

## 6.1. ЭЛЕМЕНТЫ ТЕОРИИ

Значительная часть задач анализа временных рядов связана с преобразованием Фурье и методами его эффективного вычисления. В этих задачах преобразование Фурье играет важную роль не только само по себе, но и как необходимый промежуточный шаг в определении плотностей спектра мощности, кросс-спектральных плотностей, передаточных функций, сверток и корреляционных функций (табл. 6.1). Кроме того, преобразование Фурье можно использовать в задачах интерполяции значений. Эта глава содержит довольно краткий обзор теоретических сведений, поскольку главный упор сделан на те свойства преобразования, которые важны в приложениях и позволяют проводить вычисления разных функций наиболее быстро и экономно. Подробнее с преобразованием Фурье и его применениями читатель может ознакомиться по многим книгам. Назовем весьма полезные для наших целей монографии Колмогорова и Фомина (1976), Винера (1963).

Вопросы, связанные с алгоритмом быстрого преобразования Фурье, освещены здесь весьма скупно. Читателю, которому потребуется более широкое знакомство с этими вопросами, рекомендуем обратиться к книгам Отнеса и Эноксона (1972) и Ахо, Хопкрофта и Ульмана (1979).

**Определение преобразования Фурье.** Приведем сначала уже упомянутое нами раньше определение преобразования Фурье на бесконечной прямой:

$$X(f) = \int_{-\infty}^{\infty} x(t) \exp(-j2\pi ft) dt, \quad -\infty < f < \infty.$$

В этом определении функция  $x(t)$  обычно бывает временной траекторией (хотя и не всегда). В таких случаях переменной  $t$  служит время, а переменной  $f$  — частота. Вопрос существования преобразования Фурье  $X(f)$  для случайной функции  $x(t)$  с математической точки зрения довольно сложен. Однако конечная длина записи и ограниченные амплитуды временных траекторий снимают этот вопрос, поскольку для таких траекторий преобразование Фурье всегда существует.

## Функции, получаемые с помощью БПФ

1. Плотность спектра мощности

$$S_{xx}(k) = \frac{1}{P} \overline{X^*(k) X(k)} = \frac{1}{P} |\overline{X(k)}|^2$$

2. Плотность кросс-спектра

$$S_{xy}(k) = \frac{1}{P} \overline{X^*(k) Y(k)}$$

3. Передаточная функция

$$-H_{xy}(k) = \frac{Y(k)}{X(k)}$$

4. Свертка <sup>a)</sup>

$$u(i) = \text{БПФ}^{-1}[U(k)], \quad U(k) = X(k) Y(k)$$

5. Ковариация <sup>a)</sup>

$$s_{xy}(i) = \frac{N}{N-i} \text{БПФ}^{-1}[S_{xy}(k)]$$

$$S_{xy}(k) = \frac{1}{P} \overline{X^*(k) Y(k)}$$

<sup>a)</sup> Для ковариации и свертки по крайней мере последние половины последовательностей  $X(k)$  и  $Y(k)$  могут быть нулями.

Величины  $x(i)$ ,  $y(i)$  и  $u(i)$ , где  $i=0, 1, \dots, N-1$ , суть конечные временные последовательности:  $X(k)$ ,  $Y(k)$  и  $U(k)$  — соответствующие БПФ ( $k=0, 1, \dots, \frac{N}{2}$  при четном  $N$ ,  $k=0, 1, \dots, \frac{N-1}{2}$  при нечетном  $N$ ). Черта означает осреднение.

Обратимся теперь к преобразованию Фурье, определенному на конечном интервале дискретных значений времени. Это преобразование вычисляется, как правило, при помощи алгоритма быстрого преобразования Фурье. Определение преобразования для комплексной функции  $z(i)$ ,  $i=0, 1, \dots, N-1$ , имеет в этом случае вид

$$Z(k) = T \sum_{i=0}^{N-1} z(i) \exp\left(-j \frac{2\pi}{N} ik\right), \quad \text{где } k=0, 1, \dots, N-1. \quad (6.1)$$

Напомним, что в этом определении для краткости вместо  $z(iT)$  используется обозначение  $z(i)$ . Такое же сокращение записи от  $Z(bk)$  к  $Z(k)$  для области частот обусловлено конечной длиной записи  $z(i)$ , приводящей к дискретным значениям частот; величина  $b$  определяется из соотношений

$$b = \frac{1}{NT} = \frac{1}{P}.$$

Такое определение величины  $b$  позволяет в условиях применимости теоремы о выборке во временной области пользоваться вариантом этой теоремы для области частот и наоборот. Знание  $N$  значений в области частот позволяет восстановить все  $N$  значений во временной области. Более того, для ограниченных во временной и частотной областях функции и ее изображения все промежуточные значения можно восстановить, используя преобразования как интерполяционные операторы (см. теорему о вы-

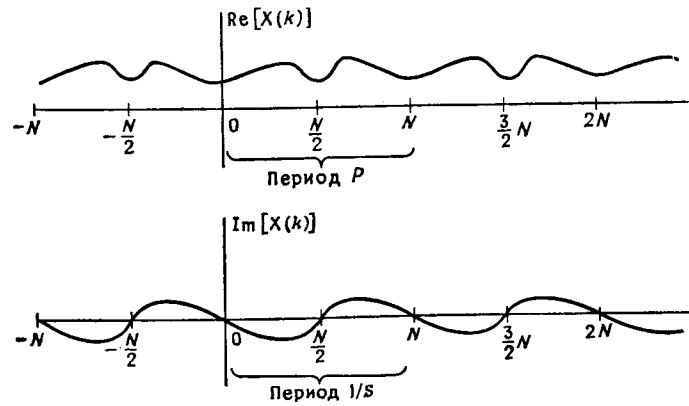


Рис. 6.1. Иллюстрация периодичности дискретного преобразования Фурье.

борке). Отметим периодичность  $Z(k)$ . Если попытаться продолжать вычисления при  $k > N$ , то полученные значения будут повторять уже имеющиеся значения  $Z(k)$ , т. е.

$$Z(N+k) = Z(k)$$

(рис. 6.1).

На практике в задачах анализа временных рядов обрабатываются не комплексные, а действительные величины. Поэтому определение (6.1) можно переписать в виде

$$X(k) = T \sum_{i=0}^{N-1} x(i) \exp\left(-j \frac{2\pi}{N} ik\right), \quad k = \begin{cases} 0, 1, \dots, \frac{N}{2} & (N - \text{четное}), \\ 0, 1, \dots, \frac{N-1}{2} & (N - \text{нечетное}). \end{cases} \quad (6.2)$$

Таким образом, для  $N$  значений данных получается примерно вдвое меньше значений частот. Если продолжать вычисления при следующих значениях  $k$ , то будут получаться числа, комплексно-

сопряженные уже имеющимся для первых значений  $k$ , а именно

$$X^*(N-k) = X(k) \quad (6.3)$$

(рис. 6.2).

Это непосредственно вытекает из формулы (6.2) и отражает тот факт, что косинус — четная, а синус — нечетная функции.

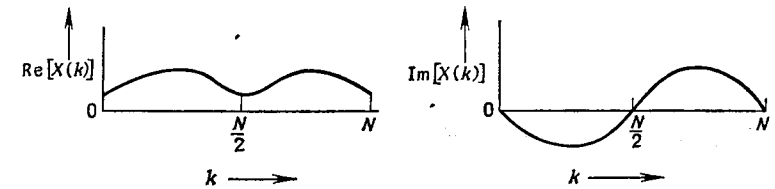


Рис. 6.2. Поведение, типичное для четной действительной и нечетной мнимой частей преобразования Фурье действительных функций.

Действительно,

$$\operatorname{Re}[X(k)] = T \sum_{i=0}^{N-1} x(i) \cos\left(\frac{2\pi}{N} ik\right)$$

есть четная (симметрическая) функция (т. е.  $\operatorname{Re} X(k) = \operatorname{Re} X(-k)$ ), а

$$\operatorname{Im}[X(k)] = -T \sum_{i=0}^{N-1} x(i) \sin\left(\frac{2\pi}{N} ik\right)$$

есть нечетная (антисимметрическая) функция ( $\operatorname{Im} X(k) = -\operatorname{Im} X(-k)$ ). Доказательство предлагаем провести читателю (см. упр. 6.3).

Нетрудно видеть, что общее число действительных и мнимых частей частотной функции равно числу исходных данных временной функции. Отметим, что

$$X(0) = X^*(N) = X^*(0), \quad (6.4)$$

и для четных  $N$

$$X\left(\frac{N}{2}\right) = X^*\left(N - \frac{N}{2}\right) = X^*\left(\frac{N}{2}\right). \quad (6.5)$$

Поэтому как в нулевой частоте, так и в частоте Найквиста значения преобразования оказываются действительными. В случае нечетного  $N$  величина  $X([N-1]/2)$  не попадает точно в частоту Найквиста и, следовательно, имеет мнимую часть. Различие случаев четного и нечетного  $N$  отражено на рис. 6.3.

Заметим, что большинство современных мини-ЭВМ, используемых в системах для анализа Фурье, имеет алгоритм преобразования для последовательностей с числом членов, равным степени 2 (т. е. четным). Однако в больших вычислительных сис-

темах используются алгоритмы, в которых число данных может быть равным не только степени 2, но и 3, 4 и 5 (Синглтон (1969)). В этих случаях необходимо учитывать различия, связанные с четностью и нечетностью числа данных.

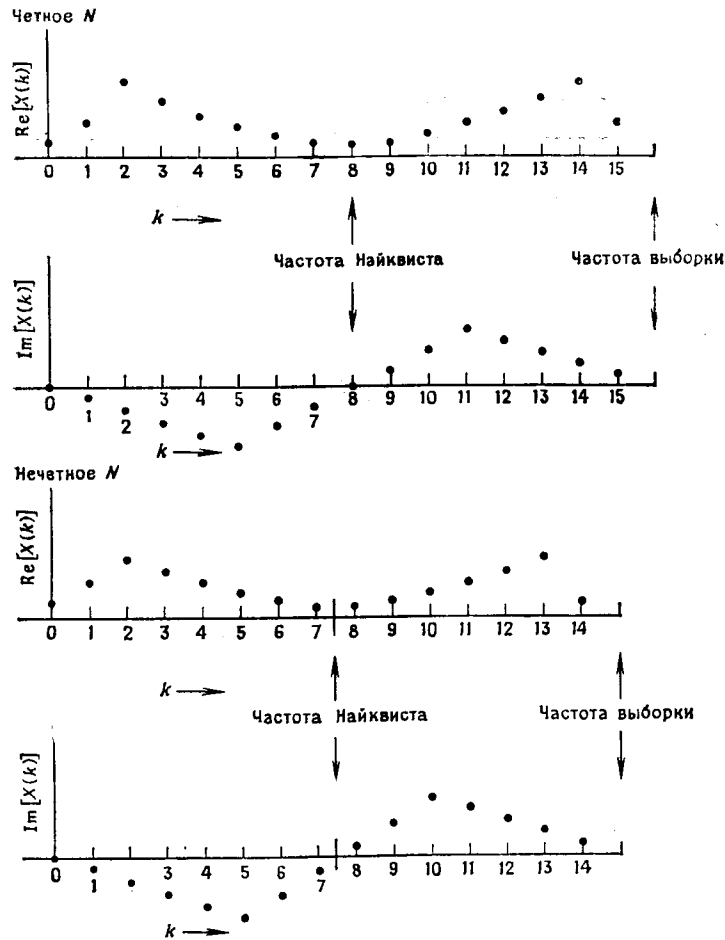


Рис. 6.3. Различия преобразований Фурье в случаях четного и нечетного числа точек.

Из формул (6.3)–(6.5) и четности действительной части преобразования Фурье вытекает, что  $N$  действительных компонент, полученных преобразованием из  $N$  действительных чисел, при четном  $N$  дают пару действительных (в частотах, соответствующих  $k=0$  и  $k=N/2$ ) и  $(N-2)/2$  комплексных чисел (на дей-

ствительную и мнимую части которых уходит по две компоненты); при нечетном  $N$  только одна компонента ( $k=0$ ) дает действительное число, а остальные  $N-1$  уходят на  $(N-1)/2$  комплексных чисел.

Следует отметить, что эти детали хотя и имеют определенное значение для правильного построения программ для ЭВМ, но в научно-технических задачах, связанных с преобразованием Фурье, не играют важной роли.

**Полярные координаты (модуль и фаза).** Значение преобразования Фурье в данной частоте — комплексное число — нередко записывают в полярных координатах

$$|X(k)| = \{\text{Re}^2[X(k)] + \text{Im}^2[X(k)]\}^{1/2},$$

$$\varphi_x(k) = \text{Arctg} \frac{\text{Im}[X(k)]}{\text{Re}[X(k)]}, \quad (6.6)$$

где обозначение  $\text{Arctg}$  введено для главного значения.

Величину  $|X(k)|$  называют по-разному — коэффициентом усиления или модулем. Величину  $\varphi_x(k)$  называют фазой или аргументом. На рис. 6.4 в качестве иллюстрации приведены графики действительной части, мнимой части, модуля и фазы преобразования Фурье.

**Обратное преобразование Фурье.** Чтобы получить формулу, определяющую обратное преобразование Фурье, достаточно в определении (6.2) заменить знаки показателей экспонент на противоположные и поменять ролями  $x(i)$  и  $X(k)$ , т. е.

$$x(i) = b \sum_{k=0}^{N-1} X(k) \exp \left[ j2\pi \frac{ik}{N} \right], \quad \text{где } i=0, 1, \dots, N-1. \quad (6.7)$$

Отметим, что пределы суммирования в этой формуле требуют знания  $N$  значений  $X(k)$ . Поскольку величины  $X(k)$  определялись только для  $k$  от 0 до  $N/2+1$  или  $(N-1)/2$ , недостающие величины нужно получить, исходя из симметрии по Эрмиту значений преобразования Фурье действительной последовательности, т. е. по формуле (6.3).

Например, действительная и мнимая части преобразования Фурье действительной функции могут выглядеть так, как это показано на рис. 6.1. Неявно предполагается, что для этого преобразования  $N$  — четное число. Случай нечетного  $N$  будет рассмотрен ниже.

Нетрудно понять, что комплексно-сопряженную симметричную часть функции  $X(k)$  совсем необязательно непосредственно вводить в правую часть (6.7). В пункте, посвященном специальному обратному преобразованию Фурье, рассмотрен метод, в котором предусмотрена такая возможность удаления лишних значений.

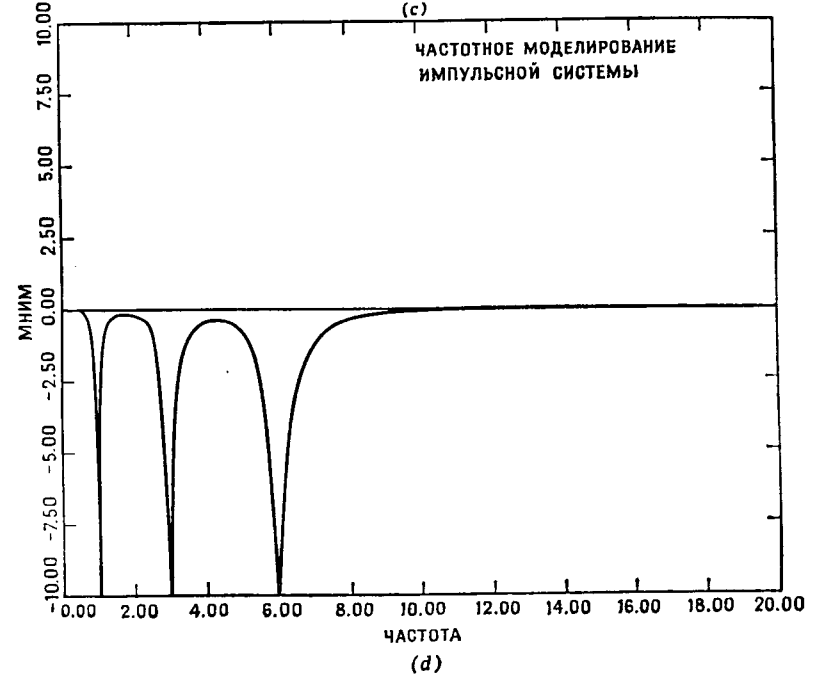
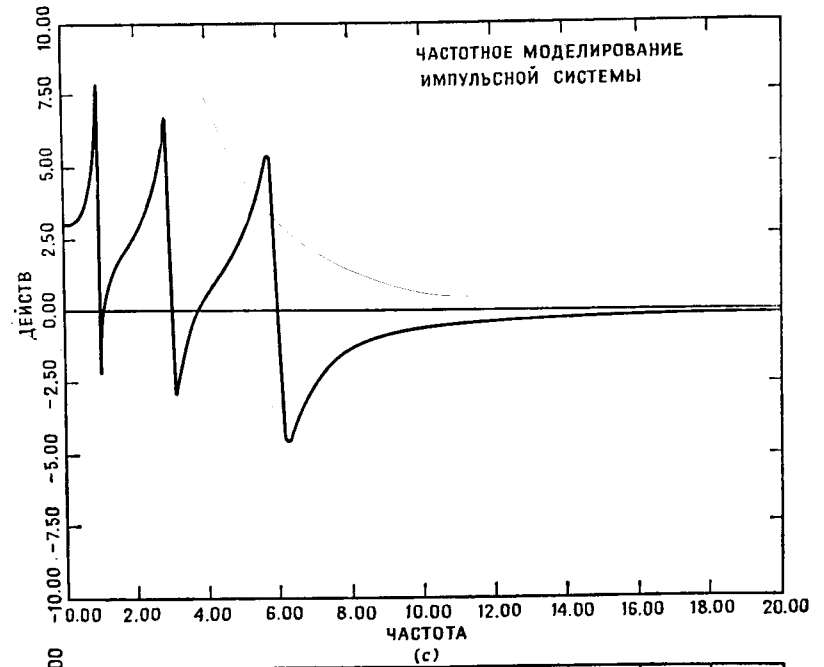
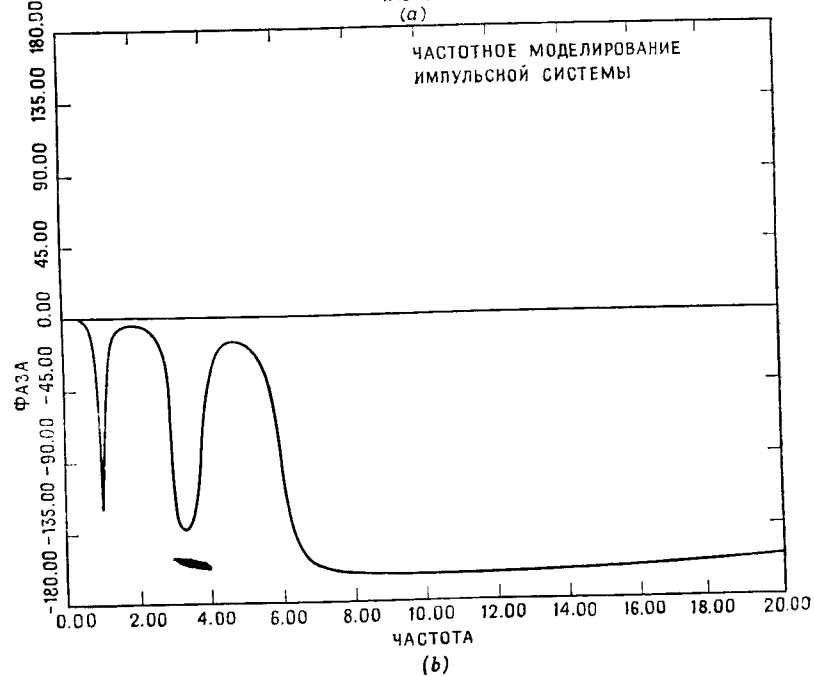
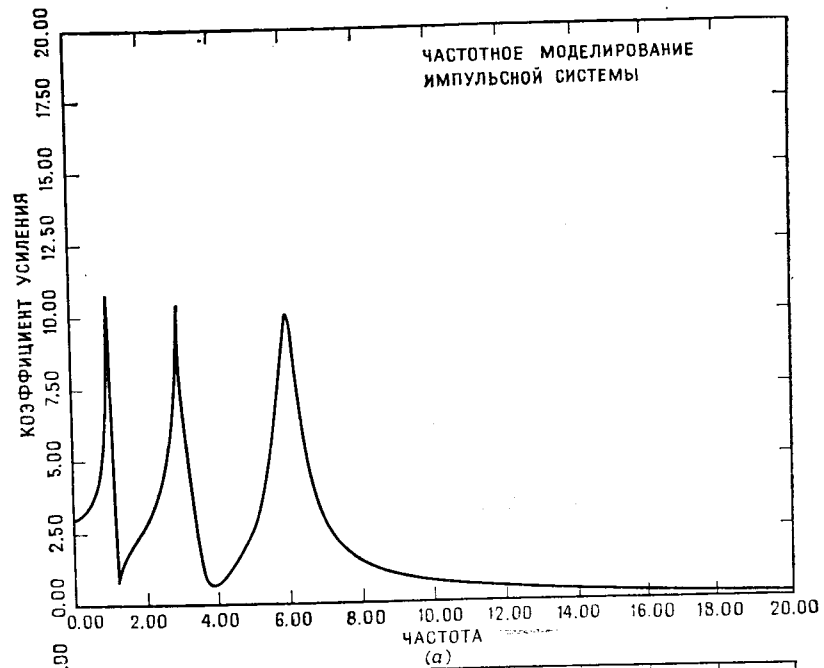


Рис. 6.4. Сопоставление декартовых и полярных координат. *a* — абсолютные

величины; *b* — фаза; *c* — действительная часть; *d* — мнимая часть.

Далее, при вычислении преобразования Фурье действительной последовательности с помощью алгоритма комплексного преобразования Фурье необязательно заполнять нулями весь массив, отведенный под мнимые части. В этом случае действительную последовательность лучше разнести по обоим массивам, отведенным под действительные и мнимые части.

Отметим, что в определениях прямого и обратного дискретных преобразований Фурье в явном виде присутствуют временной и частотный интервалы выборок. Во многих учебниках и статьях приводится несколько измененный вариант определения преобразования, соответствующий случаю нормированного временного интервала  $T$ , равного 1. При этом шаг  $b$ , с которым чередуются частоты, равен

$$P^{-1} = (NT)^{-1} = N^{-1}.$$

Формулы для преобразований имеют в этом случае вид

$$X(k) = \sum_{i=0}^{N-1} x(i) \exp \left[ -j2\pi \frac{ik}{N} \right],$$

$$x(i) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) \exp \left[ j2\pi \frac{ik}{N} \right].$$

Теперь рассмотрим несколько соотношений, полезных при вычислениях преобразования Фурье. Одно из них — формула преобразования «двойной длины» — служит основой алгоритма быстрого преобразования Фурье с числом данных, равным степени 2.

**Спаренное преобразование.** Обычные алгоритмы быстрого преобразования Фурье предназначены для последовательностей комплексных данных, хотя на практике временные ряды представляют собой последовательности действительных величин. К счастью, алгоритмы для комплексных последовательностей нетрудно приспособить для вычисления преобразований действительных последовательностей. Допустим, что имеются две временные последовательности  $x(i)$  и  $y(i)$ ,  $i=0, 1, \dots, N-1$ . Представим эти две последовательности в качестве действительной и мнимой частей комплексной последовательности  $z(i)$ . Определив преобразование Фурье

$$Z(k) = T \sum_{i=0}^{N-1} z(i) \exp \left[ -j2\pi \frac{ik}{N} \right], \quad \text{где } k=0, 1, \dots, N-1,$$

можно получить из него преобразования  $X(k)$  и  $Y(k)$  по формулам

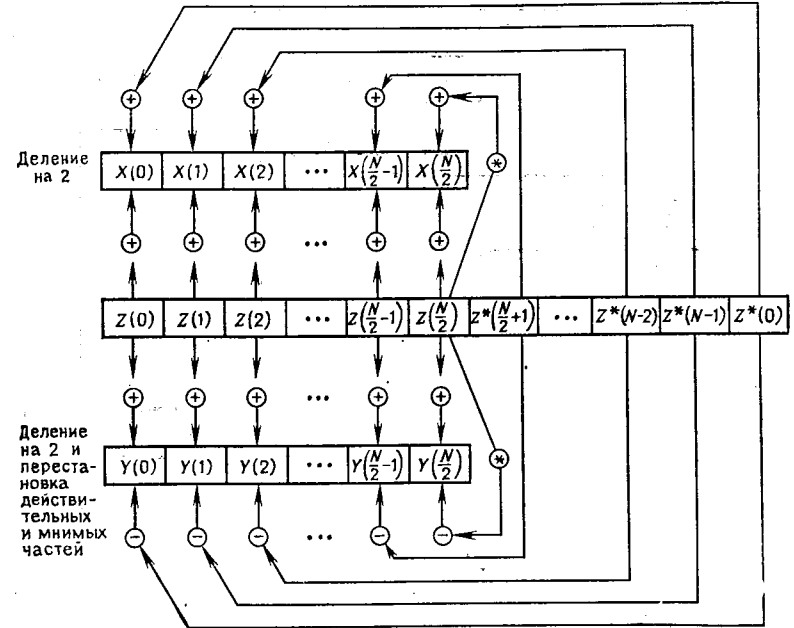
$$X(k) = \frac{Z(k) + Z^*(N-k)}{2}, \tag{6.8}$$

$$Y(k) = \frac{Z(k) - Z^*(N-k)}{2j}, \tag{6.9}$$

где

$$k = \begin{cases} 0, 1, \dots, \frac{N}{2}, & \text{если } N \text{ — четное,} \\ 0, 1, \dots, \frac{N-1}{2}, & \text{если } N \text{ — нечетное.} \end{cases}$$

При этом получается по  $N/2$  значений преобразования. Как отмечалось выше, для действительных последовательностей этого достаточно. Схема, иллюстрирующая вычисление преобразований по формулам (6.8) и (6.9), изображена на рис. 6.5.



6.5. Диаграмма вычисления спаренного преобразования расщеплением.

**Преобразование двойной длины.** Так называется метод получения преобразования Фурье (комплексной) последовательности  $z(i)$ , где  $i=0, 1, \dots, 2N-1$ , по преобразованиям Фурье пары последовательностей вдвое меньшей длины. В этом методе по последовательности  $z(i)$  строятся последовательности

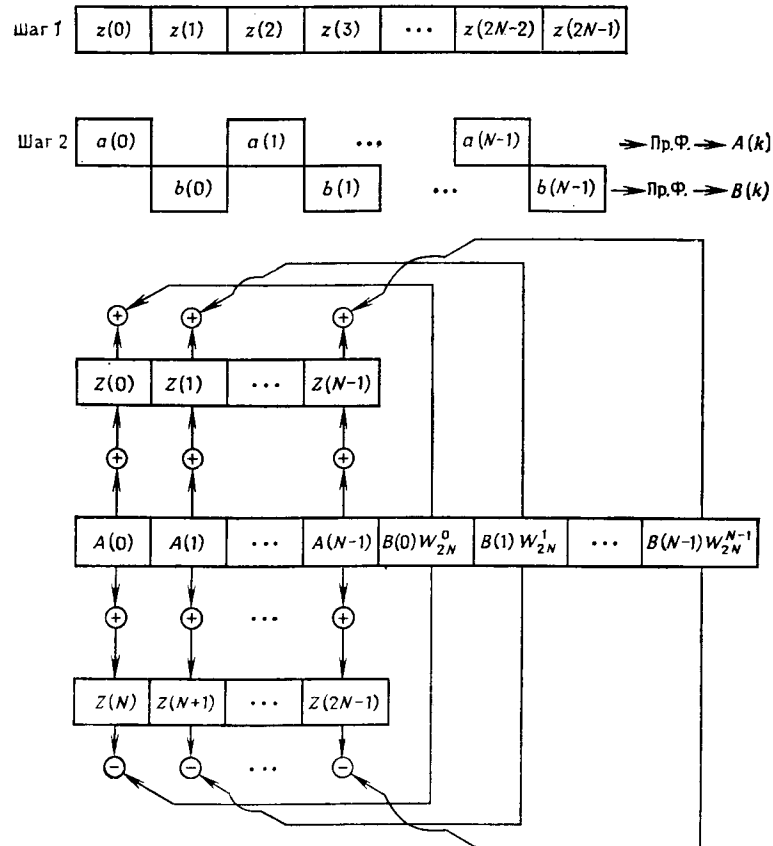
$$a(i) = z(2i), \quad b(i) = z(2i+1),$$

где  $i=0, 1, \dots, N-1$ . После этого вычисляются преобразования

$$A(k) = T \sum_{i=0}^{N-1} a(i) \exp \left( -j2\pi \frac{ik}{N} \right)$$

и

$$B(k) = T \sum_{i=0}^{N-1} b(i) \exp\left(-j2\pi \frac{ik}{N}\right),$$

где  $k = 0, 1, \dots, N-1$ .

6.6. Диаграмма вычисления преобразования двойной длины.

Нужную последовательность значений преобразования Фурье получают по формулам

$$Z(k) = A(k) + B(k) W_{2N}^k, \quad Z(N+k) = A(k) - B(k) W_{2N}^k, \quad (6.10)$$

где  $k = 0, 1, \dots, N-1$  и

$$W_{2N} = \exp\left(-j \frac{2\pi}{2N}\right) = \exp\left(-j \frac{\pi}{N}\right).$$

Заметим, что для получения преобразования временной последовательности длины  $N$  можно воспользоваться формулами (6.10) для комплексного преобразования длины  $N/2$ . Схема вычисления по формулам (6.10) изображена на рис. 6.6.

**Ряды Фурье.** Понятие ряда Фурье (см. Колмогоров и Фомин (1976), с. 406)

$$x(t) = \frac{1}{2} a_0 + \sum_{k=1}^{\infty} (a_k \cos(k\omega_0 t) + b_k \sin(k\omega_0 t)),$$

где  $\omega_0 = 2\pi/P$ , тесно примыкает к комплексному преобразованию Фурье. Дискретным вариантом ряда Фурье служит разложение

$$x(i) = \left[ \frac{1}{2} a_0 + \sum_{k=1}^{N/2} \left( a_k \cos \frac{2\pi ik}{N} + b_k \sin \frac{2\pi ik}{N} \right) \right],$$

где

$$a_k = 2T \sum_{i=0}^{N-1} x(i) \cos \frac{2\pi ik}{N}, \quad b_k = 2T \sum_{i=0}^{N-1} x(i) \sin \frac{2\pi ik}{N}.$$

Заметим, что

$$a_k = X(k) + X^*(k), \quad b_k = -[X(k) - X^*(k)],$$

т. е. классический ряд Фурье периодической функции можно получить непосредственно из комплексного преобразования Фурье.

Однако следует отметить, что с математической точки зрения ряды и преобразования Фурье имеют существенное различие. Для исследования преобразования Фурье случайной функции требуются более сложные математические построения, чем для рядов Фурье детерминированных функций. Главное различие связано с вопросами существования или сходимости интегралов, входящих в определения. Впрочем, в практическом анализе данных редко приходится сталкиваться с этими тонкостями. Более того, те вычислительные алгоритмы, которые используются в анализе данных, равно хорошо применимы как для вычисления разложения в ряд Фурье детерминированной периодической функции, так и для получения конечного преобразования Фурье цифровой выборки случайного процесса на бесконечном интервале.

**Вычисление обратного преобразования.** Как уже было отмечено, преобразование Фурье действительной последовательности требует определения только  $N/2$  (или  $(N-1)/2$ ) комплексных величин. Это обусловлено комплексной сопряженностью значений преобразования действительной последовательности (см. рис. 6.1). Наоборот, такая комплексная сопряженность симметричных относительно середины последовательности значений

преобразования приводит при обратном преобразовании Фурье к действительной последовательности. Очевидно, что в этом случае вторая половина значений прямого преобразования Фурье не несет никакой дополнительной информации и может быть отброшена. На этом соображении основано вычисление обратного преобразования. Сначала вычисляется последовательность значений  $U(k)$  по формуле

$$U(k) = Z(k) + Z^*(M-k) + j[Z(k) - Z^*(M-k)]W_M^{-k/2}, \quad (6.11)$$

где  $k=0, 1, \dots, M-1$ ,  $M=N/2$  или  $(N-1)/2$ , а  $Z(k)$  — последовательность длины  $N$  значений преобразования Фурье последовательности  $z(i)$ . Из общего числа значений для вычислений по формуле (6.11) требуется только  $N/2$  или  $(N-1)/2$ , что и позволяет не учитывать комплексно-сопряженной части преобразования последовательности. Само вычисление обратного преобразования Фурье проводится по формуле

$$\begin{aligned} z(2i) &= \operatorname{Re} u(i), \\ z(2i+1) &= \operatorname{Im} u(i), \quad i=0, 1, \dots, M-1, \end{aligned} \quad (6.12)$$

в которой действительные и мнимые части чисел  $u(i)$  определяются из обратного преобразования Фурье последовательности  $U(k)$ :

$$u(i) = b \sum_{k=0}^{M-1} U(k) \exp \left[ j \frac{2\pi ik}{M} \right].$$

Преимущество способа (6.12) расположения результатов связано с тем, что при работе на Фортране именно таким образом в памяти ЭВМ на сердечниках располагаются соответствующие комплексные числа. Схема вычисления показана на рис. 6.7.

Чтобы получить обратное преобразование Фурье двух функций одновременно, можно, обратив соотношения (6.8) и (6.9), т. е. записав их в виде

$$Z(k) + Z^*(N-k) = 2X(k), \quad Z(k) - Z^*(N-k) = 2jY(k),$$

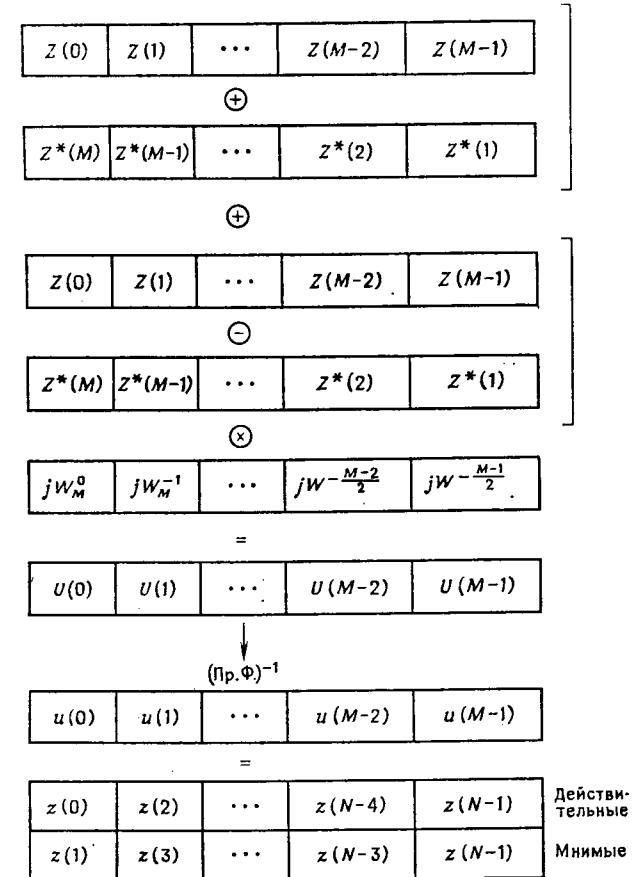
получить значения преобразования  $Z(k)$  и  $Z^*(N-k)$ . Для этого нужно сложить оба равенства и вычесть второе из первого:

$$\begin{aligned} Z(k) &= X(k) + jY(k), \\ Z^*(N-k) &= X(k) - jY(k). \end{aligned}$$

Здесь

$$k = \begin{cases} 0, 1, \dots, \frac{N}{2}, & \text{если } N \text{ — четное,} \\ 0, 1, \dots, \frac{N-1}{2}, & \text{если } N \text{ — нечетное.} \end{cases}$$

Получив таким образом  $N$  значений, следует применить обратное преобразование Фурье. В результате действительные последовательности  $x(i)$  и  $y(i)$  будут получены как действительные и мнимые части последовательности  $z(i)$  соответственно.



6.7. Иллюстрация вычисления обратного преобразования по формулам (6.11) и (6.12).

**Шаг в области частот как функция длины записи.** В следующем параграфе мы остановимся на двух алгоритмах быстрого преобразования Фурье (БПФ) для последовательностей, число членов которых равно степени 2 или, в более общем случае, произведению степеней 2, 3, 4 и 5 (т. е.  $N = 2^a 3^b 4^c 5^d$ ). Если число членов последовательности не совпадает в точности с одним из этих чисел, то прибегают к увеличению длины последова-

тельности за счет добавления нулей в начале и конце. При этом изменяется шаг в области частот. Если исходная длина равна  $N$ , то при получении преобразования Фурье шаг в области частот будет равен

$$b = \frac{1}{NT} = \frac{S}{N} = \frac{F}{N/2}.$$

Напомним, что получающиеся значения  $X(k)$ , где

$$k = \begin{cases} 0, 1, \dots, \frac{N}{2}, & \text{если } N \text{ — четное,} \\ 0, 1, \dots, \frac{N-1}{2}, & \text{если } N \text{ — нечетное.} \end{cases}$$

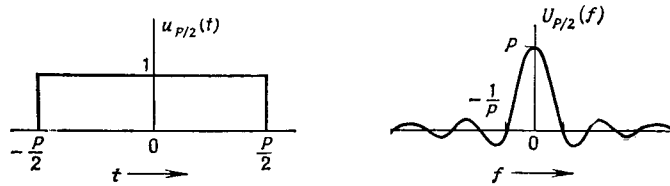
располагаются в частотах  $kb$ , т. е.  $X(k) = X(bk)$ . Первое из значений располагается в частоте  $f=0$  и равно

$$X(0) = \sum_{i=0}^{N-1} x(i).$$

Для последнего значения, расположенного в случае четного  $N$  в частоте  $f=F$ , справедлива формула

$$X\left(\frac{N}{2}\right) = \sum_{i=0}^{N/2-1} [x(2i) - x(2i+1)].$$

**Добавление нулей.** При получении длины последовательности  $N'$  за счет добавления  $\Delta N$  нулей к исходной последовательности



6.8. Прямоугольный импульс и его преобразование.

длины  $N$  шаг в частотной области изменяется следующим образом:

$$b' = \frac{1}{(N + \Delta N)T} = \frac{1}{N'T}.$$

При  $N$ , близких к  $N'$ , т. е. при малых относительно  $N'$  значениях  $\Delta N$ , выполняется приближенное равенство  $b \approx b'$ . При малых значениях  $N'$  разница между  $b$  и  $b'$  может оказаться довольно существенной. В качестве иллюстрации такого изменения шага в области частот рассмотрим преобразование Фурье  $U_{P/2}(f)$  прямоугольного импульса  $u_{P/2}(t)$  с длиной носителя  $P$ . Эти функции изображены на рис. 6.8.

Результат применения быстрого преобразования Фурье без добавления нулей при  $NT=P$  изображен на рис. 6.9.

Если до преобразования добавлено  $N$  нулей (по  $N/2$  нулей перед прямоугольным импульсом и после него), то в результате



6.9. БПФ без добавления нулей.

применения БПФ к последовательности с  $2N$  членами произойдет изменение, показанное на рис. 6.10.

Таким образом, при добавлении нулей происходит интерполяция значений функции  $U_{P/2}(f)$  тем точнее, чем больше нулей

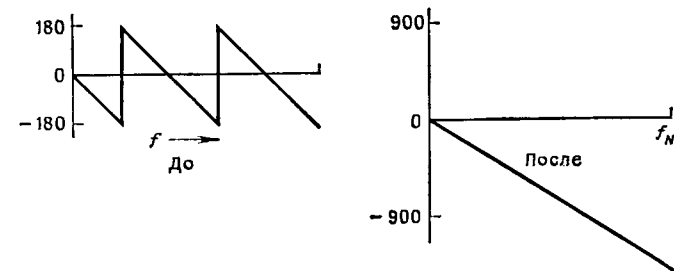


6.10. БПФ с добавлением  $N$  нулей.

добавляется. Такого же рода интерполяция происходит для всякой функции, к которой добавляются нули.

Позднее будет показано, что, хотя шаг в области частот  $b$  изменяется, эффективная разрешающая полоса  $B_e$  остается неизменной вне зависимости от числа добавленных нулей.

**Длина записи.** В анализе цифровых временных рядов число данных, преобразованных в цифровую форму, ограничено воз-

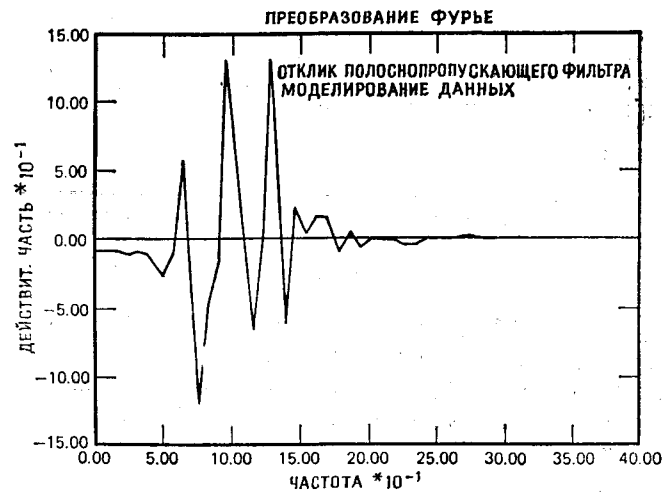


6.11. Продолжение фазы.

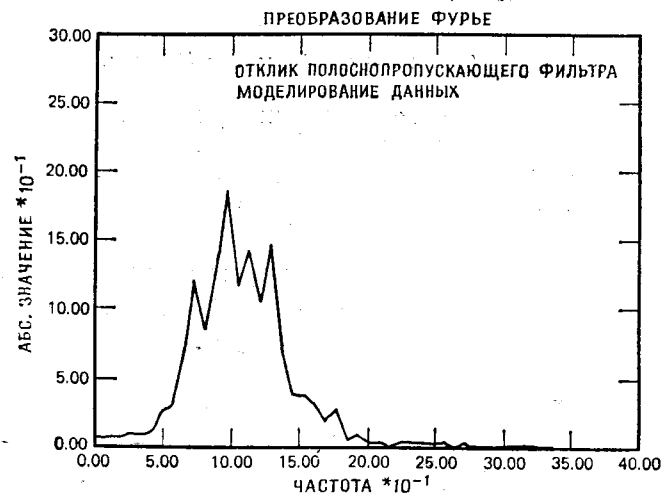
можностями ЭВМ. Длина выборки зависит от скорости преобразования в цифровую форму и ограничена величиной

$$P = NT = \frac{N}{S}.$$





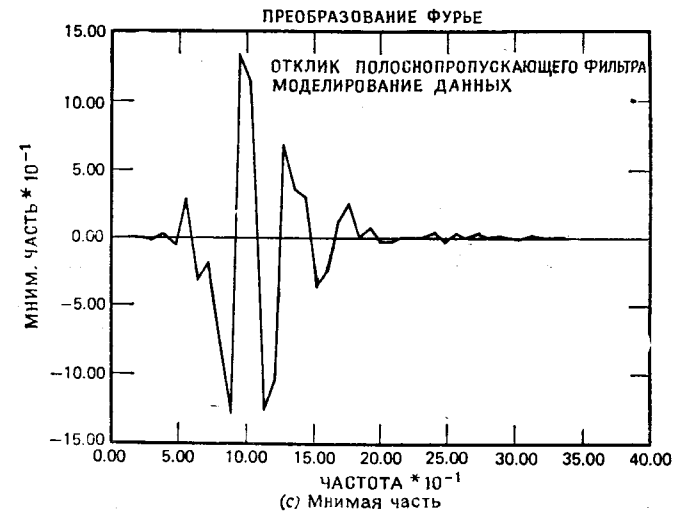
(a) Действительная часть



(b) Абсолютная величина

Рис. 6.12. График преобразования Фурье отклика

Для мини-ЭВМ максимальная величина  $N$  колеблется обычно в пределах от 1024 ( $2^{10}$ ) до 8192 ( $2^{13}$ ). Как правило, в этих случаях выбирается алгоритм быстрого преобразования Фурье для  $N$ , равного степени 2. Для ЭВМ средних и широких возможностей величина  $N$  колеблется от 1024 до 32 768 ( $2^{15}$ ), и для ЭВМ широких возможностей чаще всего выбирается алгоритм БПФ с  $N$ , равным произведению степеней 2, 3, 4 и 5.



(c) Мнимая часть



(d) Фаза

узкополосного фильтра (узкополосного шума).

В обоих случаях нередко требуются длины записей, большие, чем максимально возможная длина выборки  $P_s = N_s T$ . Такие ограничения обходят с помощью двух методов. Во-первых, для получения точного преобразования Фурье на более длинных интервалах выборки используют алгоритм двойной длины или его варианты. Во-вторых, часто вычисляют преобразование, усредненное по нескольким частям выборки.

В первом случае получается лучшее разрешение для спектра, равное  $b = 1/NT$ . Во втором случае разрешение ограничено величиной возможной выборки и равно  $b_s = 1/N_s T$ . В восьмой главе будут рассмотрены вопросы связи шума со степенями свободы и соотношением этих двух методов.

**Продолжение фазы.** Фигурирующее в формуле (6.6) главное значение арктангенса заключено в границах от  $-180^\circ$  до  $+180^\circ$ .

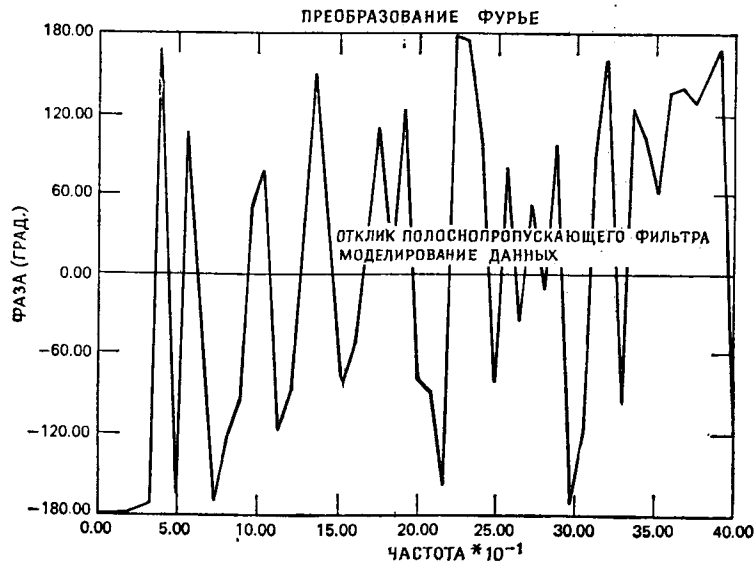


Рис. 6.13. График фазы с рис. 6.12, а без продолжения фазы.

«Продолжением фазы» мы называем процедуру, позволяющую устранять разрывы при переходах через  $180^\circ$ . При этом предполагается, что фазовая функция — «гладкая» функция частоты. Проиллюстрируем на примере большой скачок значения фазы, который может произойти при переходе от  $+(180-\epsilon)^\circ$  к  $-(180-\epsilon)^\circ$ . Величина этого скачка достигает  $\Delta\varphi = (180-\epsilon)^\circ + (180-\epsilon)^\circ = (360-2\epsilon)^\circ$ , хотя сам скачок может быть равным всего  $2\epsilon$ , поскольку  $-(180-\epsilon)^\circ = (180+\epsilon)^\circ$ . Выбор последнего варианта, отвечающего непрерывному случаю, предусмотрен в подпрограмме TTRAN с рис. А.7 (в приложении А).

Эффект продолжения фазы демонстрируется на рис. 6.11. На нем приведены графики линейной фазы (соответствующей случаю простого временного сдвига) до и после продолжения фазы. Второй из них — типичный пример графика, полученного на ЭВМ при продолжении фазы. На рис. 6.12 показаны действительная и мнимая части, коэффициент усиления и продолжен-

ная фаза псевдослучайного белого шума, подвергнутого узкополосной фильтрации. На рис. 6.13 приведен график той же фазы, что и на предыдущем рисунке, но на этот раз без продолжения фазы, т. е. на графике приведены соответствующие величины главного значения арктангенса. На рис. 6.13 хорошо прослеживается нерегулярность графика, полученного без продолжения фазы.

## 6.2. АЛГОРИТМ БЫСТРОГО ПРЕОБРАЗОВАНИЯ ФУРЬЕ

В этом параграфе нами будет дан краткий вывод формул для быстрого преобразования Фурье. При этом мы будем следовать подходу Кули—Тьюки (Кули и Тьюки (1965)), основанному на умножении матриц. Кроме того, будет рассмотрен второй вариант алгоритма БПФ, называемый алгоритмом Сэнда—Тьюки (Джентльмен и Сэнд (1966)). Оба рассматриваемых алгоритма можно назвать каноническими в том смысле, что все остальные методы получения БПФ сводятся в конечном счете к одному из них. По причинам, которые станут яснее позже, вариант Кули—Тьюки принято называть «децимацией по времени», а вариант Сэнда—Тьюки — «децимацией по частотам».

Мы обсудим три программы для ЭВМ. Первые две из них входят в обзор Мэйнерда (1973) под вторым и первым номерами соответственно. (Всего в этот обзор включено 10 программ.) Первая рассмотренная здесь нами программа (Отнес, Натанс и Энноксон (1969)) имеет очень простую запись и предназначена для решения читателем вычислительных задач этой главы. Эта программа составлена для последовательностей, длина которых равна степени 2. Более общий случай последовательностей, длина которых равна произведению степеней 2, 3, 4 и 5, предусмотрен во второй программе (Синглтон (1969)). Третья программа, предназначенная для вычисления преобразований последовательностей с длиной, превышающей внутренний (на сердечниках) объем памяти ЭВМ, также принадлежит Синглтону (1967).

**Матричная запись алгоритма БПФ.** Напомним уже встречавшееся раньше обозначение для комплексной экспоненты

$$W_N = \exp \left[ -j \frac{2\pi}{N} \right] = \cos \frac{2\pi}{N} - j \sin \frac{2\pi}{N}.$$

Заметим, что, используя это обозначение, экспоненты, входящие в формулу преобразования Фурье, можно переписать в виде

$$W_N^{ik} = \exp \left[ -j \frac{2\pi ik}{N} \right] = \cos \left( \frac{2\pi}{N} ik \right) - j \sin \left( \frac{2\pi}{N} ik \right). \quad (6.13)$$

Расположение величин  $W_N$  на комплексной плоскости показано на рис. 6.14. Эти величины в теории функций комплексного

переменного называют *корнями N-й степени из единицы* (в данном случае  $N=16$ ). Из формулы (6.13) можно выводить некоторые довольно сложные тригонометрические соотношения; два из них приведены в упр. 6.11.

Матричный подход для объяснения БПФ был предложен МакКованом (1966). Дальнейшее развитие этот подход получил в работе Тейлхеймера (1969). Мы в качестве пояснения матричного

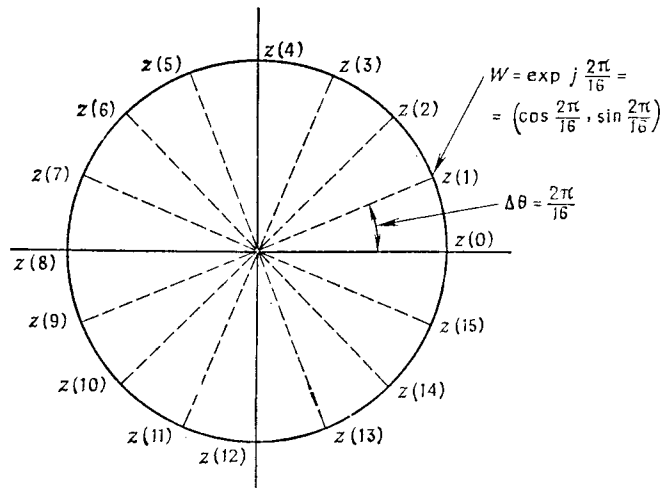


Рис. 6.14. Точки данных, расположенные на единичной окружности комплексной области.

подхода рассмотрим пример, в котором маневр самолета раскладывается на составляющие компоненты — отклонение от курса, наклон относительно поперечной оси и бортовой наклон. В матричном виде поворот плоскости координат на угол  $\theta$  можно записать следующим образом:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}.$$

В трехмерном пространстве всякий поворот можно свести к поворотам в трех координатных двумерных плоскостях (см. рис. 6.15) — повороту в координатной плоскости  $(y, z)$  (бортовой наклон), повороту в координатной плоскости  $(x, z)$  (наклон относительно поперечной оси) и повороту в плоскости  $(x, y)$  (отклонение от курса).

Таким образом, полный поворот описывается произведением трех матриц, соответствующих двумерным преобразованиям:

$$T(\theta_1, \theta_2, \theta_3) = \begin{pmatrix} \cos \theta_3 & -\sin \theta_3 & 0 \\ \sin \theta_3 & \cos \theta_3 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos \theta_2 & 0 & -\sin \theta_2 \\ 0 & 1 & 0 \\ \sin \theta_2 & 0 & \cos \theta_2 \end{pmatrix} \times \\ \times \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \theta_1 & -\sin \theta_1 \\ 0 & \sin \theta_1 & \cos \theta_1 \end{pmatrix}.$$

отклонение от курса      наклон относительно поперечной оси  
бортовой наклон

Поэтому вычисления, необходимые для получения преобразования координат, можно свести к последовательному вычислению

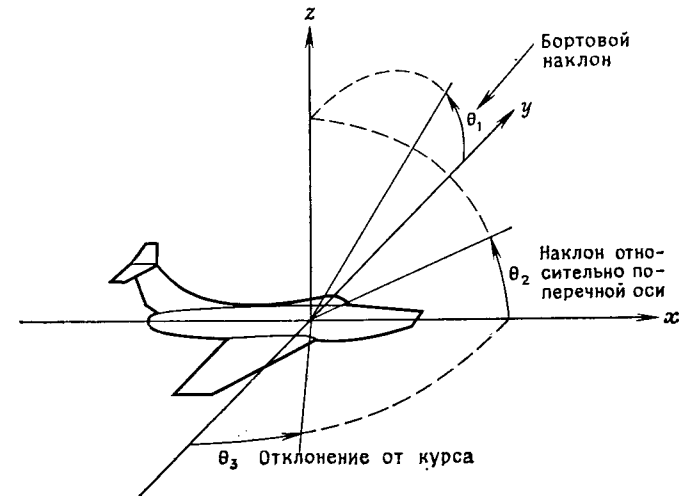


Рис. 6.15. Иллюстрация компонент маневра — отклонения от курса, наклона относительно поперечной оси и бортового наклона.

трех двумерных преобразований. Если такого разбиения вычисления на этапы не проводить, то потребуется  $3^2=9$  операций (для умножения матрицы  $3 \times 3$  на трехмерный вектор). При разбиении число операций равно  $3 \times 2^2=12$ . В четырехмерном случае потребуется  $4^2=16$  и  $4 \times 2^2=16$  операций соответственно (здесь вращение можно свести к четырем вращениям двумерных плоскостей). Уже в пятимерном случае, где прямое вычисление требует  $5^2=25$ , а с разбиением — только  $5 \times 2^2=20$  операций, видна возможность сокращения необходимого для выполнения операций

времени. Это уменьшение становится все более заметным с ростом размерности пространства и показывает преимущество второго метода вычисления. По сути дела, то же самое происходит и при вычислении при помощи алгоритмов БПФ. Дискретное конечное преобразование Фурье  $N$  данных можно представлять как вращения в  $N$ -мерном пространстве. (В линейной алгебре такие вращения называют ортогональными преобразованиями.) В матрице, соответствующей этому преобразованию, показатель величины  $W$  определяется произведением номеров строки и столбца:

$$\begin{bmatrix} X_0 \\ X_1 \\ X_2 \\ \vdots \\ X_{N-1} \end{bmatrix} = \begin{bmatrix} W^{0 \cdot 0} & W^{0 \cdot 1} & W^{0 \cdot 2} & \dots & W^{0 \cdot (N-1)} \\ W^{1 \cdot 0} & W^{1 \cdot 1} & W^{1 \cdot 2} & \dots & W^{1 \cdot (N-1)} \\ W^{2 \cdot 0} & W^{2 \cdot 1} & W^{2 \cdot 2} & \dots & W^{2 \cdot (N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ W^{(N-1) \cdot 0} & W^{(N-1) \cdot 1} & W^{(N-1) \cdot 2} & \dots & W^{(N-1) \cdot (N-1)} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_{N-1} \end{bmatrix},$$

или, в более короткой матричной записи,  $X = Wx$ .

Приведем явный вид матрицы в частном случае  $N = 2^3 = 8$ ,  $p = 3$ :

$$W = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & W^1 & W^2 & W^3 & W^4 & W^5 & W^6 & W^7 \\ 1 & W^2 & W^4 & W^6 & W^8 & W^{10} & W^{12} & W^{14} \\ 1 & W^3 & W^6 & W^9 & W^{12} & W^{15} & W^{18} & W^{21} \\ 1 & W^4 & W^8 & W^{12} & W^{16} & W^{20} & W^{24} & W^{28} \\ 1 & W^5 & W^{10} & W^{15} & W^{20} & W^{25} & W^{30} & W^{35} \\ 1 & W^6 & W^{12} & W^{18} & W^{24} & W^{30} & W^{36} & W^{42} \\ 1 & W^7 & W^{14} & W^{21} & W^{28} & W^{35} & W^{42} & W^{49} \end{bmatrix}.$$

Эту матрицу можно разложить в произведение ( $p+1=4$ ) матриц:

$$W = W_3 W_2 W_1 W_0$$

или, в более общем случае, в произведение

$$W = W_p W_{p-1} \dots W_1 W_0. \quad (6.14)$$

Последняя матрица  $W_p$  — матрица перестановки, переупорядочивающая полученный на последнем шаге вектор перестановкой битов индекса каждого его элемента в обратном порядке.

Правило получения представления  $W$  вида (6.14) сформулировано Тейлхеймером (1969). Для алгоритма Кули — Тьюки, в кото-

ром  $N = 2^p$ , представление вида (6.14) получить довольно просто. Такое представление в частном случае  $N = 2^3$  было получено Мак-Кованом (1966):

$$W_0 = \begin{bmatrix} 1 & 0 & 0 & 0 & W^0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & W^0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & W^0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & W^0 \\ 1 & 0 & 0 & 0 & W^4 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & W^4 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & W^4 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & W^4 \end{bmatrix},$$

$$W_1 = \begin{bmatrix} 1 & 0 & W^0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & W^0 & 0 & 0 & 0 & 0 \\ 1 & 0 & W^4 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & W^4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & W^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & W^2 \\ 0 & 0 & 0 & 0 & 1 & 0 & W^6 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & W^6 \end{bmatrix},$$

$$W_2 = \begin{bmatrix} 1 & W^0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & W^4 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & W^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & W^6 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & W^1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & W^5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & W^3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & W^7 \end{bmatrix}.$$

Обозначение  $W^0$  для единицы оставлено только из соображений симметрии. Учет при программировании особенностей вида этих матриц (нет необходимости производить умножения на нули и единицы) позволяет сократить время работы программы.

**Вариант Кули — Тьюки.** Обозначим символом  $A_l$  вектор данных, полученный после  $l$ -го умножения матриц. Вариант Кули — Тьюки алгоритма преобразования Фурье основан на вычислении

этого вектора данных по формулам

$$\begin{aligned}
 A_l(k_0, k_1, \dots, k_{l-2}, \overset{k_{l-1}}{\downarrow} 0, i_{p-l-1}, \dots, i_0) = \\
 = A_{l-1}(k_0, k_1, \dots, k_{l-2}, \overset{i_{p-l}}{\downarrow} 0, i_{p-l-1}, \dots, i_0) + \\
 + A_{l-1}(k_0, k_1, \dots, k_{l-2}, 1, i_{p-l-1}, \dots, i_0) \times \\
 \times W^{(k_{l-2}2^{l-2} + \dots + k_0) \downarrow 2^{p-l}}, \quad l=1, 2, \dots, p,
 \end{aligned}
 \tag{6.15a}$$

```

SUBROUTINE FFTRAN (SIGN,T,X,NPOM)
C
C   МЕТОД КУЛИ-ТЬЮКИ ВЫЧИСЛЕНИЯ ПРЕОБРАЗОВАНИЯ ФУРЬЕ
C   ВКЛЮЧАЕТ ВЫЧИСЛЕНИЯ СИНУСОВ, КОСИНУСОВ И ПЕРЕУПОРЯДОЧЕНИЕ
C   ДАННЫХ ЧЕРЕЗ ОБРАТНУЮ ПЕРЕСТАНОВКУ БИТОВ ИНДЕКСОВ
C
C   SIGN = УКАЗАТЕЛЬ НАПРАВЛЕНИЯ ПРЕОБРАЗОВАНИЯ ФУРЬЕ
C   = -1. ДЛЯ ПРЯМОГО ПРЕОБРАЗОВАНИЯ
C   = 1. ДЛЯ ОБРАТНОГО ПРЕОБРАЗОВАНИЯ
C   T     = ШАГ ПО ВРЕМЕНИ
C   X     = МАССИВ ДЛЯ ПРЕОБРАЗОВАНИЯ ФУРЬЕ
C   NPOM  = СТЕПЕНЬ 2 (ОБЪЕМ БЛОКА = 2**NPOM)
C
DIMENSION X(1), CS(2), MSK(13)
COMPLEX X,CXCS,HOLD,XA
EQUIVALENCE (CXCS,CS)
NMAX=2**NPOM
ZZ=6.283185306*SIGN/FLOAT(NMAX)
DELTA=T
IF (SIGN) IO,IO,5
DELTA=1./(T*FLOAT(NMAX))
MSK(1)=NMAX/2
DO 15 I=2,NPOM
MSK(I)=MSK(I-1)/2
NN=NMAX
MM=2
C
C   ВНЕШНИЙ ЦИКЛ ДЛЯ СЛОЕВ NPOM
C
DO 45 LAYER=1,NPOM
NN=NN/2
NW=0
DO 40 I=1,MM,2
II=NN*I
C
C   CXCS = СЕХР(2*PI*NW*SIGN/NMAX)
C
W=FLOAT(NW)*ZZ
CS(1)=COS(W)
CS(2)=SIN(W)
C
C   ВЫЧИСЛЕНИЕ ЭЛЕМЕНТОВ ДЛЯ ОБЕИХ ПОЛОВИН КАЖДОГО БЛОКА
C
DO 20 J=1,NN

```

Рис. 6.16а. Программа быстрого преобразования Фурье, часть I. Программа написана Л. Д. Эноксоном, Ховардом Натансом и Р. К. Отнесом.

$$\begin{aligned}
 A_l(k_0, k_1, \dots, k_{l-2}, \overset{k_{l-1}}{\downarrow} 1, i_{p-l-1}, \dots, i_0) = \\
 = A_{l-1}(k_0, k_1, \dots, k_{l-2}, \overset{i_{p-l}}{\downarrow} 0, i_{p-l-1}, \dots, i_0) - \\
 - A_{l-1}(k_0, k_1, \dots, k_{l-2}, 1, i_{p-l-1}, \dots, i_0) \times \\
 \times W^{(k_{l-2}2^{l-2} + \dots + k_0) \downarrow 2^{p-l}}, \quad l=1, 2, \dots, p.
 \end{aligned}
 \tag{6.15b}$$

```

C   II=II+1
C   IJ=IJ-NN
C   XA=CXCS*X(II)
C   X(II)=X(IJ)-XA
C   X(IJ)=X(IJ)+XA
C
C   ВЫЧИСЛЕНИЕ ОБРАТНЫХ АДРЕСОВ
C
DO 25 LOC=2,NPOM
LL=NW-MSK(LOC)
IF (LL) 30,35,25
NW=LL
25 30 NW=MSK(LOC)+NW
GO TO 40
35 40 NW=MSK(LOC+1)
CONTINUE
C
C   ПРОВОДИТСЯ ЗАКЛЮЧИТЕЛЬНОЕ ПЕРЕУПОРЯДОЧЕНИЕ
C   А ТАКЖЕ УМНОЖЕНИЕ НА DELTA
C
45 MM=MM*2
NW=0
DO 80 I=1,NMAX
NW1=NW+1
HOLD=X(NW1)
IF (NW1-II) 60,55,50
X(NW1)=X(II)*DELTA
X(II)=HOLD*DELTA
C
C   ВЫЧИСЛЕНИЕ ОБРАТНОГО АДРЕСА
C
60 DO 65 LOC=1,NPOM
LL=NW-MSK(LOC)
IF (LL) 70,75,65
NW=LL
70 NW=MSK(LOC)+NW
GO TO 80
75 80 NW=MSK(LOC+1)
CONTINUE
RETURN
END

```

Рис. 6.16б. Программа быстрого преобразования Фурье, часть II.

Значительно проще эти соотношения выглядят в программе на Фортране (рис. 6.16). Им соответствуют предложения 48 и 49 во внутреннем цикле, имеющие вид

$$\begin{aligned} X(IJ) &= X(IJ) + X(II) * CXCS \\ X(II) &= X(IJ) - X(II) * CXCS \end{aligned} \quad (6.16)$$

где

$$\begin{aligned} CXCS &= W^{(k_{l-2}2^{l-2} + \dots + k_0)2^{l-1}} \\ IJ &= (k_0, k_1, \dots, k_{l-2}, 0, i_{p-l-1}, \dots, i_0) \\ II &= (k_0, k_1, \dots, k_{l-2}, 1, i_{p-l-1}, \dots, i_0) \end{aligned}$$

Отметим, что (6.16) — формулы для двухточечного преобразования Фурье (см. упр. 6.5), в которых дополнительно производится умножение на экспоненту. Читателю следует обратить внимание на следующие особенности алгоритма.

1. Внутренний цикл, в котором вычисляется двухточечное преобразование Фурье с дополнительным умножением на экспоненту, соответствует формулам (6.10) для преобразования двойной длины.

2. Исходные данные — массив  $A_0$  и временной индекс — раскладываются на двоичные компоненты. На каждом шаге рекурсии вычисляется преобразование для одной компоненты.

3. Рекурсия проводится, начиная со старшего бита временного индекса ( $i_{p-1}$ ), который преобразуется в младший бит частотного индекса ( $k_0$ ).

4. В результате перестановки компонент индекса в обратном порядке возникает следующее соотношение для элементов массива  $X$ :

$$A_i(k_{p-1}, k_{p-2}, \dots, k_0) = X(k_0, k_1, \dots, k_{p-1}).$$

Следовательно, для того чтобы частоты в распечатке или графике имели естественный порядок, следует переупорядочить элементы полученного массива в соответствии с обратной перестановкой битов индексов.

В программе БПФ с рис. 6.16 для обратной перестановки битов индексов предусмотрена такая процедура. Все биты предыдущего индекса вплоть до первого, содержащего 0, полагаются равными нулю. Заменяя в этом первом бите с нулем нуль на единицу, получают новый индекс. Например, при  $p=3$  начинают с  $000_2 = 0_{10}$  и получают  $100_2 = 4_{10}$  (т. е. индекс, порядок битов в котором противоположен порядку битов индекса  $001_2 = 1_{10}$ ). Следующий индекс получается равным  $010_2 = 2_{10}$  и т. д. (Нам

сообщали, что для этой цели существуют машинные команды, правда, не на всех ЭВМ.) Как в варианте Кули—Тьюки, так и в варианте Сэнда—Тьюки предусмотрена обратная перестановка битов и переупорядочение (программно) конечных результатов в естественную очередность. Отметим, что некоторыми уловками, например предварительной перестановкой исходных данных, можно достичь того же результата.

Процедура перестановки встречается в программе два раза: в строках 55—61 и в строках 79—85 (рис. 6.16). Оператор IF в строке 57 предназначен для проверки очередного бита. Первый раз процедура используется для вычисления аргументов синусов и косинусов, а во второй — для окончательного упорядочения данных.

Сделаем еще два замечания относительно программы БПФ с рис. 6.16. Вычисление синусов и косинусов комплексных экспонент проводится через библиотеку подпрограмм Фортрана (строки 39 и 40). Далее в вычислениях предусмотрены шаги  $T$  и  $b$ , соответствующие  $dt$  и  $df$  в интеграле Фурье. Эти величины из формул (6.1) и (6.7) входят в произведения на строках 73 и 74.

Этим алгоритмом следует пользоваться во всех упражнениях с четными номерами в конце главы.

В другом варианте алгоритма БПФ — Сэнда и Тьюки — рекурсивные соотношения выглядят следующим образом (см., например, книгу Отнеса и Эноксона (1972), в которой приведены соответствующие выкладки):

$$\begin{aligned} & \hat{A}_l(k_0, k_1, \dots, k_{l-2}, \underset{\substack{\downarrow \\ k_{l-1}}}{0}, i_{p-l-1}, \dots, i_0) = \\ & = [\hat{A}_{l-1}(k_0, k_1, \dots, k_{l-2}, \underset{\substack{\downarrow \\ i_{p-l}}}{0}, i_{p-l-1}, \dots, i_0) + \\ & + \hat{A}_{l-1}(k_0, k_1, \dots, k_{l-2}, 1, i_{p-l-1}, \dots, i_0)] \times \\ & \times W^{(k_{l-1}2^{l-1} + \dots + i_0)2^{l-1}}, \\ & \hat{A}_l(k_0, k_1, \dots, k_{l-2}, \underset{\substack{\downarrow \\ k_{l-1}}}{1}, i_{p-l-1}, \dots, i_0) = \\ & = [\hat{A}_{l-1}(k_0, k_1, \dots, k_{l-2}, \underset{\substack{\downarrow \\ i_{p-l}}}{0}, i_{p-l-1}, \dots, i_0) - \end{aligned} \quad (6.17a)$$

$$-A_{l-1}(k_0, k_1, \dots, k_{l-2}, 1, i_{p-l-1}, \dots, i_0) \times$$

$$\begin{matrix} k_{l-1} \\ \downarrow \\ \times W^{(i_{p-l-1}2^{p-l-1} + \dots + i_0) 2^{l-1}} \end{matrix} \quad (6.17b)$$

Внутренний цикл в Фортран-программе выглядит несколько иначе, чем у Кули и Тьюки:

$$X(I) = X(I) + X(II)$$

$$X(II) = (X(I) - X(II)) * CXCS$$

В варианте Сэнда — Тьюки комплексная экспонента имеет вид

$$CXCS = W^{(i_{p-l-1}2^{p-l-1} + \dots + i_0) 2^{l-1}}$$

И в этом случае двухточечное преобразование Фурье дополняется умножением на комплексную экспоненту.

В принципе оба варианта требуют одинакового количества вычислений. Однако на практике вариант Сэнда — Тьюки (С — Т) часто оказывается удобнее для перестановки данных в естественный порядок из-за показателя степени у  $W$ . Что же касается варианта Кули — Тьюки (К — Т), то он может оказаться проще для предварительно переставленных данных (Отнес и Эноксон (1972), с. 154). Как видно из формул (6.15) и (6.17), и в первом, и во втором варианте индексы, для которых вычисляются значения  $A_i$  и  $\hat{A}_i$ , можно разбить на две части:

$$\underbrace{k_0 2^{p-1} + k_1 2^{p-2} + \dots + k_{l-2} 2^{p-l-3} + k_{l-1} 2^{p-l-2}}_{1\text{-я часть}} + \underbrace{i_{p-l-1} 2^{p-l-1} + \dots + i_0}_{2\text{-я часть}} \quad (6.18)$$

Отметим, что в варианте К — Т показатель экспоненты  $(k_{l-1} 2^{l-1} + \dots + k_0) i_p 2^{p-l}$  есть побитно переставленная в обратном порядке 1-я часть суммы (6.18), умноженная на степень 2. В варианте С — Т показатель экспоненты есть 2-я часть суммы (6.18), умноженная на степень 2. Никаких перестановок битов в этом случае нет. Поэтому в некотором смысле легче получить необходимый для вычисления аргумента комплексной экспоненты итог, пользуясь вариантом С — Т, а не К — Т.

Часто вариант К — Т называют «децимацией по времени». Из диаграммы на рис. 6.17 (ее называют графом потока сигналов) видно, что в алгоритме начинается вычисление величин комплексных экспонент для данных, разделенных  $N/2$  позициями. Следовательно, происходит децимация данных с индексом  $N/2$ . Аналогичная диаграмма потока сигналов на рис. 6.18 для варианта С — Т объясняет, почему этот вариант называют «децимацией

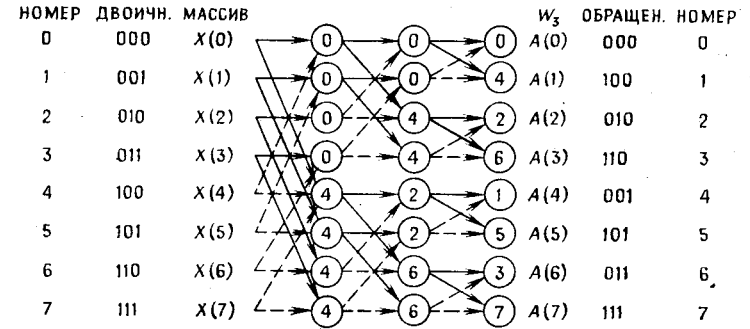


Рис. 6.17. Диаграмма алгоритма Кули — Тьюки,  $N=8, p=3$  (Мак-Кован (1966)).

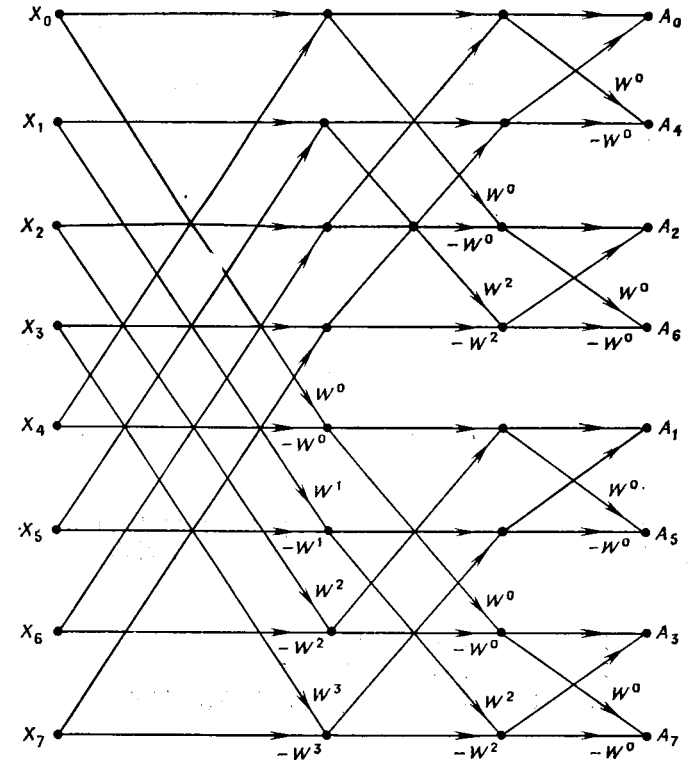


Рис. 6.18. Граф потока сигналов Сэнда — Тьюки (из статьи What Is the Fast Fourier Transform, IEEE Transactions on Audio Electroacoustics, Vol. AU-15, No. 2, pp. 45—55, 1969).

по частотам». Как видно из рисунка, на последней стадии алгоритма величины комплексных экспонент также разделяются  $N/2$  позициями.

**Пояснения к алгоритму.** Предложенная на рис. 6.16 Фортран-программа — наиболее простой вариант для алгоритма К — Т, предназначенный для обработки исходных данных, расположенных в их естественном порядке, и для переупорядочения получающихся результатов в порядке, соответствующем исходному. Эта программа позволяет получать и прямое, и обратное преобразование Фурье последовательностей длины  $N = 2^p$  с комплексными членами. Помимо перестановки получающихся результатов и выбора направления преобразования в программе предусмотрено задание величин  $T$  и  $p$ . Максимальное значение, которое может принимать  $p$ , выбрано в этой программе равным 13.

Массивы исходных данных определяют предложения

```
15      DIMENSION X(1), CS(2), MSK(13)
16      COMPLEX X, CXCS, HOLD, XA
17      EQUIVALENCE (CXCS, CS)
```

В предложении

```
18      NMAX=2**NPOW
```

вычисляется значение  $N = 2^p$ , где  $p \leq 13$ . Затем вычисляется величина  $ZZ = 2\pi/N$ , необходимая для определения аргументов комплексных экспонент:

```
19      ZZ=6.283185306*SIGN/FLOAT(NMAX)
```

Для прямого преобразования Фурье выбирается значение  $DELTA = T$ , а для обратного —  $DELTA = b = 1/NT$ :

```
20      DELTA=T
21      IF (SIGN) 10,10.5
22      5  DELTA=1./(T*FLOAT(NMAX))
```

После этого формируется последовательность двоичных векторов, используемых в переупорядочении:

```
23      10  MSK(1)=NMAX/2
24      DO 15 I=2,NPOW
25      15  MSK(I)=MSK(I-1)/2
```

В результате получается массив

$$\begin{aligned} MSK(1) &= 10 \dots 0 \\ MSK(2) &= 010 \dots 0 \\ MSK(p) &= \underbrace{000 \dots 01}_{p \text{ битов}} \end{aligned}$$

Внутренний цикл вычисления БПФ составляют команды

```
44      DO 20 J=1,NN
45      II=II+1
46      IJ=II-NN
47      XA=CXCS*X(II)
48      X(II)=X(IJ)-XA
49      20  X(IJ)=X(IJ)+XA
```

Как было указано выше, CXCS есть комплексная экспонента

$$W^{(k_{l-2} 2^{l-2} + \dots + k_0) 2^{p-l}}$$

В строках 48 и 49 производится вычисление двухточечного преобразования Фурье, в которое дополнительно введен множитель поворота, т. е. экспонента. Отметим, что адреса данных разделены расстоянием, равным множителю NN (биты  $i_{p-l}$  или  $k_{l-1}$ ); этот множитель определяет, кроме того, число проходов внутреннего цикла. Предложение  $II = II + 1$ , включенное в этот цикл, проводит значения индекса Фортран-программы от 1 до  $N$ , что отвечает пробеганию индексов нашего уравнения от 0 до  $N-1$ .

В последовательности предложений, которая сейчас будет приведена, предусмотрены установка начального значения аргумента комплексной экспоненты ( $NW = 0$ ), вычисление комплексной экспоненты, определение необходимого для аргумента комплексной экспоненты адреса с обращенным порядком битов и проверка для выхода из цикла по большей значащей части индекса данных. Поэтому индексация  $I$  отвечает первой части (6.18),



в то время как индекс  $J$  внутреннего цикла — второй части (6.18):

```

31      NN=NN/2
32      NW=0
33      DO 40 I=1,MM,2
34      II=NN*I
35      C
36      C          CXCS=CEXP(2*PI*NW*SIGN/NMAX)
37      C
38      W=FLOAT(NW)*ZZ
39      CS(1)=COS(W)
40      CS(2)=SIN(W)
:
50      C
51      C
52      C
53      C      ВЫЧИСЛЕНИЕ ОБРАТНОГО АДРЕСА
54      C
55      DO 25 LOC=2,NPOW
56      LL=NW-MSK(LOC)
57      IF (LL) 30,35,25
58      25  NW=LL
59      30  NW=MSK(LOC)+NW
60      GO TO 40
61      35  NW=MSK(LOC+1)
62      40  CONTINUE

```

Как следует из формулы (6.15), аргумент для вычисления синусов и косинусов получается умножением на  $2^{p-l}$ , т. е. смещением расположения битов на  $p-l$  позиций влево. Мы избегаем непосредственного умножения, получая нужный индекс из большей значащей части битов, и уменьшаем тем самым объем работы.

Новые адреса с обратным порядком следования битов  $NW$  определяются по правилу, которое уже выше упоминалось. Напомним, что по этому правилу происходит проверка старших битов предыдущего адреса с обратным порядком битов  $NW$ . Для этой цели служит массив  $MSK$ .

Заметим, что переменная  $NN$ , определяющая число прохождений внутреннего цикла, каждый раз уменьшается вдвое (см. предложение 31). Следовательно, в первый раз цикл проходит  $N/2$  раз. Последний раз этот внутренний цикл проходит только один раз.

Теперь обратимся к последовательности предложений, в которой задаются пределы изменения индексов:  $NN$  для внутреннего цикла,  $MM$  для среднего цикла. Индекс внешнего цикла  $LL$ , который соответствует нижнему индексу  $l$  в формуле (6.15), пробегает значения от 1 до  $p$ :

```

26      NN=MAX
27      MM=2
28      C
29      C          ВНЕШНИЙ ЦИКЛ ДЛЯ СЛОЕВ NPOW
30      DO 45 LAYER-1,NPOW
:
63      45  MM=MM*2

```

Отсюда видно, что число  $MM$ , определяющее число проходов среднего цикла, сначала равно 2, а затем каждый раз удваивается. Если учесть, что вычисления в среднем цикле проводятся с шагом 2, то для него по формуле  $\frac{MM}{2} \times \frac{NN}{2}$  можно определить число выполнений операций в строках 47—49. Для внешнего цикла это число будет равно (после  $p$  проходов)

$$\frac{2}{2} \frac{N}{2} + \frac{4}{2} \frac{N/2}{2} + \dots + \frac{N}{2} 1 = p \frac{N}{2}.$$

Следовательно, в центральной части БПФ производится примерно  $Np/2$  комплексных умножений, комплексных сложений и комплексных вычитаний.

На последнем этапе работы проводится переупорядочение данных, соответствующее адресации с обратным порядком битов, и умножение результатов на  $T$  или  $b$ . Этим операциям отвечают строки программы с 68 по 86:

```

64      C
65      C          ПРОВОДИТСЯ ЗАКЛЮЧИТЕЛЬНОЕ УПОРЯДОЧЕНИЕ
66      C          А ТАКЖЕ УМНОЖЕНИЕ НА DELTA
67      C

```

```

68      NW=0
69      DO 80 I=1,NMAX
70      NW1=NW+1
71      HOLD=X(NW1)
72      IF(NW1-1) 60,55,50
73      50  X(NW1)=X(I)*DELTA
74      55  X(I)=HOLD*DELTA
75      C
76      C
77      C          ВЫЧИСЛЕНИЕ ОБРАТНОГО АДРЕСА
78      C
79      60  DO 65 LOC=1,NPOW
80      LL=NW-MSK(LOC)
81      IF (LL) 70,75,65
82      65  NW=LL
83      70  NW=MSK(LOC)+NW
84      GO TO 80
85      75  NW=MSK(LOC+1)
86      80  CONTINUE

```

Вся эта небольшая программа довольно проста. Ее работу можно несколько убыстрить, предварительно переупорядочив массив исходных данных. Это позволяет избежать тех переадресаций, которые необходимы при получении аргументов для синусов и косинусов. Скорость работы программы можно увеличить также рекурсивным получением синусов и косинусов. Однако, добровольно жертвуя памятью для увеличения скорости за счет выбора значений синусов и косинусов из таблицы, приходится отказываться от предварительного переупорядочения данных, поскольку оно не приносит желаемой выгоды из-за того, что соответствующее переупорядочение можно провести и непосредственно в таблицах синусов и косинусов.

В программу можно внести и другие усовершенствования. Например, при первом прохождении среднего цикла (т. е. при  $NW=0$ ) нет необходимости умножений на  $CXCS$ , что позволяет оставить только сложения и вычитания (см. упр. 6.13). Внимательный учет такого рода особенностей алгоритма дает бесчисленное множество различных возможностей повышения эффективности программы БПФ. Количество таких усовершенствований диктуется лишь объемом и сложностью соответствующих изменений в программе. Никаких общих рекомендаций на этот счет привести нельзя, поскольку в каждом частном случае может оказаться наилучшим свой вариант.

**Алгоритм Синглтона для произведения степеней 2, 3, 4 и 5.** Исследованию программ БПФ для последовательностей с длиной, равной произведению степеней произвольных целых чисел,

посвящена работа Синглтона (1969). Наилучшей из этих программ оказалась программа для произведений степеней 2, 3, 4 и 5. Наиболее удобный и эффективный в этом случае предложен Мэйнэрдом (1973). Основным достоинством его программы — это сложность, поскольку программа в себя почти 400 предложений Фортрана. Мы не будем приводить текста программы, которую читатель может взять из книги Мэйнэрда, и остановимся лишь на особенностях алгоритма. В табл. 6.2 приведены числа от 2 до 5000, которые можно представить в виде произведения степеней 2, 3 и 5. Эти числа гораздо плотнее, чем степени 2, заполняют выбранный промежуток. Синглтон определил прямое преобразование формулой

$$\alpha_k = \sum_{i=0}^{N-1} x_i \exp \left[ \frac{j2\pi ik}{N} \right].$$

Его определение соответствует нашему определению обратного преобразования, правда, без множителя  $b$ . При использовании его программы это расхождение нетрудно ликвидировать.

БПФ со смешанной базой  $N=N_1 N_2 \dots N_p$  основано на представлении преобразования Фурье в матричном виде

$$\mathbf{X} = \mathbf{W}\mathbf{x}, \quad (6.19)$$

где матрицу  $\mathbf{W}$  можно разложить в произведение

$$\mathbf{W} = \mathbf{P}\mathbf{W}_p \mathbf{W}_{p-1} \dots \mathbf{W}_2 \mathbf{W}_1,$$

в котором матрицы  $\mathbf{W}_i$  соответствуют сомножителям  $N_i$ , а  $\mathbf{P}$  — матрица перестановки. Каждую из матриц  $\mathbf{W}_i$  можно разбить в свою очередь на  $N/N_i$  подматриц размерности  $N_i$ . Далее, матрицы  $\mathbf{W}_i$  можно представить в виде произведения

$$\mathbf{W}_i = \mathbf{R}_i \mathbf{T}_i,$$

где  $\mathbf{R}_i$  — диагональная матрица вращения, а матрица  $\mathbf{T}_i$  распадается на  $N/N_i$  одинаковых квадратных матриц.

Сомножители в алгоритме Синглтона располагаются симметрично. Например, для  $N=270$  множители располагаются в таком порядке:  $3 \times 2 \times 3 \times 5 \times 3$ . После этого выбирается перестановка, соответствующая такому расположению сомножителей алгоритма.

**Алгоритм Синглтона, использующий внешние запоминающие устройства.** В другой известной работе Синглтона (1967) рассмотрено несколько способов вычисления БПФ последовательностей, длина которых не позволяет поместить все данные в быстродействующее запоминающее устройство на магнитных сердечниках. Мы рассмотрим алгоритм менее общего вида, чем

Таблица 6.2

Целые числа от 1 до 5000, представимые в виде  $N = 2^i 3^j 5^k$ 

$N$	$i$	$j$	$k$	$N$	$i$	$j$	$k$	$N$	$i$	$j$	$k$
2	1	0	0	243	0	5	0	1458	1	6	0
3	0	1	0	250	1	0	3	1500	2	1	3
4	2	0	0	256	8	0	0	1536	9	1	0
5	0	0	1	270	1	3	1	1600	6	0	2
6	1	1	0	288	5	2	0	1620	2	4	1
8	3	0	0	300	2	1	2	1728	6	3	0
9	0	2	0	320	6	0	1	1800	3	2	2
10	1	0	1	324	2	4	0	1875	0	1	4
12	2	1	0	360	3	2	1	1980	7	1	1
15	0	1	1	375	0	1	3	1944	3	5	0
16	4	0	0	384	7	1	0	2000	4	0	3
18	1	2	0	400	4	0	2	2025	0	4	2
20	2	0	1	405	0	4	1	2048	11	0	0
24	3	1	0	432	4	3	0	2160	4	3	1
25	0	0	2	450	1	2	2	2187	0	7	0
27	0	3	0	480	5	1	1	2250	1	2	3
30	1	1	1	486	1	5	0	2304	8	2	0
32	5	0	0	500	2	0	3	2400	5	1	2
36	2	2	0	512	9	0	0	2430	1	5	1
40	3	0	1	540	2	3	1	2500	2	0	4
45	0	2	1	576	6	2	0	2560	9	0	1
48	4	1	0	600	3	1	2	2592	5	4	0
50	1	0	2	625	0	0	4	2700	2	3	2
54	1	3	0	640	7	0	1	2880	6	2	1
60	2	1	1	648	3	4	0	2916	2	6	0
64	6	0	0	675	0	3	2	3000	3	1	3
72	3	2	0	720	4	2	1	3072	10	1	0
75	0	1	2	729	0	6	0	3125	0	0	5
80	4	0	1	750	1	1	3	3200	7	0	2
81	0	4	0	768	8	1	0	3240	3	4	1
90	1	2	1	800	5	0	2	3375	0	3	3
96	5	1	0	810	1	4	1	3456	7	3	0
100	2	0	2	864	5	3	0	3600	4	2	2
108	2	3	0	900	2	2	2	3645	0	6	1
120	3	1	1	960	6	1	1	3750	1	1	4
125	0	0	3	972	2	5	0	3840	8	1	1
128	7	0	0	1000	3	0	3	3888	4	5	0
135	0	3	1	1024	10	0	0	4000	5	0	3
144	4	2	0	1080	3	3	1	4050	1	4	2
150	1	1	2	1125	0	2	3	4096	12	0	0
160	5	0	1	1152	7	2	0	4320	5	3	1
162	1	4	0	1200	4	1	2	4374	1	7	0
180	2	2	1	1215	0	5	1	4500	2	2	3
192	6	1	0	1250	1	0	4	4608	9	2	0
200	3	0	2	1280	8	0	1	4800	6	1	2
216	3	3	0	1296	4	4	0	4860	2	5	1
225	0	2	2	1350	1	3	2	5000	3	0	4
240	4	1	1	1440	5	2	1				

тот, который был предложен Синглтоном, и ограничимся следующими случаями.

Во-первых, будем считать, что алгоритм предназначен для последовательностей с длиной, равной степени 2.

Во-вторых, предположим, что в нашем распоряжении имеются либо четыре запоминающих устройства на ленте, либо один или более дисков, которые можно использовать как четыре устройства ввода-вывода.

Говоря о четырех запоминающих устройствах на ленте, которые встречаются в описании алгоритма, мы должны помнить, что для тех же целей можно воспользоваться одним запоминающим устройством на диске, в котором имеется 4 соответствующих области запоминания.

Допустим, что ряд, для которого нужно получить преобразование Фурье, содержит  $N = 2^p$  точек (комплексных величин) и что данные записаны блоками по  $2^r$  комплексных значений в каждом блоке, причем  $2^r$  значительно меньше  $2^p$ . Оказывается, для того чтобы получить преобразование этого ряда и переставить в нужном порядке данные, требуется  $2p - r - 2$  проходов. Эти проходы распадаются на два этапа: в первом производится  $p - r$  вычислительных проходов и  $p - r - 2$  перестановочных проходов; во втором этапе производится  $r$  вычислительных проходов.

Например, если  $N = 2^{15}$  и  $2^r = 256 = 2^8$  (на один блок данных приходится, следовательно, по 512 слов для действительных чисел), то первый этап сведется к 7 вычислительным и 5 перестановочным проходам, а второй — к 8 вычислительным.

Вычисление начинается с записи первых  $N/2$  комплексных величин, для которых нужно получить преобразование, на первую ленту и остальных  $N/2$  комплексных величин на вторую ленту. Все четыре ленты перематываются к началу.

На первом этапе два блока данных, по одному с каждой из двух первых лент, записываются в память ЭВМ на сердечниках как отдельные массивы. Из каждого массива последовательно берутся комплексные величины, преобразуются и затем помещаются на место исходных величин. Преобразование проводится по формулам

$$Y(k) = X(k) + X(k + N/2) \exp \left[ -j\pi \left( \frac{2(k/(p-q))}{2^{q-1}} \right) \right],$$

$$Y\left(k + \frac{N}{2}\right) = X(k) - X(k + N/2) \exp \left[ -j\pi \left( \frac{2(k/(p-q))}{2^{q-1}} \right) \right],$$

где  $X(k)$  есть  $k$ -е значение на ленте 1,  $k = 0, \dots, N/2$ ;  $X(k + N/2)$  есть  $k$ -е значение на ленте 2,  $k = 0, \dots, N/2$ ;  $Y(k)$  — замена для  $X(k)$ , а  $Y(k + N/2)$  — замена для  $X(k + N/2)$ ;  $q$  — номер вычислительного прохода,  $q = 1, 2, \dots, m$ . Вычисления с величинами

в круглых скобках должны проводиться в целочисленной арифметике.

Такие же операции проводятся на всех  $m$  вычислительных проходах.

После обработки  $2^r$  величин каждого блока результаты записываются на ленты 3 и 4. Те результаты, которые получены из данных, находящихся на первой ленте, записываются на третью, а те, которые получены из данных со второй ленты, — на четвертую.

Вслед за первыми блоками с первой и второй лент считываются и обрабатываются вторые. Так продолжается до тех пор, пока вся информация с 1-й и 2-й лент не будет считана, обработана и записана на ленты 3 и 4. Все эти операции составляют первый вычислительный проход, входящий в первый этап получения преобразования.

На перестановочных проходах 1-го этапа проводятся обратная перемотка всех четырех лент и перестановка данных. При этом начинают с третьей ленты: блоки считываются с первой ее половины и записываются на 1-ю ленту, со второй половины блоки записываются на 2-ю ленту. После того как вся третья лента переписана, та же процедура повторяется для четвертой ленты.

Так производится перестановка при  $q=1$ . На следующих перестановочных проходах, по одному на каждый вычислительный проход, число блоков, последовательно записываемых с лент 3 и 4, равно  $2^{p-r-1-q}$ . Следовательно, при  $q=1$  число блоков, последовательно выбираемых и записываемых на первые две ленты, равно  $2^{p-r-2}$ , при  $q=2$  число таких блоков равно  $2^{p-r-3}$  и т. д. При  $q=p-r-2$  последовательно берется по два блока, а при  $q=p-r-1$  и  $q=p-r$  никаких перестановок не производится. Например, для  $N=2^{15}$ ,  $r=8$  и  $p=15$  число последовательно записываемых блоков при  $q=1$  равно 32, при  $q=2$  равно 16 и т. д. При  $q=5$  последовательно записываются пары блоков, а при  $q=6$  и  $q=7$  перестановочных проходов нет. Таким образом, для двух последних вычислительных проходов первого этапа перестановочные проходы отсутствуют.

Вообще говоря, при  $q=p-r-1$  (в нашем примере при  $q=6$ ) должен быть один перестановочный проход. Но соответствующая перестановка учитывается на последнем вычислительном этапе, т. е. при  $q=p-r$ , следующим образом: при обработке берутся не пары блоков по одному с каждой ленты, как это было на предыдущих вычислительных проходах, а пары блоков с одной ленты (сначала с первой, а затем, после того как все данные с нее будут обработаны, со второй). Обработка данных этих блоков проводится так же, как и прежде. Такой ход вычислений используется только один раз, при  $q=p-r$ , т. е. на пос-

леднем вычислительном проходе 1-го этапа вычисления преобразования.

На втором этапе, включающем в себя  $r$  проходов, вычисления и перестановки объединены вместе. На каждом из этих проходов с данными поступают следующим образом. На вход подаются блски по одному, начиная с первого блока первой ленты входа. После окончания первой ленты на вход подаются блоки данных со второй ленты, начиная с последнего блока. В отличие от первого этапа теперь требуется 3 массива данных объемом  $2^r$  (комплексных) — на один больше, чем раньше. Один из этих массивов предназначен для записи данных обрабатываемого блока. В двух других помещаются результаты для вывода на ленты.

Перестановка на  $q$ -м вычислительном шаге, где теперь  $p-r-1 \leq q < p$ , производится следующим образом. Для обработки берутся пары комплексных значений блока, разделенные  $2^{p-q}$  величинами. В нашем примере при  $q=8$  на первой стадии вычислений расстояние между парой обрабатываемых величин равно  $2^{15-8} = 2^7 = 128$ , поэтому вычисления на втором этапе начинаются для 1-й и 129-й величин, затем обрабатывается пара из 2-й и 130-й величин и т. д. Формулы для вычислений остаются теми же, что и на первом этапе. Результаты последовательно запоминаются в выходные массивы: величина  $Y(k)$  в первый массив, а  $Y(k+N/2)$  — во второй. После того как оба массива окажутся заполненными, они записываются на первую и вторую ленты выхода. Поскольку данные для обработки берутся из входного массива парами, для двух выходных массивов требуется два входных массива.

После каждого такого вычислительного прохода 2-го этапа ленты перематываются и меняются ролями: вход становится выходом и наоборот.

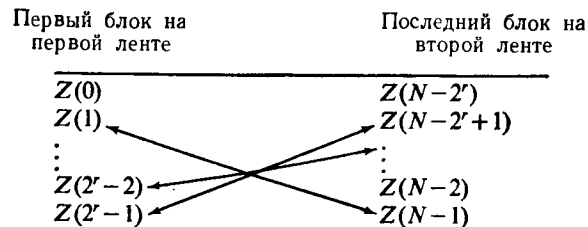
На последнем  $r$ -м проходе 2-го этапа на выходе получается искомое преобразование Фурье данных. Первая половина преобразования записана на первой ленте выхода, а вторая — на второй ленте.

Если исходная функция была действительной, т. е. все мнимые части были равны 0, то в результате получится нужное преобразование, и никаких дополнительных действий проводить не нужно.

Если же проводится спаренное преобразование Фурье, т. е. проводится оновременное определение преобразований двух действительных функций, скажем  $x(i)$  и  $y(i)$ , которые выступают в качестве действительной и мнимой частей исходного ряда данных соответственно, то полученный результат еще необходимо разъединить. При этом используется алгоритм, основу которого составляют формулы (6.8) и (6.9).

Для этой цели сначала берутся первый блок результатов с первой ленты и последний блок результатов со второй ленты, затем обрабатываются второй и предпоследний блоки с первой и второй лент соответственно и т. д. Иначе говоря, необходимо чтение с конца второй ленты и возвращение ленты к началу предыдущего блока.

Следует учесть, что блоки результатов при такой процедуре накладываются не совсем точно, и одно комплексное значение должно запоминаться для следующей пары блоков. Как это получается, нетрудно понять, взглянув на следующую диаграмму:



### 6.3. ПРИМЕРЫ

В этом параграфе мы рассмотрим в качестве иллюстраций применения преобразования Фурье несколько примеров; в частности, вычисления преобразований: суммы синусов и косинусов, стационарного случайного процесса, переходного сигнала и временного ряда вибрационных данных, полученных с помощью акселерометра, как мы надеемся, дадут читателю возможность «почувствовать» преобразование Фурье и представить его вид в наиболее характерных ситуациях. Рассмотренные примеры помогут понять, как преобразование Фурье используется для интерполяции числовых функций, как проявляется эффект добавления к данным нулей и чем отличается рассмотренный выше случай широкополосного случайного шума от случая узкополосного шума; эти примеры помогут также оценить значение преобразования Фурье для практических задач и покажут те искажения, которые происходят из типичных ошибок в данных.

**Суммы синусов и косинусов.** Напомним, что преобразование Фурье синусоиды частоты  $f_0$  — пара дельта-функций, расположен-

ных в точках  $\pm f_0$  (см. гл. 1). В силу линейности для суммы синусоид преобразованием Фурье будет служить сумма соответствующих пар дельта-функций. Далее, напомним, что конечное дискретное преобразование Фурье суммы синусоид будет суммой не пар дельта-функций, а суммой заменяющих их функций  $(\sin x)/x$ . Рассмотрим сумму трех дискретных синусоид:

$$x(i) = \sum_{k=1}^3 a_k \cos(2\pi T f_k i + \varphi_k), \quad (6.20)$$

где  $i = 0, 1, \dots, N-1$ . Эти синусоиды имеют следующие частоты и фазы:

$k$	$a_k$	$f_k$	$\varphi_k$
1	0.986	$F/16 = 1/8T$	$\pi/3 = 60^\circ$
2	1.225	$F/8 = 1/4T$	$\pi/5 = 36^\circ$
3	0.707	$3F/16 = 3/8T$	$\pi/4 = 45^\circ$

Последовательность значений из 128 точек, полученных на ЭВМ по слегка измененной формуле

$$x(i) = \sum_{k=1}^3 a(k) \cos\left(\frac{2\pi k}{16} i + \varphi_k\right), \quad (6.21)$$

где  $i = 0, 1, \dots, 127$ , изображена на рис. 6.19. На рис. 6.20 изображены графики преобразования Фурье этой последователь-

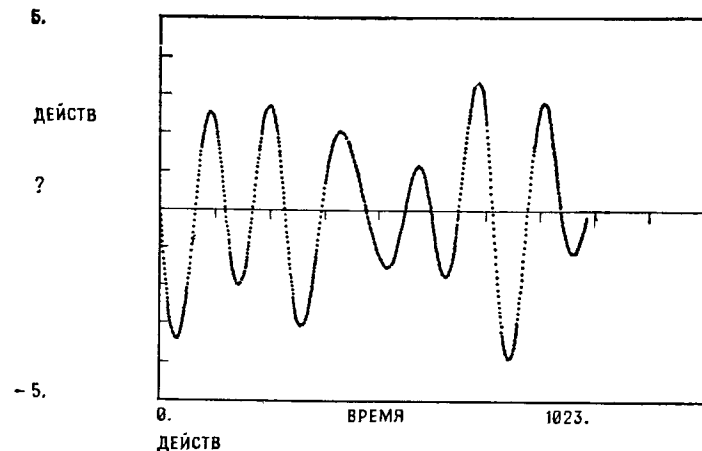


Рис. 6.19. График для трех синусоид без фаз.

ности в декартовой и в полярной системах координат. Рис. 6.21, на котором в полярных координатах изображено 1024-точечное

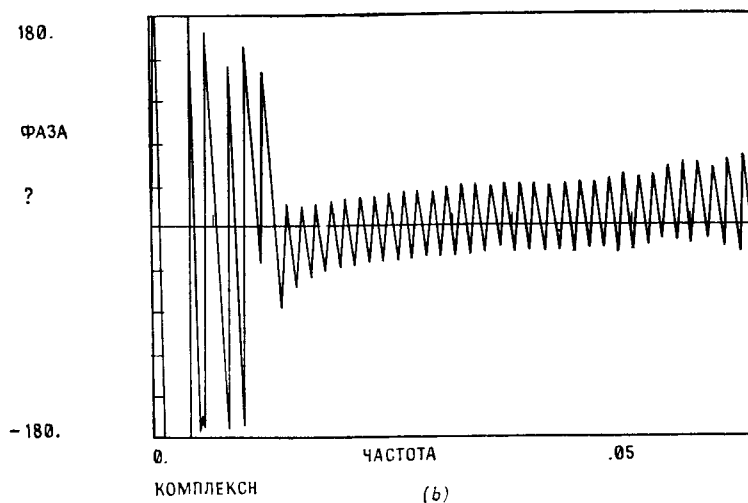
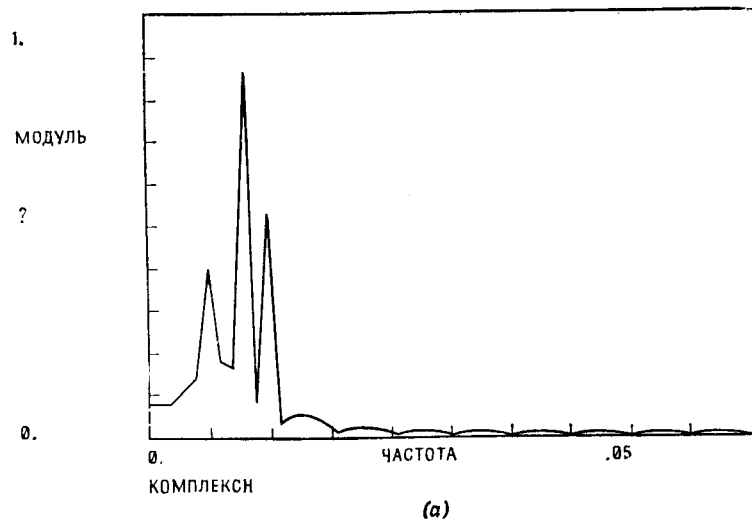
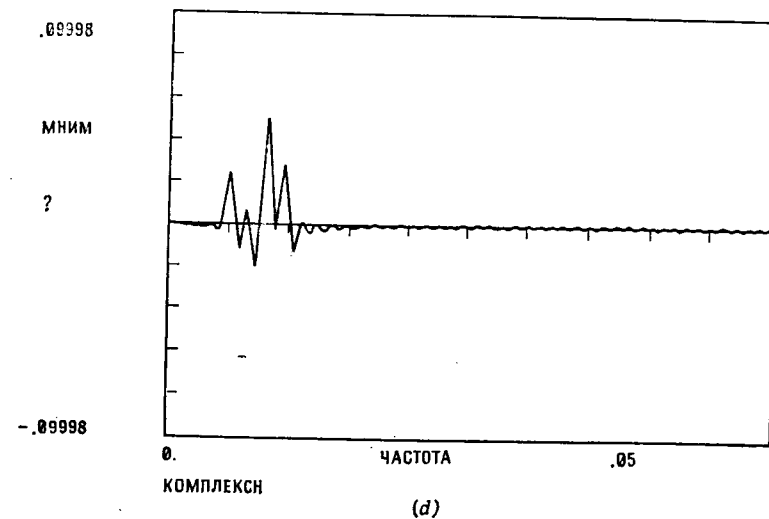
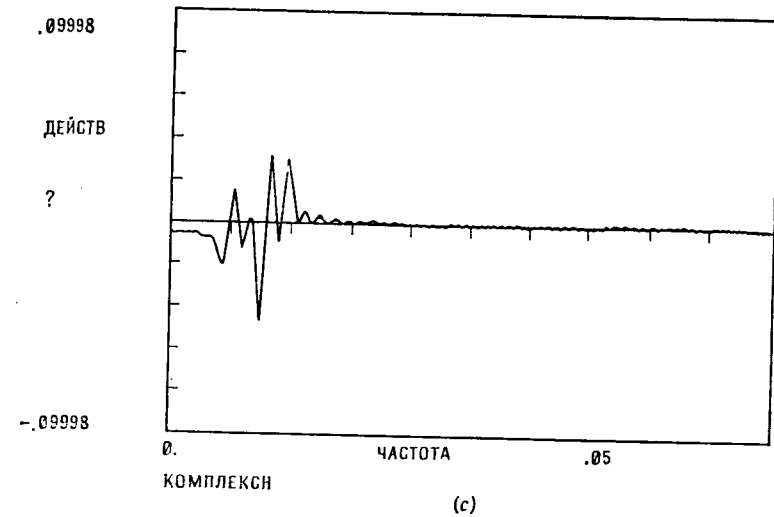


Рис. 6.20. Преобразование Фурье временной последовательности, изображенной на рис. 6.19.

преобразование Фурье той же последовательности, дает возможность проследить интерполяцию в частотной области.

**Стационарный случайный шум.** Рассмотрим преобразование Фурье стационарного случайного процесса. Для этой цели восполь-



зуется генератором аналогового гауссовского белого случайного шума, выход которого преобразуется в цифровую форму. По 1024 точкам во временной области получим 513 точек в области частот. Для случайного шума  $\mu_x = 0$ ,  $\sigma_x^2 = 1$  и значения данных почти независимы друг от друга. Действительная и мнимая части

преобразования Фурье имеют гауссовское распределение с параметрами

$$\begin{aligned} E[X_R(k)] &= E[X_I(k)] = 0, \\ D[X_R(k)] &= D[X_I(k)] = \frac{1}{2NT}. \end{aligned} \quad (6.22)$$

На рис. 6.22 показана сама временная последовательность, преобразованная в цифровую форму, а на рис. 6.23 — ее преоб-

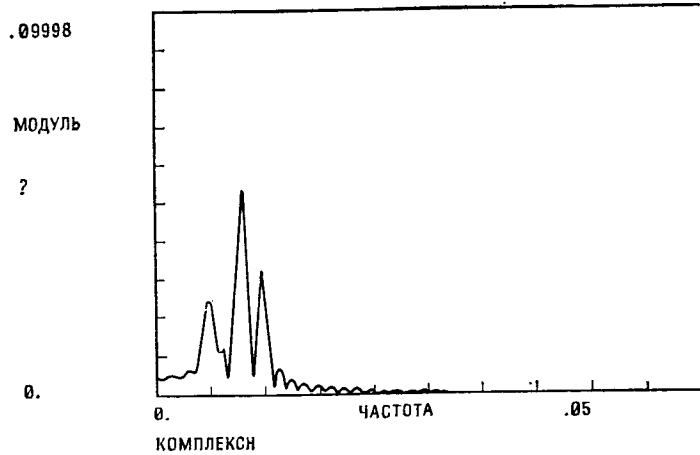


Рис. 6.21. Преобразование Фурье последовательности (6.21), дополненной 896 нулями.

разование Фурье. Амплитуда имеет распределение  $[\chi(2)]$  с двумя степенями свободы, где переменная  $\chi(n)$  есть квадратный корень из величины  $\chi^2(n)$ . Это распределение называют обычно распределением Релея.

**Переходные данные, полученные при помощи акселерометра.** Присоединим акселерометр к бойку небольшого ударника, который используется для удара по лопастям турбины. Это позволит определить импульсную функцию отклика при импульсной нагрузке (т. е. дельта-функции). Подробнее этот вопрос будет затронут в гл. 9. Сейчас мы хотим только показать вид преобразования Фурье для переходной (нестационарной) временной последовательности.

Данные преобразуются в цифровую форму со скоростью 4000 выб./с. График полученной временной последовательности приведен на рис. 6.24. Преобразование Фурье этой последовательности показано на рис. 6.25. Статистические результаты, которые справедливы для стационарных процессов, в данном случае неприменимы. Для переходных процессов такого рода

возникают многочисленные статистические трудности, обусловленные тем, что между полосами частот уже нет статистической независимости.

**Интерполяция функций на ограниченной полосе частот.** В этом пункте мы покажем, как проводится интерполяция значений дискретной функции на ограниченной полосе частот на примере распределения напряжения, полученного при структурном анализе строительных панелей.

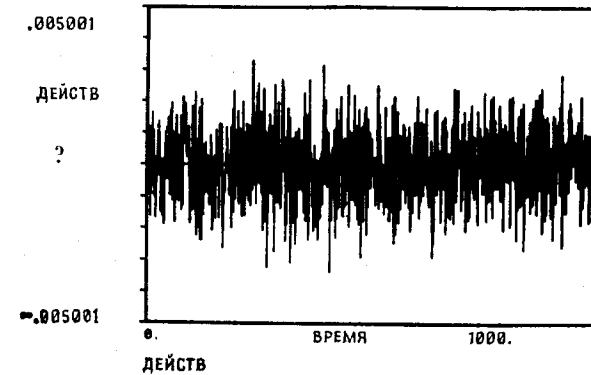


Рис. 6.22. Выход генератора гауссовского белого случайного шума, преобразованный в цифровую форму.

На рис. 6.26 показано, как определяется напряжение в двумерной плоскости при действии динамических точечных нагрузок. Вертикальное напряжение определяется при помощи конечного числа элементов, поэтому напряжение можно получить только в отдельных точках. Такой способ анализа дает общее представление о характере распределения напряжения, но сам по себе недостаточен для определения непрерывного распределения напряжения. На рис. 6.27 показано распределение напряжения на уровне, расположенном на 1.5 дюйма ниже плоскости нагрузки. Такая дискретная картина не дает точного представления о пиковых значениях напряжения под нагрузкой на 1.5-дюймовом уровне, поэтому ищут такие методы, чтобы оценить их более точно.

На рис. 6.28 приведен результат интерполяции имеющейся последовательности значений при помощи БПФ. Бросается в глаза, что пиковые значения напряжения интерполированной функции оказываются по величине значительно больше, чем у исходной функции. В этом и состоит основной результат проведенного анализа.

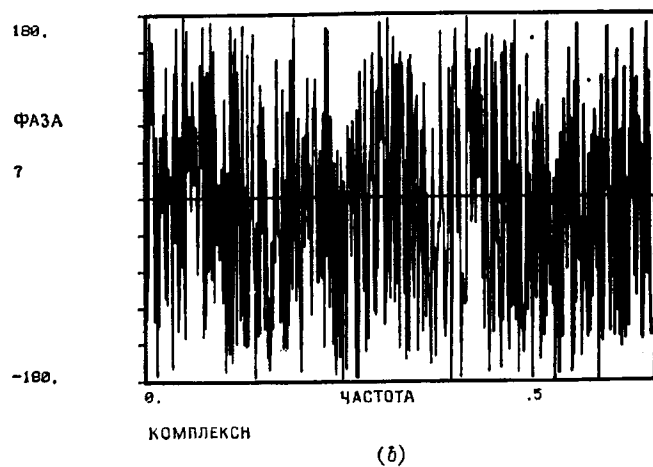
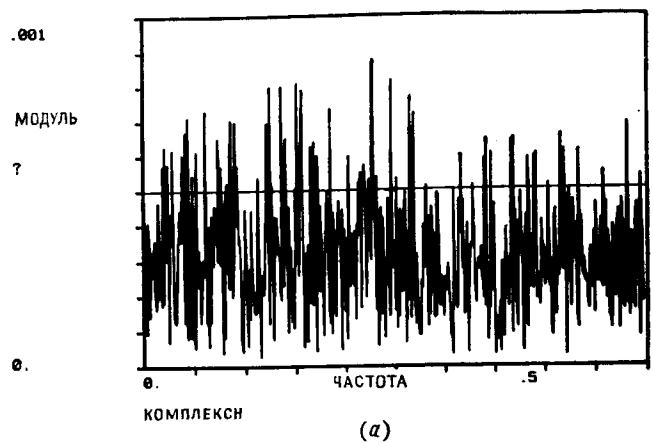
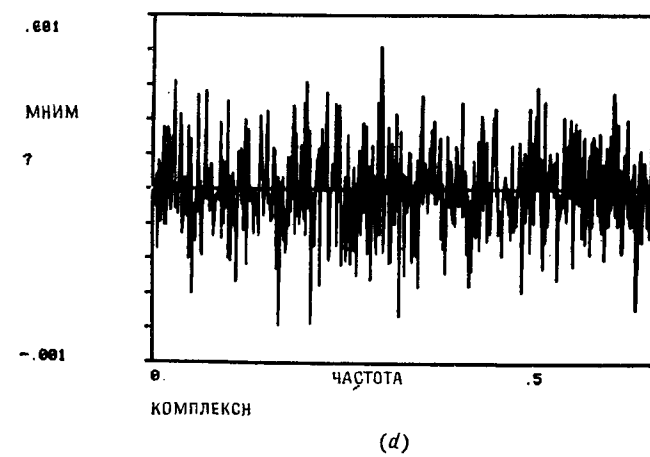
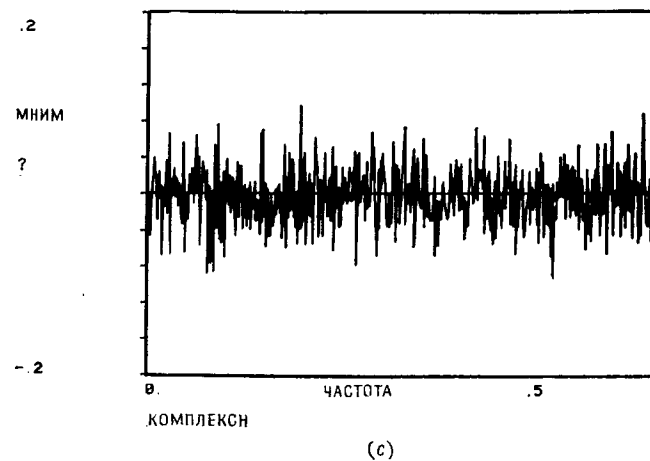


Рис. 6.23. Преобразование Фурье



стационарного белого шума.



При интерполяции к исходной последовательности сначала были добавлены нули, а затем для новой последовательности вычислялось преобразование Фурье. В исходной последователь-

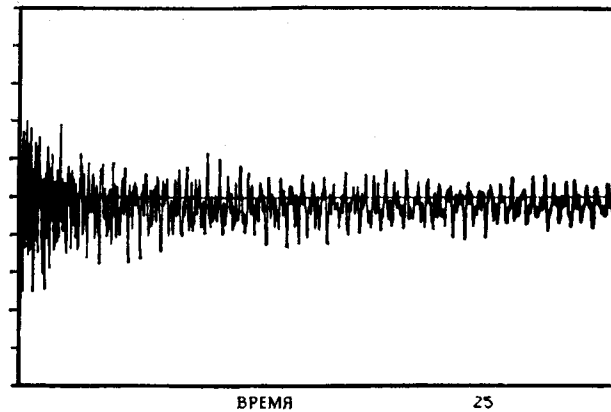


Рис. 6.24. Временная последовательность переходных данных, полученных при помощи акселерометра.

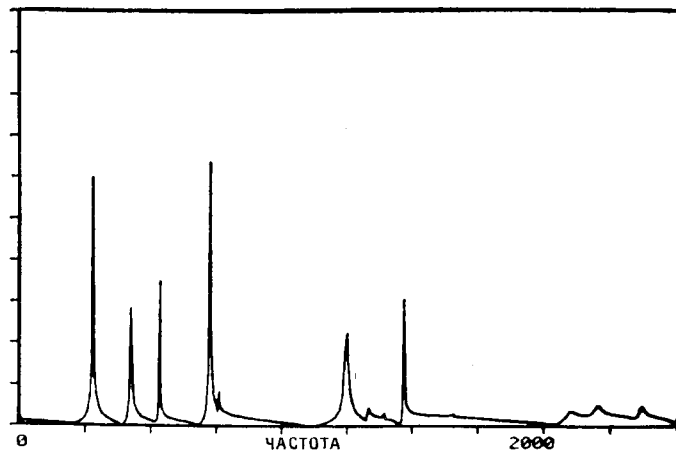


Рис. 6.25. Абсолютная величина преобразования Фурье переходных данных, приведенных на рис. 6.24.

ности было 16 точек, а в новой — 128 точек, т. е. на одно значение добавлялось по семь нулей.

**Узкополосный случайный шум.** На рис. 6.29, а—е показана картина, которую дает узкополосный шум. График на рис. 6.29

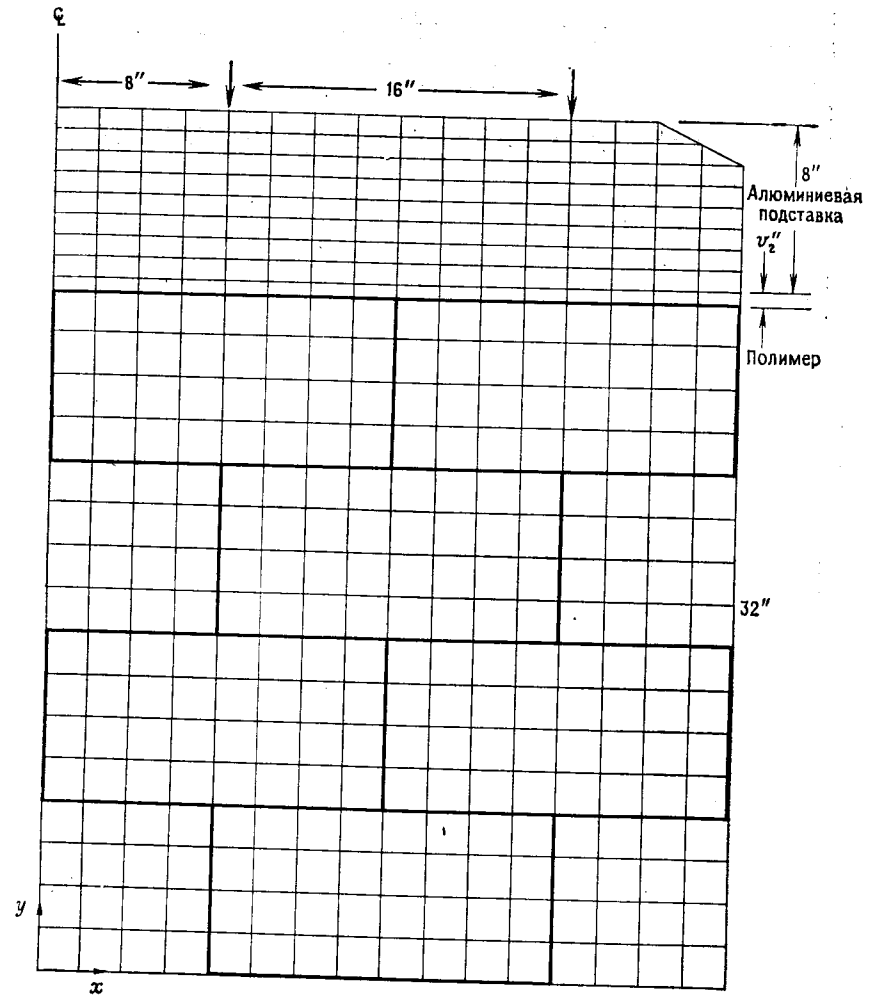


Рис. 6.26. Структурный анализ строительных панелей.

имеет известное сходство с модулированной синусоидой. Вместе с тем эти случаи имеют существенные различия, поскольку последовательность для шума дает узкую полосу в спектре и имеет случайную фазу.

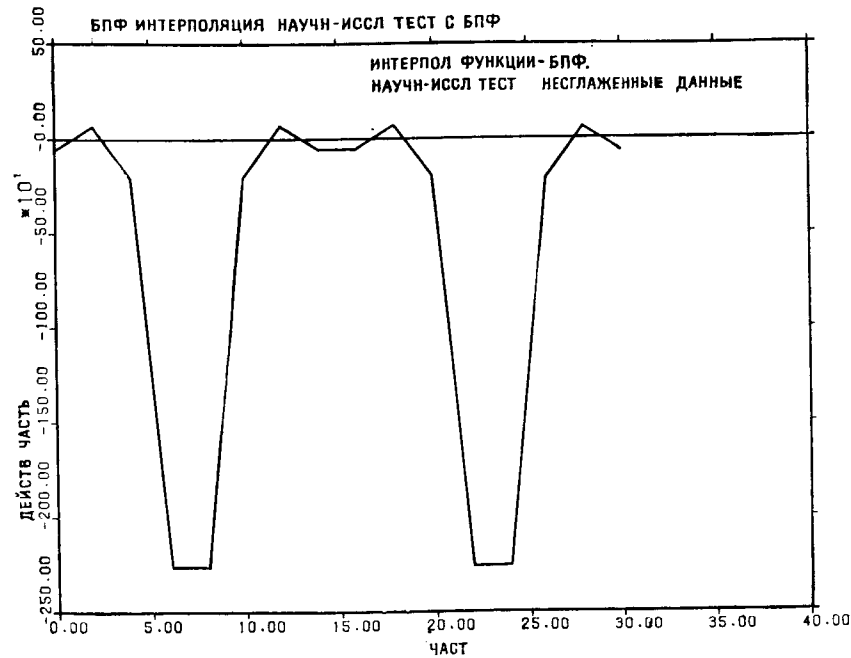


Рис. 6.27. Исходные дискретные значения функции напряжения.

### Упражнения

6.1. Докажите, что  $Z(N+k) = Z(k)$ , где

$$Z(k) = T \sum_{i=0}^{N-1} z(i) \exp\left(-j \frac{2\pi}{N} ik\right).$$

Ответ: поскольку  $\exp(-j2\pi i) = 1$  для всех  $i$ , имеем

$$\begin{aligned} z(i) \exp\left(-j \frac{2\pi}{N} 2[k+N]i\right) &= z(i) \exp\left(-j \frac{2\pi}{N} ik\right) \exp\left(-j \frac{2\pi}{N} iN\right) = \\ &= z(i) \exp\left(-j \frac{2\pi}{N} ik\right); \end{aligned}$$

следовательно,

$$\begin{aligned} Z(k+N) &= T \sum_{i=0}^{N-1} z(i) \exp\left(-j \frac{2\pi}{N} i[k+N]\right) = \\ &= T \sum_{i=0}^{N-1} z(i) \exp\left(-j \frac{2\pi}{N} ik\right). \end{aligned}$$

6.2. Вычислите преобразование для последовательности действительных величин из 128 точек, используя только один 64-точечный комплексный массив.

6.3. Докажите, что  $X(k) = X^*(N-k)$ , если  $x(i)$  — действительные числа.

6.4. Вычислите обратное преобразование Фурье последовательности, полученной в упр. 6.2, также используя один 64-точечный массив.

6.5. Покажите, что преобразование Фурье последовательности, состоящей из двух величин  $x(0)$ ,  $x(1)$ , дает величины  $X(0) = (x(0) + x(1))T$  и  $X(1) = (x(0) - x(1))T$ .

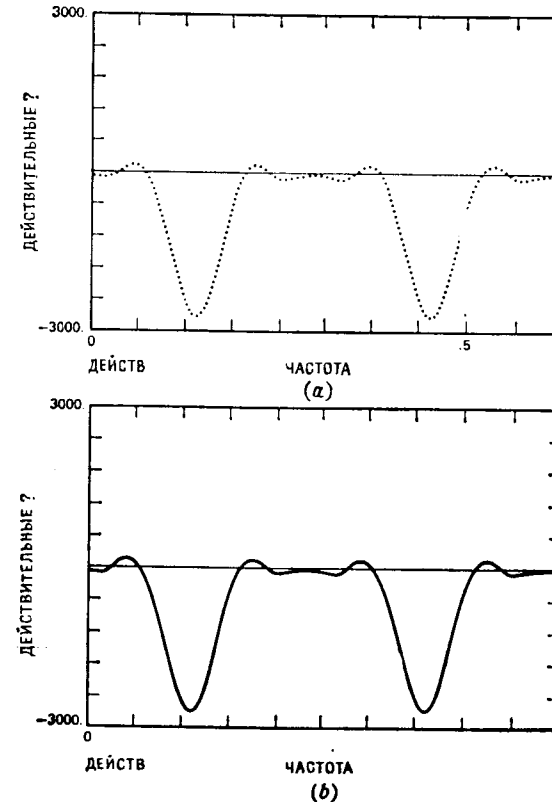


Рис. 6.28. Результат интерполяции функции напряжения посредством БПФ с добавлением нулей 8 к 1.

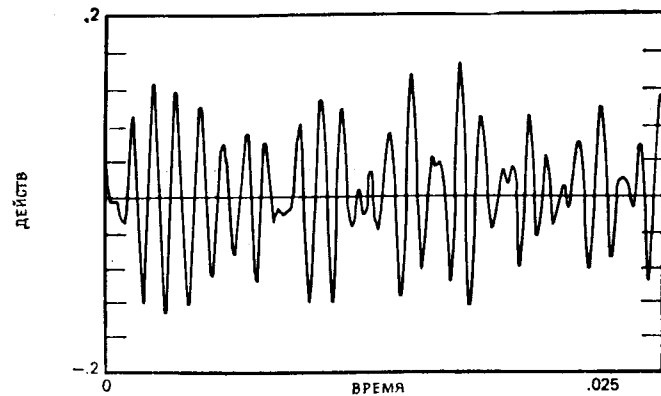
6.6. Получите преобразование, которое требуется в упр. 6.2, используя 32-точечный массив.

6.7. Докажите, что  $X(0) = T \sum_{i=0}^{N-1} x(i)$ .

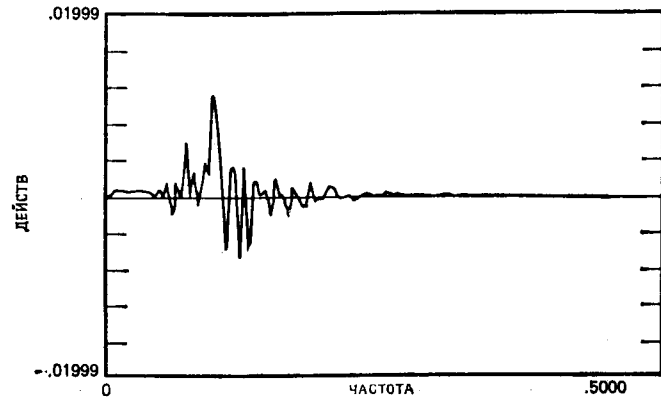
6.8. Переведите результаты упр. 6.2 в полярные координаты, т. е. выразите их через абсолютную величину и фазу.

6.9. Докажите, что при четном  $N$  выполняется соотношение

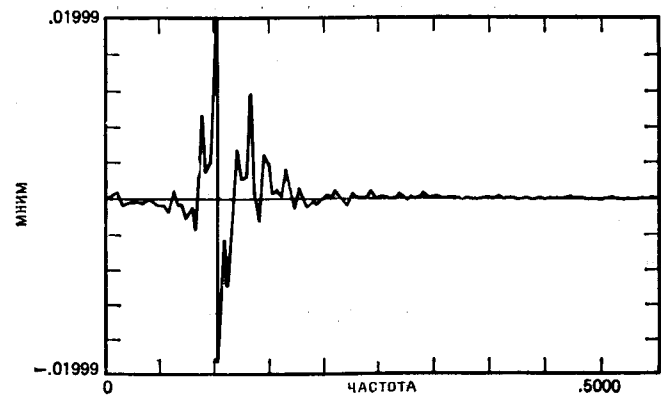
$$X(N/2) = \sum_{i=0}^{N-1} x(2i) - \sum_{i=0}^{N-1} x(2i+1).$$



(a) Временная последовательность



(b) Действительная часть преобразования Фурье



(c) Мнимая часть преобразования Фурье

6.10. Проведите продолжение фазы для последовательности, полученной в упр. 6.8.

6.11. Приведите вывод формул Эйлера

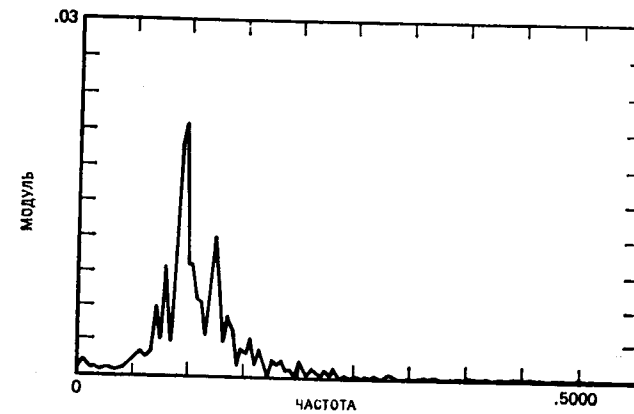
$$\sin(n\alpha) = 2 \sin((n-1)\alpha) \cos \alpha - \sin((n-2)\alpha),$$

$$\cos(n\alpha) = 2 \cos((n-1)\alpha) \cos \alpha - \cos((n-2)\alpha).$$

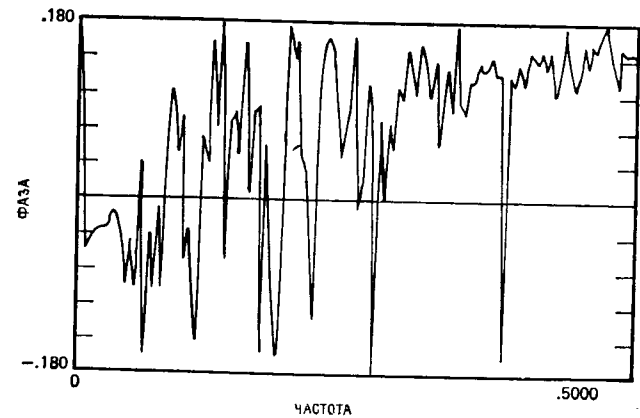
6.12. Получите выборку псевдослучайного белого шума, вычислите ее преобразование Фурье и изобразите результаты графически, воспользовавшись подпрограммой TRAND или какой-нибудь другой эквивалентной подпрограммой.

6.13. Перепишите подпрограмму FFT, в которой предусмотрено, что в первом проходе внешнего цикла умножения не нужны. Сколько операций сохранит такое усовершенствование?

6.14. Получите узкополосный процесс, изменив преобразование Фурье, полученное в упр. 6.12, и вычислите обратное преобразование при помощи



(d) Абсолютная величина преобразования Фурье



(e) Фаза преобразования Фурье

Рис. 6.29. Узкополосный случайный процесс и его преобразование Фурье.

обратного БПФ. Приведите график для получившейся временной последовательности.

6.15. Получите формулу (6.21) из определения (6.20).

Ответ: поскольку  $f_k = k/(16T)$ , для суммы

$$\sum a_k \cos(2\pi T f_k + \varphi_k)$$

имеем

$$\sum a_k \cos\left(2\pi T \frac{k}{16T} + \varphi_k\right) = \sum a_k \cos\left(2\pi k \frac{i}{16} + \varphi_k\right) = \sum a_k \cos\left(\frac{2\pi k}{16} i + \varphi_k\right).$$

6.16. Повторите графики, изображенные на рис. 6.29, используя результаты упр. 6.14.

6.17. Объясните, как получаются равенства (6.22).

Ответ: а) в силу того, что преобразование Фурье линейно, справедливы равенства  $E[L(x)] = LE(x) = 0$ ;

б) интеграл квадрата преобразования Фурье должен быть равным  $\sigma_x^2$ , поскольку

$$\Delta f = b = \frac{1}{NT}, \quad E|X^2| \frac{1}{NT} = \sigma_x^2$$

и

$$E \frac{X_R^2}{NT} + E \frac{X_I^2}{NT} = \sigma_x^2;$$

так как  $EX_R^2 = EX_I^2$ , то  $2E[X_R^2/NT] = \sigma_x^2$ , что и требовалось доказать.

6.18. Докажите, что модуль  $|S(f) + a \exp[-i2\pi ft]|$  содержит синусоидальную составляющую.

Ответ:

$$\begin{aligned} |S(f) + a e^{-i\omega t}|^2 &= |S(f)|^2 + a^2 + 2\operatorname{Re}[aS(f) e^{-i\omega t}] = \\ &= |S(f)|^2 + a^2 + 2a \{\operatorname{Re}[S(f) \cos(\omega t)] + \operatorname{Im}[S(f) \sin(\omega t)]\}. \end{aligned}$$

## Глава 7

### КОВАРИАЦИЯ И СВЕРТКА

#### 7.1. ЭЛЕМЕНТЫ ТЕОРИИ

Содержание этой главы составляют вопросы, связанные с ковариационными функциями и сверткой. Несмотря на то что во многих приложениях ковариация и свертка играют различные роли, мы рассмотрим их вместе, поскольку в вычислительном плане получение их мало чем отличается.

Хотя в анализе временных рядов термины «корреляционная функция» и «ковариационная функция» зачастую выступают как синонимы, в этой главе мы будем их разграничивать и пользоваться более устоявшимся в статистике термином ковариация, оставляя за корреляцией понятие нормированной величины со значениями в отрезке  $[-1, 1]$ . Далее будем придерживаться статистической терминологии.

Ковариационные (корреляционные) функции находят широкое применение в анализе вибраций и ударов, в задачах, связанных с передачей информации, радарным обнаружением, системами управления, а также в анализе данных океанографических исследований, словом, во всех областях, где уместно применение соотношений спектрального анализа, использующих преобразования Фурье. Если в задаче для анализа используются спектр мощности и кросс-спектр, то соответствующие корреляционные функции в принципе также могут применяться, поскольку они тесно связаны со своими преобразованиями Фурье. На практике могут существовать веские причины для выбора той или иной возможности. Например, время относительного смещения для двух сигналов можно определять по фазе кросс-спектра, но нередко более целесообразным в этом случае оказывается применение (кросс-) ковариационной функции. Уточнение «кросс» для ковариации излишне, однако мы часто прибегаем к этому, чтобы не путать ее с автоковариацией.

Можно говорить и о столь же широком применении функции свертки. Следует, впрочем, отметить, что большую часть таких применений можно отнести к цифровой фильтрации, поскольку свертка временной последовательности с импульсной функцией отклика линейного фильтра дает отклик (выход) фильтра. Поэтому функция свертки полезна при решении таких задач, как сглаживание (низкополосная фильтрация) данных и генерирование