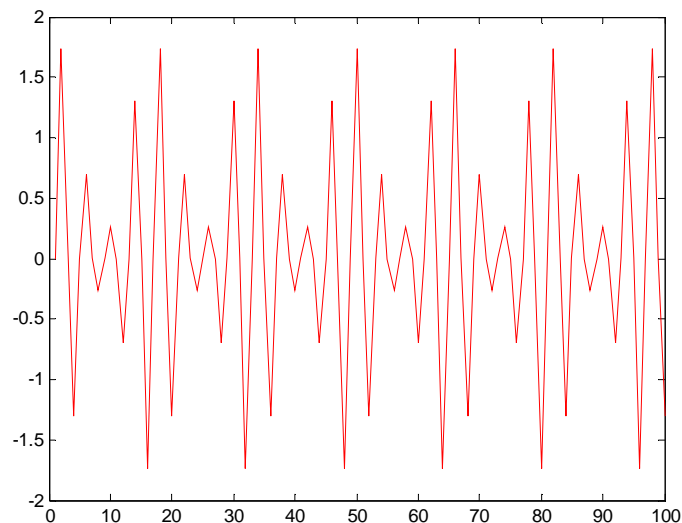


Рис. 1 – Заданный трехчастотный сигнал

Применим прямое преобразование Фурье для созданного сигнала:

```
>> Y=fft(y,1024); plot(f,Pyy(1:151)),grid
```

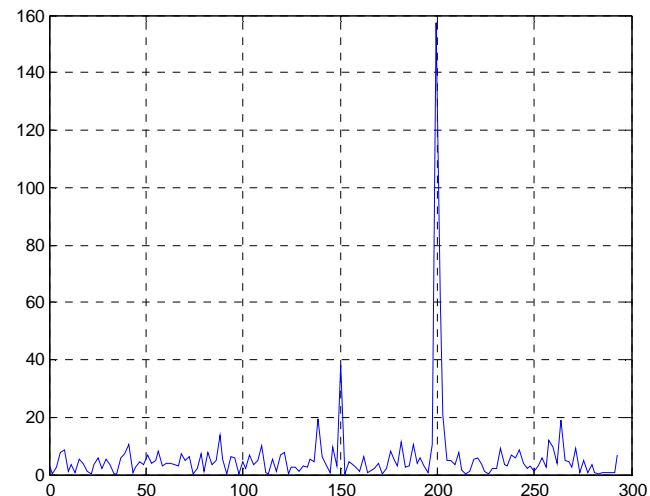


Рис. 2 – График спектральной плотности для заданного сигнала

Очень полезно сравнить полученные амплитуды и местоположение пиков спектра (рис. 1) с определением первоначальных синусоид.

Исключительно важными представляются два параметра, называемые соответственно граничной частотой и частотой Найквиста. Граничная частота  $N0$  определяет нижнюю, а частота Найквиста  $NN$  — верхнюю границу аргумента вычисленного спектра. Кроме того, важно, что интервал дискретизации Фурье-спектра также равен  $N0$ , а общее число вычисляемых точек спектра составляет  $N/2$  (в нашем примере  $N/2=64$ ):

```
>> N0=1/xMAX
N0 =
    0.0100
>> NN=N/(2*xMAX)
NN =
    0.6400
>> ND=(i+1)*N0
ND =
Columns 1 through 4
    0.0100    0.0200    0.0300    0.0400
Columns 5 through 8
```

```

0.0500 0.0600 0.0700 0.0800
.....
Columns 61 through 64
0.6100 0.6200 0.6300 0.6400
Column 65
0.6500

```

## Упражнение 2. Функции обратного преобразования Фурье.

Для расчета обратного Фурье-преобразования (восстановления сигнала по имеющемуся действительному спектру) следует использовать следующие встроенные функции (они также реализуют алгоритм БПФ):

- **ifft(F)** — возвращает результат дискретного обратного преобразования Фурье вектора F. Если F — матрица, то ifft возвращает обратное преобразование Фурье для каждого столбца этой матрицы;
- **ifft(F,n)** — возвращает результат n-точечного дискретного обратного преобразования Фурье вектора F;
- **ifft(F,[],dim)** и **y = ifft(X,n,dim)** — возвращают результат обратного дискретного преобразования Фурье массива F по строкам или по столбцам в зависимости от значения скаляра dim.

Для любого X результат последовательного выполнения прямого и обратного преобразований Фурье  $\text{ifft}(\text{fft}(x))$  равен X с точностью до погрешности округления. Если X — массив действительных чисел,  $\text{ifft}(\text{fft}(x))$  может иметь малые мнимые части.

Пример 1. Для исходного вектора применим прямое и обратное преобразование Фурье:

```

>> V=[1 1 1 1 0 0 0 0];
>> fft(V)
ans =
Columns 1 through 4
4.0000 1.0000 - 2.4142i 0 1.0000 - 0.4142i
Columns 5 through 8
0 1.0000 + 0.4142i 0 1.0000 + 2.4142i
>> ifft(fft(V))
ans =
1 1 1 1 0 0 0 0

```

Обратите внимание, что аргумент функций, реализующих обратное преобразование Фурье, может быть как действительным, так и комплексным. А вот результат их работы является вектором, составленным из действительных чисел. Если аргумент является N-компонентным вектором, где  $N = 1 + 2^n$ , то в результате получается в два раза больший вектор из  $2(N - 1) = 2^{n+1}$  компонент.

## Упражнение 3. Анализ спектра модели «сигнал+шум».

Пока мы использовали в качестве примера детерминированный сигнал, представляющий собой сумму трех синусоид. Несмотря на единство термина "дискретное преобразование Фурье", прикладное применение спектрального анализа можно довольно четко разделить на две категории.

Сигнал, подвергающийся спектральному анализу, получен в условиях пренебрежимо малой погрешности, т. е. его можно, фактически, считать детерминированным. Такая ситуация характерна для экспериментальной оптики и (разного рода) спектроскопии. В этом случае для большинства задач анализа сигналов бывает вполне достаточно использовать простые спектры Фурье, рассмотренные выше..

Сигнал, полученный в присутствии значительной шумовой компоненты, которая существенно искажает его структуру. В этом случае следует говорить о смеси (к счастью, чаще всего аддитивной) "полезный сигнал + шум", причем в большинстве случаев заранее известна некоторая информация о статистике шумовой компоненты. Данная ситуация очень часто встречается в экспериментальной геофизике и радиофизике. В этом случае подходить к интерпретации спектров следует с вероятностной точки зрения. Как раз этому вопросу мы и посвятим данный пример.

Для иллюстрации применения преобразования Фурье создадим трехчастотный сигнал на фоне сильного шума, создаваемого генератором случайных чисел, а именно внесем минимальное добавление в виде четвертого слагаемого сигнала: псевдослучайную величину **randn**:

```

>> t=0:0.0005:1;
>> x=sin(2*pi*200*t)+0.4*sin(2*pi*150*t)+0.4*sin(2*pi*250*t);

```

```
>> y=x+2*randn(size(t));
>> plot(y(1:100),'b')
```

Этот сигнал имеет среднюю частоту 200 рад/с и два боковых сигнала с частотами 150 и 250 рад/с, что соответствует амплитудно-модулированному сигналу с частотой модуляции 50 рад/с и глубиной модуляции 0.8 (амплитуда боковых частот составляет 0.4 от амплитуды центрального сигнала). На рис. 3 показан график этого сигнала (по первым 100 отсчетам из 2000). Нетрудно заметить, что из него никоим образом не видно, что полезный сигнал — амплитудно-модулированное колебание, настолько оно забито шумами:

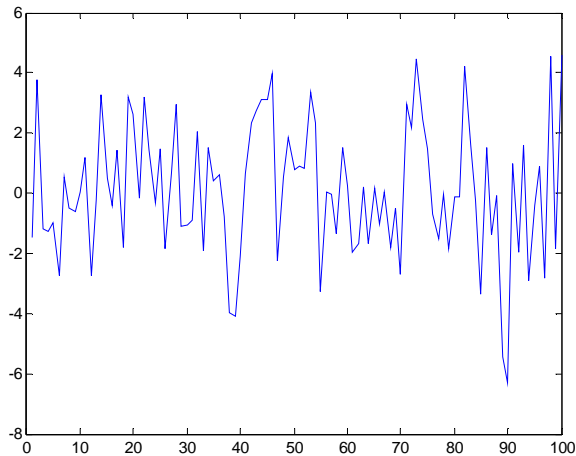


Рис. 3 – Заданный сигнал с шумовой компонентой

Теперь построим график спектральной плотности полученного сигнала с помощью прямого преобразования Фурье, по существу переводящего временное представление сигнала в частотное. Этот график в области частот до 300 Гц строится с помощью следующих команд:

```
>> Y=fft(y,1024); Pyy=Y.*conj(Y)/1024;
>> f=2000*(0:150)/1024; plot(f,Pyy(1:151)),grid
```

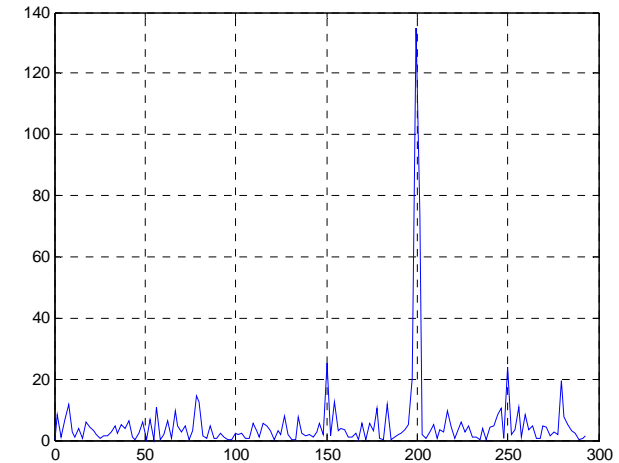


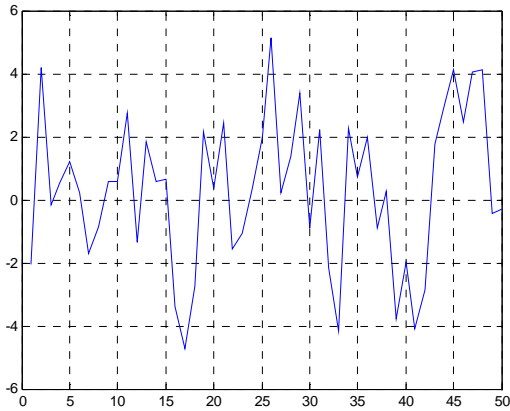
Рис. 4 – График спектральной плотности для заданного сигнала с шумовой компонентой

График спектральной плотности сигнала, построенный в этом примере, представлен на рис. 4. Даже беглого взгляда на рисунок достаточно, чтобы убедиться в том, что спектрограмма сигнала имеет явный пик на средней частоте амплитудно-модулированного сигнала и два боковых пика. Все эти три частотные составляющие сигнала явно выделяются на общем шумовом фоне. Таким образом, данный пример наглядно иллюстрирует технику обнаружения слабых сигналов на фоне шумов, лежащую в основе работы радиоприемных устройств.

Проанализируем действие прямого и обратного преобразования Фурье на сигнал с шумовой компонентой. Рассмотрим данные, поступающие с частотой 1000 Гц. Сформируем сигнал, содержащий регулярные составляющие с частотами 50 Гц и 120 Гц и случайную аддитивную компоненту с нулевым средним.

```
>> t = 0:0.001:0.6;
>> x = sin(2 * pi * 50 * t) + sin(2 * pi * 120 * t);
>> y = x + 2 * randn(size(t));
>> plot(y(1:50)), grid
```

На рис. 5 показан этот сигнал.

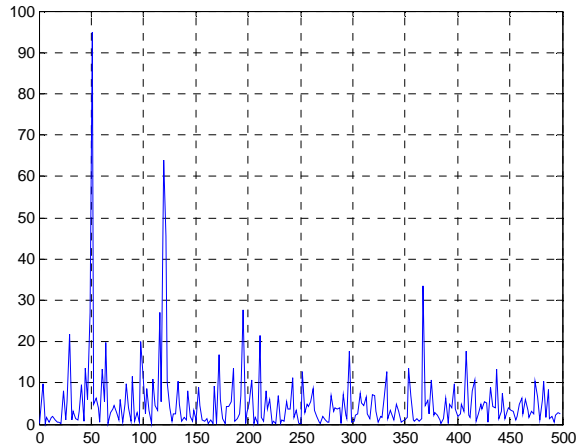


**Рис. 5 – График двухсусидального сигнала в присутствии шумов в виде функции случайной величины**

Глядя на него, трудно сказать, каковы частоты его регулярных составляющих. Реализуя одномерное преобразование Фурье этого сигнала на основе 512 точек:

```
>> Y = fft(y, 512); Pyy = Y.*conj(Y)/512;
>> f = 1000 * (0:255)/512; figure(2), plot(f, Pyy(1:256)), grid
```

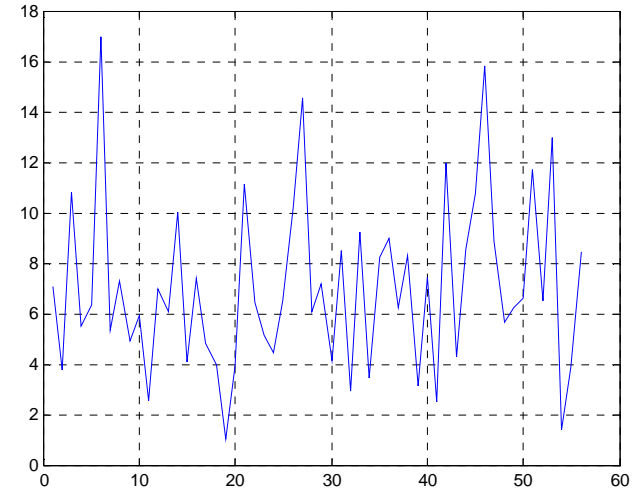
и построив график спектральной плотности (рис. б), можно выделить две частоты, на которых амплитуда спектра максимальна. Это частоты 120 и 50 Гц:



**Рис. 6 – График спектральной плотности исходного сигнала**

Применим к полученному массиву обратное быстрое преобразование Фурье и выведем полученные данные на график:

```
> figure;plot(1:56,abs(y))
>> plot(1:56,abs(y));grid
```



**Рис. 7 – Результат действия обратного преобразования Фурье**

**Практические задания лабораторной работы № \_\_\_\_\_**

**Задание 1.** Построить график исходной заданной функции:

№ п/п	Частота поступающего сигнала	Частоты регулярных составляющих		
		$\omega_1$	$\omega_2$	$\omega_3$
1	1000	100	50	300
2	2000	200	60	150
3	1500	300	70	200
4	2500	150	80	220
5	3000	160	90	230
6	1200	210	100	310
7	1300	170	90	250
8	2200	180	80	130
9	2100	210	70	140
10	1100	220	60	150

11	1200	230	50	160
11	1600	240	40	170
12	1700	250	110	180
13	1800	260	120	120
14	1900	270	130	330

**Задание 2.** Построить график спектральной плотности для исходного сигнала, включив четвертым слагаемым шумовую составляющую в виде функции случайной величины.

**Задание 3.** Применяв обратное преобразование Фурье, получить исходный сигнал.

### Контрольные вопросы

1. Назовите области применения спектрального анализа.
2. Что такое фильтрация данных?
3. Применительно к описанию сигнала приведите формулировки прямого и обратного преобразований Фурье.
4. Что такое дискретное преобразование Фурье?
5. Перечислите варианты вызова функций Matlab, реализующих быстрое прямое и обратное преобразования Фурье.