

Ордена Ленина

ИАЭ-3178

Институт атомной энергии

им. И. В. Курчатова

В. И. Носов, Г. В. Компаниец,
Р. П. Петрушенко

Нормирование функций Бесселя и поиск главного собственного значения в программе РНК

Москва 1979

ОРДЕНА ЛЕНИНА
ИНСТИТУТ АТОМНОЙ ЭНЕРГИИ им. И. В. КУРЧАТОВА

В.И. Носов, Г.В. Компаниец,
Р.П. Петрушенко

НОРМИРОВАНИЕ ФУНКЦИЙ БЕССЕЛЯ
И ПОИСК ГЛАВНОГО СОБСТВЕННОГО ЗНАЧЕНИЯ
В ПРОГРАММЕ РНК

М о с к в а

1979

Ключевые слова: нормирование функций Бесселя, гетерогенный реактор, поиск первого корня.

Излагается методика нормирования функций Бесселя целочисленного порядка N действительного и мнимого аргумента.

Рассматривается вопрос поиска положительного главного собственного значения (первого корня) применительно к гетерогенной программе *PNK*.

Описывается метод нормировки потоков нейтронов на единичную мощность.

Данная работа является дополнением к ранее опубликованному препринту тех же авторов [1].

В в е д е н и е

В работе [1] в рамках двухгруппового диффузионного приближения производится расчет эффективного коэффициента размножения, потоков нейтронов и сопряженных функций для цилиндрического реактора, состоящего из концентрических зон разного состава в любом сочетании, при этом в одной из зон произвольно по сечению могут быть расположены цилиндрические блоки конечного размера, введенные на всю эффективную высоту реактора. Решение для потока нейтронов в [1] было получено в матричной форме с помощью рядов Фурье, учитывающих азимутальную зависимость потока по реактору и около поверхности блоков. При реализации разработанного алгоритма на ЭВМ в полученном решении на поверхности блоков была учтена только первая гармоника (монопольный член). В разложении, которое определяет угловую зависимость потока по реактору, может быть оставлено при проведении расчетов до 40 высших членов ряда (сумма по n). Как было отмечено в [1], при больших геометрических размерах зон системы аргумент цилиндрических функций в полученном решении может достигать значений 50–100 и даже более (для функций $I_n(x)$, $K_n(x)$). В этом случае при проведении расчетов на ЭВМ происходит останов машины или потеря точности счета, поскольку значения функций $I_n(x)$ по порядку величины могут

превышать 10^{19} , а функции $K_n(x)$ при этих значениях аргумента будут стремиться к нулю. В подобных случаях [2] может оказаться полезной специальная нормировка функций, чтобы при вычислениях избежать потерь значащих цифр, связанных с фиксированной разрядностью БЭСМ-6. С учетом этого обстоятельства в полученном решении были введены нормированные функции для индекса $n = 0$, так чтобы функция $I_0^{\text{норм}}(x)$ равнялась 1 на внешней границе в каждой из зон реактора, а функция $K_0^{\text{норм}}(x)$ была равна 1 на ее внутренней границе ($x = x_{\text{min}}$). Однако, как показали дальнейшие расчеты, при такой нормировке с хорошей степенью точности учитываются только первые гармоники в разложении по n , что может оказаться явно недостаточным при вычислении распределения потоков нейтронов в задачах с резко выраженной асимметрией. Следует отметить, что при расчете высших гармоник с увеличением порядка n значения функций $K_n(x)$ довольно сильно возрастают, а значения функций $I_n(x)$ резко убывают в области малых величин аргумента. Аналогичным образом изменяются и функции действительного аргумента $Y_n(x)$ и $J_n(x)$ с ростом индекса n при $n > x$. В конечном итоге это также приводит либо к останову машины, либо к потере точности счета в процессе вычислений. Во избежание возникновения подобной ситуации и была произведена нормировка всех функций Бесселя $K_n(x)$, $I_n(x)$ и $Y_n(x)$, $J_n(x)$ любого порядка n с учетом характера поведения каждой из них в отдельности. Результаты этого исследования изложены в первом разделе данной работы. Рассмотренные вопросы представляют в известной степени и самостоятельный интерес, поскольку введение нормированных функций Бесселя любого порядка n позволяет устранить расчетные трудности, которые могут иметь место при использовании функций Бесселя для изучения подобного рода задач (цилиндрические системы).

При проведении расчетов значительные сложности возникают также и в процессе поиска главного собственного значения задачи (пер-

вого корня) вследствие наличия особых точек. Наибольшие трудности при поиске наблюдаются в тех случаях, когда корни находятся вблизи особых точек или расположены слишком близко друг к другу (большие реакторы). Предлагается несколько способов определения первого корня $K_{эф}$, что дает возможность пользователю при проведении расчетов по программе **PNK** выбрать оптимальный вариант подбора $K_{эф}$ с учетом особенностей рассматриваемой задачи. Вышеуказанные вопросы нашли свое отражение во втором разделе работы.

И, наконец, в последнем разделе кратко изложен метод нормирования потоков нейтронов на единичную мощность реактора.

1. НОРМИРОВАНИЕ ФУНКЦИЙ БЕССЕЛЯ

Описывается методика нормирования функций Бесселя действительного и мнимого аргументов $J_n(x)$, $Y_n(x)$, $I_n(z)$, $K_n(z)$ первого и второго рода для действительного целого индекса n (порядок). Вычисление вышеуказанных функций Бесселя производится с учетом возможных максимальных значений аргументов, которые могут иметь место в практике расчета подобного класса физических задач. Для функций $I_n(z)$, $K_n(z)$ значение аргумента z может достигать 100 и более, в то же время значение аргумента функций $J_n(x)$, $Y_n(x)$ по порядку величины обычно не превосходит 10.

В последних двух подразделах данного раздела излагается методика расчета произведений функций $J_n(z) * Y_n(z)$ и $I_n(z) * K_n(z)$, которые применяются в гетерогенных задачах при использовании теоремы сложения для бесселевых функций [3] в результате перехода от системы полярных координат с полюсами в центре блоков к полярным координатам с полюсом в центре реактора ($Z > z$).

1.1. функции Бесселя мнимого аргумента второго рода $K_n(x)$

Функции Бесселя $K_n(x)$ в области больших значений аргумента убывают по экспоненциальному закону с ростом x , а при $x \rightarrow 0$

неограниченно возрастают. С увеличением порядка n функции $K_n(x)$ сильно возрастают при фиксированном x , особенно в области малых значений аргумента (рис. 1,а). В данной работе предлагается нормировать функции Бесселя $K_n(x)$ на их максимальные значения на границах цилиндрических зон реактора, так чтобы все $K_n^{\text{норм}}(x)$ при любом n были равны 1 на внутренней границе в каждой из зон ($x = x_{\min}$):

$$K_n^{\text{норм}}(x) = \frac{K_n(x)/K_0(x)}{K_n(x_{\min})/K_0(x_{\min})} * \frac{K_0(x)}{K_0(x_{\min})} * A_n. \quad (1)$$

Здесь A_n - некоторый нормирующий множитель, зависящий от порядка функций n ($A_n = 1$ при $n = 0, 1$).

Расчет нормированных функций $K_n^{\text{норм}}(x)$ в программе *PNK* начинается для нулевого и первого порядков. Если при этом значения аргументов x и x_{\min} меньше 20, то расчет $K_0(x)$, $K_1(x)$ и $K_0(x_{\min})$, $K_1(x_{\min})$ производится по соответствующим подпрограммам *BESK0(z)* и *BESK1(z)*, имеющимся в общественной библиотеке ЭВМ.

Тогда соотношение (1) принимает вид

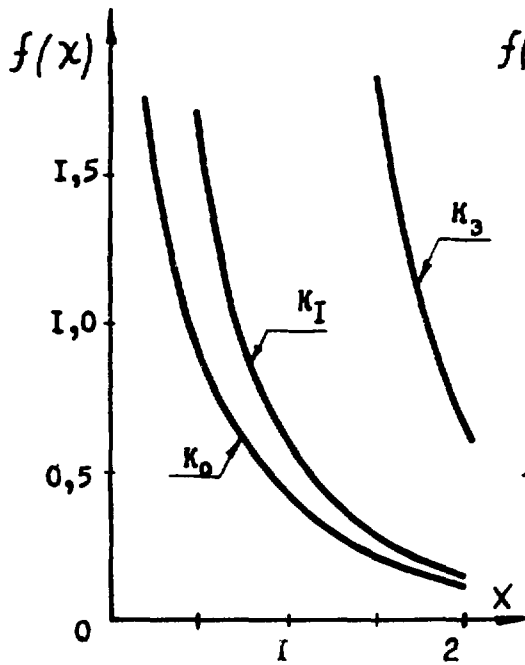
$$K_0^{\text{норм}}(x) = \frac{K_0(x)}{K_0(x_{\min})}, \quad K_1^{\text{норм}}(x) = \frac{K_1(x)}{K_1(x_{\min})}.$$

Если же значения аргументов x и x_{\min} больше 20, то вычисление бесселевых функций производится по соответствующим асимптотическим разложениям [3]:

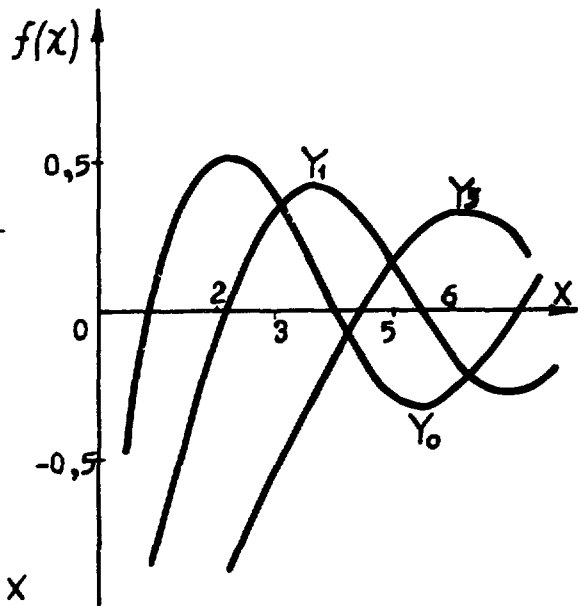
$$K_0(z) = e^{-z} * FT(0, z), \quad K_1(z) = e^{-z} * FT(1, z), \quad (2)$$

где

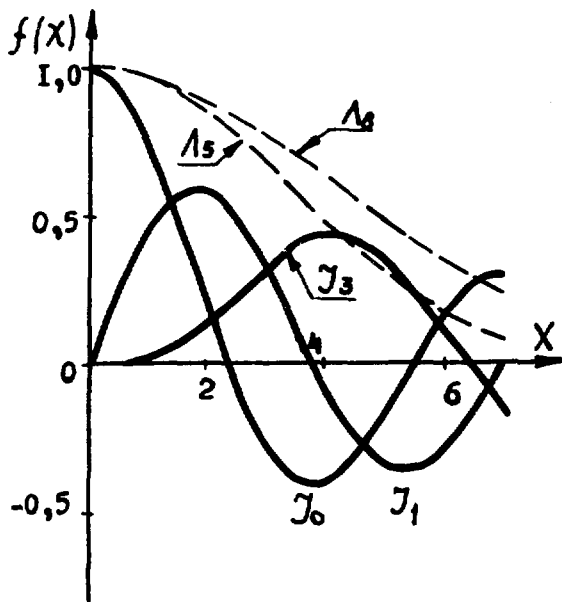
$$FT(0, z) = \sqrt{\frac{\pi}{2z}} * \left(1 - \frac{1^2}{8z} + \frac{1^2 \cdot 3^2}{2! \cdot (8z)^2} - \frac{1^2 \cdot 3^2 \cdot 5^2}{3! \cdot (8z)^3} + \dots \right);$$



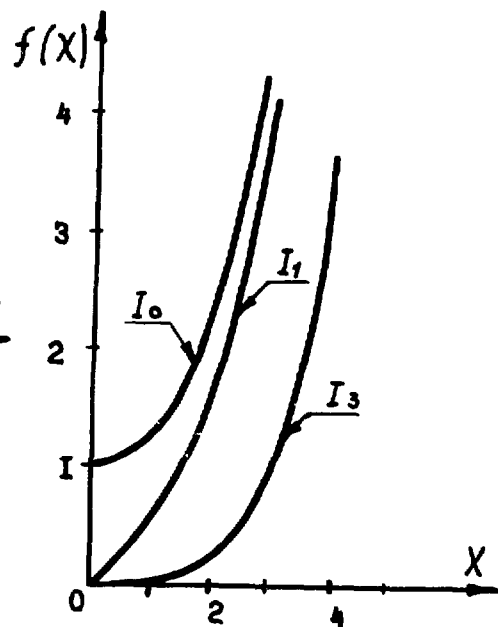
а) Функции $K_n(x)$



б) Функции $Y_n(x)$



в) Функции $J_n(x), L_n(x)$



г) Функции $I_n(x)$

Рис. 1. Графики функции Бесселя

$$FT(1,2) = \sqrt{\frac{9\pi}{22}} * \left(1 + \frac{1.3}{82} - \frac{1^2 \cdot 3 \cdot 5}{2! \cdot (82)^2} + \frac{1^2 \cdot 3^2 \cdot 5 \cdot 7}{3! \cdot (82)^3} + \dots \right).$$

В этом случае расчет функций $K_0^{\text{норм}}(x)$ и $K_1^{\text{норм}}(x)$ происходит следующим образом:

$$K_0^{\text{норм}}(x) = e^{(x_{\min} - x)} * \frac{FT(0, x)}{FT(0, x_{\min})}, \quad K_1^{\text{норм}}(x) = e^{(x_{\min} - x)} * \frac{FT(1, x)}{FT(1, x_{\min})}.$$

Если аргумент x лежит в области значений больше 20, а x_{\min} меньше 20, то вычисление рассматриваемых функций осуществляется по формулам

$$K_0^{\text{норм}}(x) = e^{-x} * \frac{FT(0, x)}{K_0(x_{\min})}, \quad K_1^{\text{норм}}(x) = e^{-x} * \frac{FT(1, x)}{K_1(x_{\min})}.$$

Расчет нормированных функций $K_n^{\text{норм}}(x)$ более высоких порядков ($n \geq 2$) производится в соответствии с соотношением (1), при этом функции $K_n(x)/K_0(x)$ и $K_n(x_{\min})/K_0(x_{\min})$ находятся из рекуррентного соотношения [3]

$$\frac{K_{m+1}(z)}{K_0(z)} = \frac{K_{m-1}(z)}{K_0(z)} + \frac{2 \cdot m}{z} * \frac{K_m(z)}{K_0(z)}. \quad (3)$$

Так, в частности, при $m = 1$ имеем

$$\frac{K_2(z)}{K_0(z)} = 1 + \frac{2 \cdot 1}{z} * \frac{K_1(z)}{K_0(z)}$$

Здесь функции $K_1(z)/K_0(z)$ вычисляются одновременно с расчетом $K_0^{норм}(x)$ и $K_1^{норм}(x)$.

В соответствии с приведенным рекуррентным соотношением (3) определяются числитель и знаменатель первого множителя в формуле (1) последовательно для всех значений n ; второй множитель в (1) есть $K_0^{норм}(x)$. Однако в связи с тем что функции $K_n(z)$ с увеличением порядка n неограниченно возрастают при фиксированном z (особенно в области малых значений аргумента), в процессе счета может произойти переполнение и останов ЭВМ. Для устранения подобной нежелательной ситуации, как только функции $K_n(x)/K_0(x)$ или $K_n(x_{min})/K_0(x_{min})$ при некотором значении n становятся больше наперед заданной величины (в программе 10^6), числитель или знаменатель первого множителя в формуле (1) автоматически умножается на один и тот же коэффициент, который изменяет только порядок числа. Указанный коэффициент может быть, вообще говоря, переменной величиной; его значение в программе принято равным 10^{-6} .

Для сохранения значений функций $K_n^{норм}(x)$ в соотношении (1) имеется нормирующий множитель A_n , который представляет собой комбинацию вышеуказанных числовых коэффициентов. Поскольку при $x_{min} < x$ функции $K_n(x_{min})/K_0(x_{min})$ возрастают быстрее, чем $K_n(x)/K_0(x)$, нормированные функции $K_n^{норм}(x)$ убывают с ростом порядка n при заданном x , а множитель A_n всегда остается равным или меньшим единицы. Следует также отметить, что в рекуррентном соотношении (3) значения всех последующих функций при $m+1 > n$ автоматически понижаются в программе на 6 порядков всякий раз, как только $K_n(z)/K_0(z)$ становится больше заданной величины 10^6 .

Производная от функции $K_n^{\text{норм}}(x)$ выражается следующим образом:

$$\dot{K}_n^{\text{норм}}(x) = \frac{\dot{K}_n(x)}{K_n(x_{\min})} * \dot{x} * A_n,$$

или

$$\dot{K}_n^{\text{норм}}(x) = \frac{\dot{K}_n(x) / K_0(x)}{K_n(x_{\min}) / K_0(x_{\min})} * \frac{K_0(x)}{K_0(x_{\min})} * \dot{x} * A_n^{(4)},$$

где $\dot{K}_n(x) = - \frac{K_{n-1}(x) + K_{n+1}(x)}{2}$.

Процедура расчета $\dot{K}_n^{\text{норм}}(x)$ аналогична вышерассмотренной при вычислении функций $K_n^{\text{норм}}(x)$.

1.2. Функции Бесселя действительного аргумента второго рода $Y_n(x)$

Функции Бесселя $Y_n(x)$ в области малых значений аргумента неограниченно убывают с ростом порядка n при фиксированном x ; при больших величинах аргумента функции $Y_n(x)$ асимптотически убывают. Нормирование функций $Y_n(x)$ проводится с учетом того обстоятельства, что они имеют корни в исследуемой области изменения аргумента (рис. 1,б). Если для любого порядка n функции $Y_n(x)$ не обращаются в нуль на внутренней границе рассматриваемой зоны реактора ($x = x_{\min}$), то в пределах этой зоны предлагается нормировать их таким образом, чтобы все $Y_n^{\text{норм}}(x)$ были равны 1 на ее внутренней границе (в программе задается условие $x_{\min} \leq 0,5$):

$$Y_n^{\text{норм}}(x) = \frac{Y_n(x) / Y_0(x)}{Y_n(x_{\min}) / Y_0(x_{\min})} * \frac{Y_0(x)}{Y_0(x_{\min})} * B_n \quad (5)$$

при $0 < x_{\min} \leq 0,5$.

Здесь и далее B_n - некоторый нормирующий множитель, зависящий от порядка функций n ($B_n = 1$ при $n = 0, 1$).

Если в рассматриваемой зоне реактора X_{min} превышает величину 0,5, то в этой зоне предлагается нормировать функции $Y_n(x)$; начиная с $n \geq 2$, нижеследующим образом:

$$Y_n^{норм}(x) = Y_n(x) \quad \text{при } n = 0, 1; \quad (6)$$

$$Y_n^{норм}(x) = \frac{Y_n(x)}{|Y_{n-1}(x_{min})| + |Y_n(x_{min})|} * B_n$$

при $n \geq 2$.

Здесь $X_{min} > 0,5$.

Естественно, выбор способа нормирования происходит в программе автоматически в процессе счета применительно к конкретной исследуемой задаче. Расчет нормированных функций $Y_n^{норм}(x)$ в программе начинается для нулевого и первого порядков, вычисление $Y_0(x)$, $Y_1(x)$ или $Y_0(x_{min})$, $Y_1(x_{min})$ производится по соответствующим подпрограммам **BESY0(z)** и **BESY1(z)**, имеющимся в общественной библиотеке ЭВМ. Вычисление нормированных функций $Y_n^{норм}(x)$ более высоких порядков ($n \geq 2$) происходит в соответствии с соотношениями (5) или (6), при этом функции $Y_n(x)/Y_0(x)$, $Y_n(x_{min})/Y_0(x_{min})$ или $Y_n(x)$, $Y_n(x_{min})$ находятся из рекуррентных соотношений [3]:

$$Y_{m+1}(z) = Y_m(z) * \frac{2m}{z} - Y_{m-1}(z) \quad \text{при } X_{min} > 0,5; \quad (7)$$

$$\frac{Y_{m+1}(z)}{Y_0(z)} = \frac{Y_m(z)}{Y_0(z)} * \frac{2m}{z} - \frac{Y_{m-1}(z)}{Y_0(z)} \quad \text{при } X_{min} \leq 0,5. \quad (8)$$

В соответствии с приведенными рекуррентными соотношениями (7), (8) определяются числитель и знаменатель первого множителя в формулах (5) или (6) последовательно для всех значений n . Поскольку функции $|Y_n(z)|$ в исследуемой области определения с увеличением порядка n неограниченно возрастают при фиксированном аргументе, то, как и в случае расчета $K_n^{\text{норм}}(x)$, числитель или знаменатель первого сомножителя в формулах (5), (6) умножается на числовой коэффициент 10^{-6} . Для сохранения значения функций $Y_n^{\text{норм}}(x)$ в указанных формулах имеется нормирующий множитель B_n , представляющий собой комбинацию вышеуказанных числовых коэффициентов. Следует отметить, что при больших величинах порядков n функции $|Y_n^{\text{норм}}(x)|$ при $x = x_{\min}$ также стремятся к единице и в случае $x_{\min} > 0,5$ ($|Y_{n-1}(x_{\min})| < |Y_n(x_{\min})|$).

Производные от функций $Y_n^{\text{норм}}(x)$ выражаются следующим образом:

$$\dot{Y}_n^{\text{норм}}(x) = \frac{\dot{Y}_n(x) / Y_0(x)}{Y_n(x_{\min}) / Y_0(x_{\min})} * \frac{Y_0(x)}{Y_0(x_{\min})} * \dot{x} * B_n;$$

при $x_{\min} \leq 0,5$

при $x_{\min} > 0,5$

$$\dot{Y}_n^{\text{норм}}(x) = \dot{Y}_n(x) \quad \text{для } n = 0, 1;$$

$$\dot{Y}_n^{\text{норм}}(x) = \frac{\dot{Y}_n(x)}{|Y_{n-1}(x_{\min})| + |Y_n(x_{\min})|} * \dot{x} * B_n \quad \text{для } n \geq 2.$$

Процедура расчета $\dot{Y}_n^{\text{норм}}(x)$ аналогична вышерассмотренной при вычислении функций $Y_n^{\text{норм}}(x)$.

1.3. Функции Бесселя действительного аргумента первого рода $J_n(x)$

Функции Бесселя $J_n(x)$ ограничены во всей области изменения аргумента x , а при $n > x$ довольно резко убывают с ростом порядка n . Нормирование функций $J_n(x)$ проводится с учетом того

обстоятельства, что они имеют корни в исследуемой области изменения аргумента при расчете подобного типа физических задач (рис. 1, в). Предлагается нормировать функции $J_n(x)$, начиная с $n \geq 2$, следующим образом:

$$J_n^{\text{норм}}(x) = J_n(x) \quad \text{при } n = 0, 1; \quad (9)$$

$$J_n^{\text{норм}}(x) = \frac{J_n(x)}{|J_n(x_{\text{max}})| + |J_{n+2}(x_{\text{max}})|} \quad \text{при } n \geq 2. \quad (10)$$

При больших величинах порядков n функции $|J_n^{\text{норм}}(x)|$ при $x = x_{\text{max}}$ стремятся к 1, так как $|J_{n+2}(x_{\text{max}})| < |J_n(x_{\text{max}})|$.

Поскольку в процессе счета знаменатель в формуле (10) может стать меньше 10^{-10} при больших n , то это, естественно, приведет к останову машины. Для устранения такой возможной ситуации в программе предусмотрен переход от функций $J_n(x)$ к функциям $\Lambda_n(x)$, как только знаменатель в соотношении (10) станет меньше наперед заданной величины (в программе 10^{-6}). Связь между функциями $J_n(x)$ и $\Lambda_n(x)$ следует из приведенного ниже соотношения [4]

$$J_n(x) = \frac{(x/2)^n}{\Gamma(n+1)} * \Lambda_n(x), \quad (11)$$

где

$$\Lambda_n(x) = \sum_{k=0}^{\infty} \frac{1}{k!} * \frac{\Gamma(n+1)}{\Gamma(n+k+1)} * \left(\frac{ix}{2}\right)^{2k}. \quad (12)$$

В отличие от функций Бесселя $J_n(x)$ функции $\Lambda_n(x)$ стремятся к единице с ростом n в области малых аргументов ($\Lambda_n(0) \equiv 1$ при любом n), а при $n > x$ возрастают с увеличением порядка n . С учетом соотношения (11) формула (10) для нормированных функций Бесселя $J_n^{\text{норм}}(x)$ при $n \geq 2$ запишется теперь следующим образом:

$$J_n^{\text{норм}}(x) = \left(\frac{x}{x_{\text{max}}}\right)^n * \frac{\Lambda_n(x)}{|\Lambda_n(x_{\text{max}})| + 0,5 * x_{\text{max}} * |\Lambda_{n+1}(x_{\text{max}})| / (n+1)}$$

при $|J_n(x_{\text{max}})| + |J_{n+1}(x_{\text{max}})| \leq 10^{-6}$.

Производные от функций $J_n^{\text{норм}}(x)$ имеют вид

$$\dot{J}_0^{\text{норм}}(x) = -J_1(x) * \dot{x}, \quad \dot{J}_1^{\text{норм}}(x) = (J_0(x) - J_2(x)) * 0,5 * \dot{x};$$

при $n \geq 2$

$$\dot{J}_n^{\text{норм}}(x) = \frac{J_{n-1}(x) - J_{n+1}(x)}{|J_n(x_{\text{max}})| + |J_{n+1}(x_{\text{max}})|} * 0,5 * \dot{x}.$$

В случае $|J_n(x_{\text{max}})| + |J_{n+1}(x_{\text{max}})| \leq 10^{-6}$ получаем с учетом соотношения (10) следующую формулу для расчета

$\dot{J}_n^{\text{норм}}(x)$ при $n \geq 2$:

$$\dot{J}_n^{\text{норм}}(x) = \left(\frac{x}{x_{\text{max}}}\right)^{n-1} * \frac{\Lambda_{n-1}(x) - \left(\frac{x}{2}\right)^2 * \frac{\Lambda_{n+1}(x)}{n \cdot (n+1)}}{\frac{x_{\text{max}}}{2} * \frac{|\Lambda_n(x_{\text{max}})|}{n} + \left(\frac{x_{\text{max}}}{2}\right)^2 * \frac{|\Lambda_{n+1}(x_{\text{max}})|}{n \cdot (n+1)}} * 0,5 * \dot{x}.$$

Вычисление нормированных функций $J_n^{\text{норм}}(x)$, а также их производных начинается в программе для нулевого и первого порядков с использованием соответствующих подпрограмм **BESJO(2)** и **BESY1(z)**, имеющихся в общественной библиотеке ЭВМ.

Для расчета $J_n^{\text{норм}}(x)$ и $\dot{J}_n^{\text{норм}}(x)$ при $n \geq 2$ используются соотношения (11) или (12).

1.4. Функции Бесселя мнимого аргумента первого рода $I_n(x)$

Функции Бесселя $I_n(x)$ в области больших значений аргумента возрастают по экспоненциальному закону с ростом x , а при малых величинах аргумента и $n > x$ довольно резко убывают с ростом порядка n при фиксированном x (рис. 1,г). Предлагается нормировать функции Бесселя $I_n(x)$ на их максимальные значения на границах цилиндрических зон реактора, так чтобы все $I_n^{\text{норм}}(x)$ при любом n были равны 1 на внешней границе в каждой из зон ($x = x_{\text{max}}$):

$$I_n^{\text{норм}}(x) = \frac{I_n(x) / I_0(x)}{I_n(x_{\text{max}}) / I_0(x_{\text{max}})} * \frac{I_0(x)}{I_0(x_{\text{max}})} \quad (13)$$

Расчет нормированных функций $I_n^{\text{норм}}(x)$ в программе начинается для нулевого и первого порядков. Если при этом значения аргументов x и x_{max} меньше 20, то расчет $I_0(x)$, $I_1(x)$ и $I_0(x_{\text{max}})$, $I_1(x_{\text{max}})$ производится по соответствующим подпрограммам $BESI0(z)$ и $BESI1(z)$, имеющимся в общественной библиотеке ЭВМ. Тогда соотношение (13) принимает вид

$$I_0^{\text{норм}}(x) = \frac{I_0(x)}{I_0(x_{\text{max}})}, \quad I_1^{\text{норм}}(x) = \frac{I_1(x)}{I_1(x_{\text{max}})}$$

Если же значения аргументов x и x_{max} больше 20, то вычисление вышеуказанных функций производится по соответствующим асимптотическим разложениям [3]:

$$I_0(z) = e^z * FS(0, z), \quad I_1(z) = e^z * FS(1, z), \quad (14)$$

где

$$FS(0, z) = \frac{1}{\sqrt{2\pi z}} * \left(1 + \frac{1^2}{8z} + \frac{1^2 \cdot 3^2}{2! \cdot (8z)^2} + \frac{1^2 \cdot 3^2 \cdot 5^2}{3! \cdot (8z)^3} + \dots \right);$$

$$FS(1, z) = \frac{1}{\sqrt{2\pi}z} * \left(1 - \frac{1.3}{8z} - \frac{1^2 \cdot 3 \cdot 5}{2! \cdot (8z)^2} - \frac{1^2 \cdot 3^2 \cdot 5 \cdot 7}{3! \cdot (8z)^3} - \dots \right).$$

В этом случае расчет функций $I_0^{\text{норм}}(x)$ и $I_1^{\text{норм}}(x)$ происходит следующим образом:

$$I_0^{\text{норм}}(x) = e^{(x-x_{\text{max}})} * \frac{FS(0, x)}{FS(0, x_{\text{max}})},$$

$$I_1^{\text{норм}}(x) = e^{(x-x_{\text{max}})} * \frac{FS(1, x)}{FS(1, x_{\text{max}})}.$$

Если аргумент x_{max} лежит в области значений больше 20, а величина x меньше 20, то вычисление нормированных функций

$I_n^{\text{норм}}(x)$ осуществляется по формулам

$$I_0^{\text{норм}}(x) = e^{-x_{\text{max}}} * \frac{I_0(x)}{FS(0, x_{\text{max}})},$$

$$I_1^{\text{норм}}(x) = e^{-x_{\text{max}}} * \frac{I_1(x)}{FS(1, x_{\text{max}})}.$$

Расчет нормированных функций $I_n^{\text{норм}}(x)$ более высоких порядков ($n \geq 2$) производится по формуле (13), при этом функции $I_n(x)/I_0(x)$ и $I_n(x_{\text{max}})/I_0(x_{\text{max}})$ могут быть, например, определены из рекуррентного соотношения [3]

$$\frac{I_{m+1}(z)}{I_0(z)} = \frac{I_{m-1}(z)}{I_0(z)} - \frac{2m}{z} * \frac{I_m(z)}{I_0(z)}. \quad (15)$$

В частности, при $m = 1$ имеем

$$\frac{I_2(z)}{I_0(z)} = 1 - \frac{2 \cdot 1}{z} * \frac{I_1(z)}{I_0(z)}.$$

Однако при $z \approx m$ расчет по формуле (15) приводит к большим погрешностям в определении $I_{m+1}(z)/I_0(z)$. Для определения более точной границы применимости рекуррентного соотношения (с приемлемой погрешностью) были проведены необходимые вычисления значений функций $I_n(z)/I_0(z)$ по формуле (15) и с помощью разложения в ряд [4]

$$I_n(x) = \frac{(x/2)^n}{\Gamma(n+1)} * BFI(n, x), \quad (16)$$

где

$$BFI(n, x) = \left[1 + \frac{(x/2)^2}{1!(n+1)} + \frac{(x/2)^4}{2!(n+1)(n+2)} + \frac{(x/2)^6}{3!(n+1)(n+2)(n+3)} + \dots \right].$$

В табл. 1 приводятся результаты вычислений функций $I_n(x)/I_0(x)$ в зависимости от порядка n для аргумента $x = 18$. Как показали расчеты, различие между точными результатами вычислений (ряд) и полученными по рекуррентному соотношению не превосходит одной тысячной процента, если значения функций $I_n(x)/I_0(x)$ по порядку величины не меньше $10^{-3} - 10^{-4}$. Поскольку на вычисления с помощью ряда тратится больше машинного времени, предлагается комбинированный способ расчета – по рекуррентному соотношению при $I_n(x)/I_0(x) > 5 \cdot 10^{-4}$ и далее с использованием формулы (16).

Таким образом, если в формуле (13) числитель и знаменатель первого сомножителя становятся меньше или равны $5 \cdot 10^{-4}$, расчет функций $I_n^{норм}(x)$ производится следующим образом:

$$I_n^{норм}(x) = \left(\frac{x}{x_{max}} \right)^n * \frac{BFI(n, x)}{BFI(n, x_{max})}$$

Таблица 1

Результаты вычислений значений функций $I_n(x)/I_0(x)$ с помощью ряда и по рекуррентному соотношению в зависимости от порядка n для аргумента $X = 18$

n	$F_1 = I_n(x)/I_0(x)$ (ряд)	$F_2 = I_n(x)/I_0(x)$ (рекур. соотн.)	$\frac{F_1 - F_2}{F_2}$, %
0	1,0000	1,0000	0,000
1	$9,7181 \cdot 10^{-1}$	$9,7181 \cdot 10^{-1}$	0,000
2	$8,9202 \cdot 10^{-1}$	$8,9202 \cdot 10^{-1}$	0,000
3	$7,7358 \cdot 10^{-1}$	$7,7358 \cdot 10^{-1}$	0,000
4	$6,3415 \cdot 10^{-1}$	$6,3415 \cdot 10^{-1}$	0,000
5	$4,9173 \cdot 10^{-1}$	$4,9173 \cdot 10^{-1}$	0,000
6	$3,6097 \cdot 10^{-1}$	$3,6097 \cdot 10^{-1}$	0,000
7	$2,5109 \cdot 10^{-1}$	$2,5109 \cdot 10^{-1}$	0,000
8	$1,6567 \cdot 10^{-1}$	$1,6567 \cdot 10^{-1}$	0,000
9	$1,0382 \cdot 10^{-1}$	$1,0382 \cdot 10^{-1}$	0,000
10	$6,1858 \cdot 10^{-2}$	$6,1858 \cdot 10^{-2}$	0,000
11	$3,5088 \cdot 10^{-2}$	$3,5088 \cdot 10^{-2}$	0,000
12	$1,8972 \cdot 10^{-2}$	$1,8972 \cdot 10^{-2}$	0,000
13	$9,7915 \cdot 10^{-3}$	$9,7915 \cdot 10^{-3}$	0,000
14	$4,8293 \cdot 10^{-3}$	$4,3293 \cdot 10^{-3}$	0,000
15	$2,2792 \cdot 10^{-3}$	$2,2792 \cdot 10^{-3}$	0,000
16	$1,0306 \cdot 10^{-3}$	$1,0306 \cdot 10^{-3}$	0,000
17	$4,4703 \cdot 10^{-4}$	$4,4703 \cdot 10^{-4}$	0,000
18	$1,8622 \cdot 10^{-4}$	$1,8622 \cdot 10^{-4}$	0,001
19	$7,4585 \cdot 10^{-5}$	$7,4594 \cdot 10^{-5}$	-0,008
20	$2,8757 \cdot 10^{-5}$	$2,8742 \cdot 10^{-5}$	0,052
21	$1,0684 \cdot 10^{-5}$	$1,0722 \cdot 10^{-5}$	-0,364
22	$3,8286 \cdot 10^{-6}$	$3,7229 \cdot 10^{-6}$	2,761
23	$1,3248 \cdot 10^{-6}$	$1,6222 \cdot 10^{-6}$	-22,447

Как только $I_n(x) / I_0(x)$ будет меньше $5 \cdot 10^{-4}$, расчетная формула будет иметь вид

$$I_n^{\text{норм}}(x) = \frac{\frac{(x/2)^n}{\Gamma(n+1)} * \frac{BFI(n, x)}{I_0(x)}}{I_n(x_{\text{max}}) / I_0(x_{\text{max}})} * \frac{I_0(x)}{I_0(x_{\text{max}})}$$

Производные от функций $I_n^{\text{норм}}(x)$ рассчитываются по формулам:

при $\frac{I_n(x)}{I_0(x)} > 5 \cdot 10^{-4}$ и $\frac{I_n(x_{\text{max}})}{I_0(x)} > 5 \cdot 10^{-4}$

$$\dot{I}_n^{\text{норм}}(x) = I_n^{\text{норм}}(x) * \left[\frac{I_{n+1}(x) / I_0(x)}{I_n(x) / I_0(x)} + \frac{n}{x} \right] * \dot{x};$$

при $\frac{I_n(x)}{I_0(x)} \leq 5 \cdot 10^{-4}$ и $\frac{I_n(x_{\text{max}})}{I_0(x)} \leq 5 \cdot 10^{-4}$

$$\dot{I}_n^{\text{норм}}(x) = \left(\frac{x}{x_{\text{max}}} \right)^{n-1} * \frac{BFI(n-1, x) + \frac{x/2}{n} * \frac{x/2}{n+1} * BFI(n+1, x)}{\frac{x_{\text{max}}/2}{n} * BFI(n, x_{\text{max}})} * \frac{\dot{x}}{2};$$

при $\frac{I_n(x)}{I_0(x)} \leq 5 \cdot 10^{-4}$ и $\frac{I_n(x_{\text{max}})}{I_0(x)} > 5 \cdot 10^{-4}$

$$\dot{I}_n^{\text{норм}}(x) = \frac{\left[\frac{(x/2)^{n-1}}{\Gamma(n)} * BFI(n-1, x) + \frac{(x/2)^{n+1}}{\Gamma(n+2)} * BFI(n+1, x) \right] * \frac{1}{I_0(x)}}{I_n(x_{\text{max}}) / I_0(x_{\text{max}})} * \frac{I_0(x)}{I_0(x_{\text{max}})} * \frac{\dot{x}}{2}.$$

Процедура расчета $\dot{I}_n^{\text{норм}}(x)$ аналогична выше рассмотренной при вычислении функций $I_n^{\text{норм}}(x)$.

1.5. Произведение бесселевых функций первого и второго рода $J_n(z) * Y_n(z)$ действительного аргумента ($z > z$)

Комплексы такого вида возникают в реакторных гетерогенных задачах при использовании теоремы сложения для бесселевых функций в случае перехода от системы координат с полюсами в центре блоков к системе координат с полюсом в центре реактора. Как уже отмечалось, функции $|Y_n(x)|$ с ростом порядка n при $n > x$ сильно возрастают, а функции $J_n(x)$ убывают при фиксированном аргументе x , в то время как их произведение изменяется более медленно. Предлагается следующая схема расчета вышеуказанного комплекса с ростом индекса n .

Функции Бесселя нулевого и первого порядков рассчитываются с помощью соответствующих стандартных подпрограмм, имеющихся в общественной библиотеке ЭВМ. Функции Бесселя второго рода $Y_n(z)$ при $n \geq 2$ определяются из рекуррентного соотношения (7), а функции первого рода $J_n(z)$ из соотношения (11) (разложение в ряд). Если в процессе счета при некотором индексе n значения функций $Y_n(z)$ становятся больше наперед заданного числа (в программе 10^6), то они автоматически умножаются на числовой множитель 10^{-6} , а для того чтобы произведение $J_n(z) * Y_n(z)$ не изменилось, функции $J_n(z)$ соответственно умножаются на коэффициент 10^6 . Такая нормировка в процессе вычислений позволяет избежать останова машины из-за переполнения при расчете функций $Y_n(z)$ и потери точности вследствие уменьшения значений функций $J_n(z)$ с ростом порядка n ("машинный" нуль).

Аналогичным образом производится расчет комплексов вида $J_n(z) * Y_n(z)$ или $J_n(z) * Y_n(z)$, возникающих при использовании краевых условий на границах между зонами.

1.6. Произведение бесселевых функций первого и второго рода $I_n(z) * K_n(z)$ мнимого аргумента ($z > z$)

Функции Бесселя второго рода $K_n(x)$ с ростом порядка n неограниченно возрастают, особенно сильно в области малых аргументов, а функции $I_n(x)$ в этой области определения довольно резко убывают. При больших значениях аргумента x функции $I_n(x)$ могут достигать значительных величин, в то время как функции $K_n(x)$ убывают по экспоненциальному закону с ростом x . Принимая во внимание такой характер поведения вышеуказанных функций, предлагается рассчитывать комплексы $I_n(z) * K_n(z)$ при $n \geq 2$ непосредственно через их произведения, вычисленные для нулевого, первого и последующих порядков. Расчет функций $I_n(z)$ и $K_n(z)$ при $n = 0, 1$ производится либо с помощью стандартных подпрограмм, имеющихся в общественной библиотеке ЭВМ, либо через соответствующие асимптотические разложения при больших значениях аргумента. Для вычисления комплекса $I_n(z) * K_n(z)$ при $n \geq 2$ берутся соответствующие рекуррентные соотношения для функций $I_n(z)$ и $K_n(z)$ и умножаются друг на друга; в итоге получаем следующую рабочую формулу:

$$K_{m+1}(z) * I_{m+1}(z) = K_{m-1}(z) * I_{m-1}(z) + \frac{2m}{z} * K_m(z) * I_{m-1}(z) - \frac{2m}{z} * I_m(z) * K_{m-1}(z) - \frac{4m^2}{z * z} * K_m(z) * I_m(z) . \quad (17)$$

В частности, при $m = 1$ имеем

$$K_2(z) * I_2(z) = K_0(z) * I_0(z) + \frac{2 \cdot 1}{z} * K_1(z) * I_0(z) - \frac{2 \cdot 1}{z} * I_1(z) * K_0(z) - \frac{4 \cdot 1^2}{z * z} * K_1(z) * I_1(z) .$$

Таблица 2

Результаты вычислений значений функций $K_n(z) * I_n(z)$
в зависимости от порядка n ($Z = 18, z = 18$)

n	$F_1 = K_n(z) * I_n(z) * 10^2$ ($I_n(z)$ через ряд)	$F_2 = K_n(z) * I_n(z) * 10^2$ ($I_n(z)$ из рекур. соотн.)	$\frac{F_1 - F_2}{F_1} \%$
0	2,7789	2,7789	0,000
1	2,7745	2,7745	0,000
2	2,7618	2,7618	0,000
3	2,7409	2,7409	0,000
4	2,7124	2,7124	0,000
5	2,6770	2,6770	0,000
6	2,6356	2,6356	0,000
7	2,5892	2,5892	0,000
8	2,5384	2,5384	0,000
9	2,4845	2,4845	0,000
10	2,4281	2,4281	0,000
11	2,3701	2,3701	0,000
12	2,3110	2,3110	0,000
13	2,2516	2,2516	0,000
14	2,1924	2,1924	0,000
15	2,1337	2,1337	0,000
16	2,0758	2,0758	0,000
17	2,0192	2,0192	0,000
18	1,9639	1,9638	0,002
19	1,9101	1,9103	-0,010
20	1,8580	1,8568	0,063
21	1,8075	1,8156	-0,447
22	1,7588	1,6992	3,387
23	1,7118	2,1831	-27,532

Однако как показали проведенные вычисления (табл. 2), расчет по формуле (17) приводит к большим погрешностям в определении комплексов $K_n(z) * I_n(z)$, начиная с некоторого порядка n . Это обусловлено тем, что применение рекуррентного соотношения для вычисления $I_n(z)$ при $n \gg z$, как это было показано в разд. 1.4, становится неоправданным. Поэтому в программе, как только $I_n(z)/I_0(z)$ становится меньше $5 \cdot 10^{-4}$, вычисление $I_n(z)$ производится с помощью ряда (16), а $K_n(z)$ находится из рекуррентного соотношения (3). Если в процессе счета при некотором индексе n значения функций $K_n(z)$ становятся больше наперед заданного числа (в программе 10^6), то они автоматически умножаются на числовой множитель 10^{-6} , а для того чтобы произведение $K_n(z) * I_n(z)$ не изменилось, функции $I_n(z)$ умножаются соответственно на коэффициент 10^6 . Такая нормировка, как уже отмечалось выше, позволяет избежать останова машины из-за переполнения при расчете функций $K_n(z)$ и потери точности вследствие убывания функций $I_n(z)$ с ростом порядка n ("машинный" нуль).

Аналогичным образом производится расчет комплексов вида $I_n(z) * \dot{K}_n(z)$ и $\dot{I}_n(z) * K_n(z)$, возникающих при использовании краевых условий на границах раздела между зонами.

2. ПОИСК ГЛАВНОГО СОБСТВЕННОГО ЗНАЧЕНИЯ $K_{эф}$

В работе [1] на решение для потока нейтронов или сопряженных функций последовательно накладываются граничные условия на поверхности реактора, границах раздела между зонами и затем на по-

верхности блоков. В конечном итоге после проведения соответствующих преобразований получаем систему линейных однородных алгебраических уравнений относительно неизвестных, входящих в нерегулярное решение, обусловленное наличием блоков в реакторе.

Для многогруппового диффузионного уравнения [5] всегда существует положительное главное собственное значение, являющееся по физическому смыслу эффективным коэффициентом размножения $K_{эф}$, которое может быть найдено из равенства нулю определителя вышеуказанной системы однородных алгебраических уравнений. Следует отметить, что могут существовать и другие собственные значения K , меньшие по величине, чем $K_{эф}$, для которых определитель DET вышеуказанной системы уравнений будет также равен нулю, что, естественно, усложняет процедуру определения $K_{эф}$. К тому же функция $DET = f(K_{эф})$ не является монотонной и непрерывной в области задания, и поэтому при поиске главного собственного значения (первого корня) возникают дополнительные трудности, вызванные наличием конечных и бесконечных разрывов. Так как ход функции $DET = f(K_{эф})$ зависит от конкретного типа рассматриваемой задачи, в программе **PNK** предусмотрено несколько способов определения первого корня, который в дальнейшем обозначается как $K_{эф}^{max}$, что дает возможность пользователю выбрать оптимальный вариант поиска с учетом специфики изучаемой задачи.

2.1. Автоматический поиск корня

Процедура поиска корня выглядит следующим образом. Первоначально производится расчет определителя $DET^{нач}$ при $K_{эф}^{нач}$, а затем при $K_{эф}^{(1)} = K_{эф}^{нач} - DKEF$, где $DKEF$ есть шаг подбора. Определяется наклон кривой $DET = f(K_{эф})$ и в зависимости от этого дальнейший поиск корня в программе идет в одном из направлений: $K_{эф}^{(i+1)} = K_{эф}^{(i)} - DKEF$ или $K_{эф}^{(i+1)} = K_{эф}^{(i)} + DKEF$, пока знаки определителей при $K_{эф}^{(i+1)}$ и $K_{эф}^{(i)}$ не станут разными.

Если функция $DET = f(K_{эф})$ является непрерывной в интервале $[K_{эф}^{(l+1)}, K_{эф}^{(l)}]$, то в указанной области содержится,

по крайней мере, один из корней задачи, в противном случае исследуемая функция может иметь на этом интервале разрыв. Подробно этот вопрос рассмотрен далее, в подразд. 2.3.

Необходимо обратить внимание на то, что рассматриваемый способ определения главного собственного значения задачи $K_{эф}^{max}$ не всегда приводит к желаемому результату. Вследствие немонотонного поведения функции $DET = f(K_{эф})$ заданное значение $K_{эф}^{нач}$ может оказаться не вполне удачным и поэтому корень может быть вообще не найден или же в процессе подбора будет получен другой корень (рис. 2, а, б).

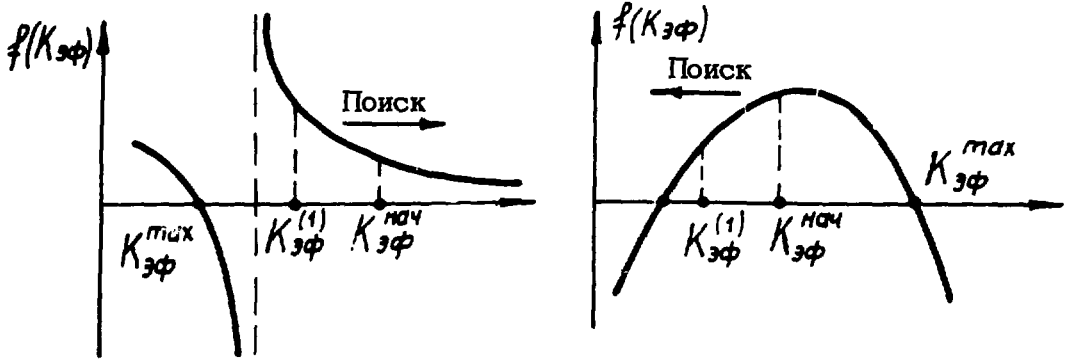
Автоматический способ поиска корня целесообразно применять при расчете небольших возмущений, вносимых в некоторый базовый вариант изучаемой системы, для которой эффективный коэффициент размножения был уже определен из предыдущих вычислений ($K_{эф}^{нач}$ задается вблизи этого значения). Применение данного способа может оказаться также полезным при исследовании определенного типа задач, для которых можно предположить, что функция $DET = f(K_{эф})$ имеет монотонный характер в широком диапазоне изменения $K_{эф}$, т.е. корни лежат достаточно далеко друг от друга и отсутствуют особые точки (гетерогенные реакторы небольших размеров).

2.2. Поиск в заданном направлении от $K_{эф}^{нач}$

2.2.1. Поиск в сторону уменьшения $K_{эф}$ от $K_{эф}^{нач}$ (рис. 2, в)

Поиск корня происходит следующим образом. Первоначально производится расчет определителя при $K_{эф}^{нач}$, затем при $K_{эф}^{(1)} = K_{эф}^{нач} - DK_{эф}$, потом при $K_{эф}^{(2)} = K_{эф}^{нач} - 2 * DK_{эф}$ и так далее, пока знаки определителей при $K_{эф}^{(i+1)}$ и $K_{эф}^{(i)}$ не станут разными. После этого в программе исследуется поведение функции $DET = f(K_{эф})$

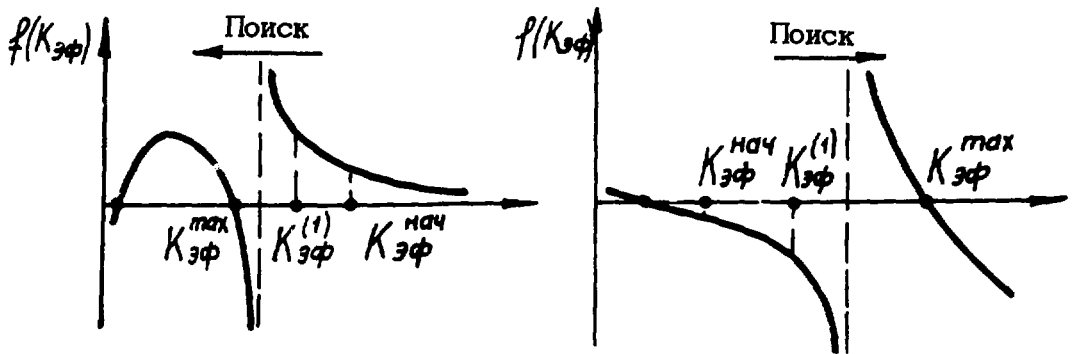
Автоматический поиск корня



а) Корень не найден

б) Будет найден 2-й корень

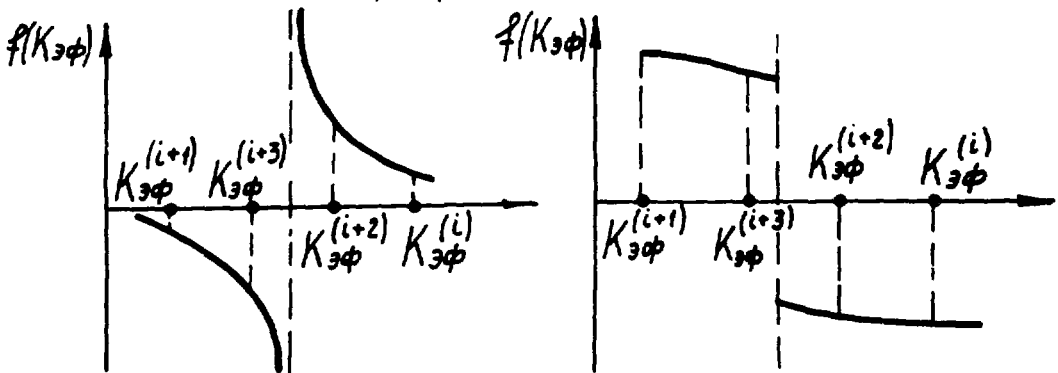
Поиск корня в заданном направлении



в) В направлении уменьшения от $K_{эф}^{нач}$

г) В направлении возрастания от $K_{эф}^{нач}$

В интервале $[K_{эф}^{(i)}, K_{эф}^{(i+1)}]$ есть особая точка



д) Бесконечный разрыв

е) Конечный разрыв

Рис. 2. Схема поиска корня

в интервале $[K_{эф}^{(i+1)}, K_{эф}^{(i)}]$; подробно этот вопрос рассмотрен в подразд. 2.3.

Подбор $K_{эф}^{max}$ сверху наиболее предпочтительно применять на начальном этапе исследований, когда реакторная система изучена недостаточно хорошо. В этом случае целесообразно задать начальное значение $K_{эф}$ с некоторым превышением по отношению к ожидаемой величине ($K_{эф}^{нач} \leq K_{\infty}$), а шаг разбиения не очень большим. Метод поиска корня в заданном направлении нецелесообразно использовать для расчета небольших возмущений неизвестного знака, вносимых в некоторый базовый вариант. В подобных случаях следует применять способ автоматического поиска корня.

2.2.2. Поиск корня в сторону возрастания $K_{эф}$ от $K_{эф}^{нач}$ (рис. 2, г).

Процедура поиска корня следующая. Производится расчет определителя при $K_{эф}^{нач}$, затем при $K_{эф}^{(1)} = K_{эф}^{нач} + DKEF$, потом при $K_{эф}^{(2)} = K_{эф}^{нач} + 2 * DKEF$ и так далее, пока знаки определителей при $K_{эф}^{(i+1)}$ и $K_{эф}^{(i)}$ не станут разными. После этого в программе исследуется поведение функции $DET = f(K_{эф})$ в интервале $[K_{эф}^{(i+1)}, K_{эф}^{(i)}]$; подробно этот вопрос рассмотрен в подразд. 2.3.

Данный способ поиска корня предпочтительно использовать в тех случаях, когда заранее известно, что возмущение, вносимое в реакторную систему, приводит к возрастанию $K_{эф}$. Например, этот эффект имеет место при извлечении поглощающего стержня из реактора, для которого эффективный коэффициент размножения со всеми стержнями был известен из предыдущих вычислений. Однако если в систему вносится достаточно большое возмущение, как, например, при извлечении всех стержней, то в случае близкого расположения корней в решении (реакторы больших размеров) появляется вероятность определения не $K_{эф}^{max}$, а другого корня; тогда целесообразно использовать предыдущий способ поиска собственного значения.

2.3. Поведение функции $DET = f(K_{эф})$

в интервале изменения ее знака

Для того чтобы выяснить характер поведения функции $DET = f(K_{эф})$ в исследуемом интервале $[K_{эф}^{(i+1)}, K_{эф}^{(i)}]$, производится расчет еще двух ее значений внутри этого интервала:

$$DET^{(i+2)} = f(K_{эф}^{(i)} - 0,5 * DKEF) ;$$

$$DET^{(i+3)} = f(K_{эф}^{(i+1)} + 0,5 * DKEF) .$$

Дальнейший анализ в программе идет по следующей схеме:

а) Если полученные значения функции в указанном интервале не убывают, то функция $DET = f(K_{эф})$ является немонотонной в этой области и считается, что она имеет особенность на данном интервале (рис. 2,д). Затем в программе начинается поиск следующей области, где функция меняет знак, причем в том направлении изменения $K_{эф}$, которое было задано пользователем или найдено в процессе счета при автоматическом способе определения корня. Следует отметить, что если какой-нибудь из определителей $DET^{(i)}$ в исследуемом интервале превысит величину 10^{19} , то авоста в процессе счета не происходит (выдается только стандартная диагностика о переполнении).

б) Если полученные значения функции убывают в исследуемом интервале, то в программе производится дополнительное исследование для выявления конечных разрывов (рис. 2,е). При наличии такой особенности дальнейший анализ в данном интервале прекращается и начинается поиск следующей области, где функция $DET = f(K_{эф})$ меняет знак.

в) Если исследуемая функция оказывается монотонной на интервале $[K_{эф}^{(i+1)}, K_{эф}^{(i)}]$, то начинается поиск корня одним из методов

последовательного приближения. В программе предусмотрены три подхода для поиска корня: метод половинного деления, метод хорд и их комбинация. В качестве базового варианта используется комбинация метода хорд и метода половинного деления. Подбор заканчивается тогда, когда полученные в процессе итераций значения $K_{эф}$, лежащие по разные стороны корня, не будут отличаться друг от друга с заданной степенью точности **EKEF**.

Необходимо еще раз подчеркнуть, что ввиду сложного характера поведения функции $DET = f(K_{эф})$ в ее области задания, а также наличия многих корней пользователь должен внимательно подходить к полученным результатам. Вышеуказанные замечания особенно важны при изучении реакторных систем больших размеров, где корни решения могут быть расположены недалеко друг от друга и вблизи точек разрывов.

2.4. Анализ некоторых особых случаев

Уже отмечалось, что процесс поиска корня начинается во всех случаях с расчета определителя $DET^{нач}$ при $K_{эф}^{нач}$, и величина определителя нормируется таким образом, чтобы она не выходила за пределы заданного диапазона **DETMIN**, **DETMAX**. Затем производится расчет определителя $DET^{(1)}$ при $K_{эф}^{(1)}$ по одному из вышеуказанных способов поиска. Затем в программе производится анализ полученного значения $DET^{(1)}$.

2.4.1. Особая точка

Если полученное значение $DET^{(1)}$ при $K_{эф}^{(1)}$ превосходит 10^{19} , то выдается стандартная диагностика о переполнении. Затем автоматически происходит изменение аргумента от особой точки на величину шага $0,1 * DKEF$, последующий расчет функции $DET = f(K_{эф})$ при новом значении аргумента и т.д. Процесс заканчивается тогда, когда полученное значение функции $DET^{(i)}$ станет меньше, чем 10^{19} . После этого в зависимости от выбранного спо-

соба подбора происходит дальнейший поиск интервала, где функция $DET = f(K_{эф})$ меняет знак (рис. 2,а,в,г).

Следует отметить, что вышеуказанный порядок операций сохраняется и в том случае, если в точке разрыва исследуемая функция не меняет свой знак или переполнение в процессе поиска указанного интервала имеет место при любом значении аргумента, а не только при $K_{эф}^{(1)}$.

2.4.2. "Машинный" нуль

Если полученное значение $DET^{(1)}$ при $K_{эф}^{(1)}$ будет меньше, чем 10^{-19} ("машинный" нуль), в программе происходит изменение аргумента от этой точки на величину шага $0,1 * DKEF$, направление которого зависит от используемого способа поиска корня. Если применяется автоматический способ поиска корня, то значение исследуемой функции определяется при $K_{эф}^{(2)} = K_{эф}^{(1)} - 0,1 * DKEF$. В случае использования метода поиска в заданном направлении изменение аргумента происходит в ту сторону, которая задается пользователем. Процесс заканчивается тогда, когда полученное значение функции $DET = f(K_{эф})$ станет больше, чем 10^{-18} . После этого продолжается дальнейший анализ поведения исследуемой функции в соответствии с выбранным способом поиска корня. Если значение функции стало меньше 10^{-19} не при $K_{эф}^{(1)}$, а при любом другом значении $K_{эф}$, изменение аргумента автоматически происходит в том направлении, в котором происходит в данный момент исследование функции $DET = f(K_{эф})$.

2.5. Нормировка функции $DET = f(K_{эф})$ в процессе поиска корня

Поиск корня начинается с расчета значения исследуемой функции (определителя) при $K_{эф}^{нач}$. Величина определителя $DET^{нач}$ в зависимости от типа задачи может колебаться в широких пределах и в некоторых случаях выходит за пределы диапазона нормализованных чи-

сел, допускаемых на БЭСМ-6 (10^{-19} и 10^{19}). Поэтому в программе предусмотрена автоматическая нормировка исследуемой функции с помощью некоторого специального множителя $AMATR$, значение которого задается первоначально равным нулю в исходных данных. Множитель $AMATR$ в программе подбирается автоматически таким образом, чтобы полученное значение $DET^{нач}$ не выходило за пределы диапазона $DETMIN$ и $DETMAX$. Величины в программе приняты равными: $DETMIN = 1 \cdot 10^{-10}$ и $DETMAX = 1 \cdot 10^{14}$.

Даже если начальное значение $AMATR$ будет задано пользователем "неудачно", так, что значение функции превысит 10^{19} , авста при счете все равно не происходит. Выдается стандартная диагностика о переполнении, а величина множителя автоматически снижается до тех пор, пока значение функции $DET^{нач}$ не станет меньше $DETMAX$. Следует отметить, что в процессе исследования поведения функции $DET = f(K_{эф})$ ее нормировка производится только при $K_{эф}^{нач}$ и в случае поиска корня в исследуемом интервале $[K_{эф}^{(i)}, K_{эф}^{(i+1)}]$, если только значение определителя становится меньше 10^{-19} .

3. НОРМИРОВКА ПОТОКОВ НЕЙТРОНОВ

Полученное в работе [1] решение для потоков нейтронов нормируется на единичную мощность в соответствии со следующим соотношением:

$$\left\{ \int_{S_{LHET}} \left| \frac{\sum_{f_1}^{LHET} \sum_{f_2}^{LHET} \Phi_1^{LHET}}{\Phi_2^{LHET}} \right| ds - \sum_{p=1}^M \int_{S_p} \left| \frac{\sum_{f_1}^{LHET} \sum_{f_2}^{LHET} \Phi_1^{LHET}}{\Phi_2^{LHET}} \right| ds_p^+ + 2\pi \sum_{N=1}^{LZ} \int_R^{R_N} \left| \frac{\sum_{f_1}^N \sum_{f_2}^N \Phi_1^N}{\Phi_2^N} \right| r dr + \sum_{p=1}^{M^*} 2\pi a_p (\bar{q}_p \bar{y}_p^-) \right\} * NORM = 1$$

- Здесь
- S_{LHET} - полная площадь зоны, где расположены блоки;
 - S_p - площадь, занятая одним блоком p ;
 - M - число блоков в зоне;
 - M^* - число размножающих блоков в зоне;
 - $LHET$ - номер зоны с блоками;
 - N - номера зон ($1-LZ$), $\delta_N = 1$ при $N \neq LHET$,
 $\delta_N = 0$ при $N = LHET$;
 - R_{N-1} - внутренний радиус зоны N ;
 - R_N - внешний радиус зоны N ;
 - a_p - геометрический радиус блока p ;
 - \vec{q}_p - вектор энерговыделения p -го блока;
 - \vec{j}_p - вектор тока нейтронов, падающего на поверхность блока p ;
 - \sum_{f1}^N, \sum_{f2}^N - сечения деления в соответствующих зонах в первой и второй группах;
 - $NORM = E_f H_{A3} / K_2$ - нормирующий сомножитель;
 - E_f - энергия деления (в программе принято значение 200 МэВ);
 - H_{A3} - высота активной зоны реактора, см;
 - $K_2 = \frac{Q_{max}}{\bar{Q}_2}$ - коэффициент неравномерности энерговыделения Q_2 по высоте активной зоны реактора H_{A3} .

Отметим, что в указанное соотношение входят суммы от частных решений уравнения для потоков нейтронов, обусловленных наличием блоков в системе (нерегулярное решение). Поэтому в условии нормировки вклад каждого блока связан с частным решением, относящимся к данному блоку p , минус вклады по всем частным решениям, проинтегрированным по площади, занимаемой блоком p . Приведенное условие нормировки включает также и случай гомогенного реактора. В связи с этим потоки нейтронов в программе **PNK** выдаются отнормированными на мощность 1 кВт.

Авторы признательны Е. С. Глушкову и Г. И. Бирюх за ценные критические замечания.

Л и т е р а т у р а

1. Носов В.И., Компаниец Г.В., Петрушенко Р.П. Препринт ИАЭ-2867, М., 1977.
2. Глушков Е.С., Пономарев-Степной Н.Н., Петушкова Н.А. Препринт ИАЭ-1938, М., 1969.
3. Ватсон Г.Н. Теория бесселевых функций. М., ИЛ, -1949.
4. Янке Е., Эмде Ф., Лёш Ф. Специальные функции. М., "Наука", 1977.
5. Белл Д., Глесстон С. Теория ядерных реакторов. М., Атомиздат, 1974.

Редактор Л. А. Кузьмина

Технический редактор Н. И. Мазаева

Корректор В. П. Горячева

Т-11914. 18.06.79. Формат 60х90/16. Уч.-изд. л. 1,50
Тираж 170. Заказ 1113. Цена 15 коп. Индекс 3624. ИАЭ

