

**Решение
дифференциальных
уравнений. Метод
прогонки.**

- Прогонкой называется модификация метода Гаусса для решения систем линейных алгебраических уравнений с трехдиагональной матрицей. Если матрица системы обладает определенными свойствами, то метод прогонки является численно устойчивым и очень эффективным методом, который позволяет практически мгновенно решать одномерные краевые задачи, одну из которых мы рассмотрели в предыдущем разделе. Большинство корректно поставленных физических задач приводит к системе уравнений с хорошей матрицей, и в этих случаях метод прогонки проявляет слабую чувствительность как к погрешностям задания начальных условий, так и к погрешностям вычислительного характера.

Суть метода прогонки.

- Суть метода прогонки заключается в том, что, используя специфику структуры матрицы системы уравнений (наличие трех диагоналей), удастся получить рекуррентные формулы для вычисления последовательности коэффициентов прогонки, которые позволяют на обратном ходу вычислить значения функции в узлах сетки. Рассматривая конечно-разностное уравнение для первой тройки узлов:
- $b_1U_1 + c_1U_2 = -a_1U_0,$
- видим, что оно связывает между собой два соседних значения U_1 , и U_2 . Перепишем его в виде:
- $d_1U_2 + e_1 = U_1, (1)$
- где d_1 и e_1 вычисляются по известным значениям. Наблюдательный читатель заметит, что это справедливо только для задач первого рода. Чуть позже мы получим общее решение. Теперь мы можем исключить U_1 из уравнения для следующей тройки узлов:
- $a_2U_1 + b_2U_2 + c_2U_3 = f_2,$
- подставив значение U_1 из уравнения (1). После этой процедуры последнее уравнение также может быть приведено к виду:
- $d_2U_3 + e_2 = U_2,$
- Подстановки можно продолжать и дальше, но для получения рекуррентного соотношения, достаточно рассмотреть одну из них для произвольного индекса i . Подставив
- $d_{i-1}U_i + e_{i-1} = U_{i-1},$
- в уравнение
- $a_iU_{i-1} + b_iU_i + c_iU_{i+1} = f_i,$
- получим:
- $U_i = -[c_iU_{i+1}/(a_i d_{i-1} + b_i)] + [f_i - a_i e_{i-1}/(a_i d_{i-1} + b_i)] (2)$
- Это соотношение дает две рекуррентные формулы для коэффициентов:
- $d_i = -c_i / (a_i d_{i-1} + b_i) (3)$
- $e_i = (f_i - a_i e_{i-1}) / (a_i d_{i-1} + b_i) (4)$

Суть метода прогонки.

- Цикл вычисления последовательности коэффициентов в соответствии с этими формулами носит название прямого хода прогонки.
- **$d_0 = y_0, e_0 = b_0,$**
- Цикл прямого хода повторяется $N-1$ раз. Последними будут вычислены коэффициенты d_{N-1} и e_{N-1} , которые связывают функции в двух узлах вблизи правой границы:
- **$U_{n-1} = d_{n-1}U_n + e_{n-1} \quad (5)$**
- Если на правой границе задано условие первого рода $U_n = c$, то уже можно вычислить U_{n-1} и далее продолжать обратный ход прогонки при $i = N - 1, \dots, 1, 0$. Если условие более сложное, то надо рассмотреть уравнение (6), определяющее граничное условие на правой границе. Напомним его:
- **$U_n = y_n U_{n-1} + b_n \quad (6)$**
- Соотношения (6) и (5) составляют систему из двух уравнений с двумя неизвестными. Используя определители, запишем ее решение.
- **$U_{n-1} = (e_{n-1} + b_n d_{n-1}) / (1 - y_n d_{n-1}) \quad (7)$**
- **$U_n = (b_n + y_n e_{n-1}) / (1 - y_n d_{n-1})$**
- Таким образом, мы нашли значения в двух узлах, лежащих вблизи правой границы расчетной области. Теперь, используя формулу (2) и уменьшая индекс i от $N = 2$ до 0 , можно вычислить все неизвестные U_i . Этот процесс носит название обратного хода прогонки. Почему-то в голову приходит лозунг нашего времени: «Цели ясны, задачи определены».

Теоретическая часть

- Пусть $Ax=b$, где A – трехдиагональная матрица. Матрица $A=[a_{ij}]$ называется $(2m+1)$ – диагональной, если $a_{ij}=0$ при $|i-j|>m$.
- Для решения систем уравнений такого вида часто наиболее целесообразно применять метод Гаусса при естественном порядке исключения неизвестных. В случае, когда этот метод применяется для решения СЛАУ, его называют методом прогонки.

$$A = \begin{pmatrix} C_1 & B_1 & 0 & 0 & \dots & 0 \\ A_2 & C_2 & B_2 & 0 & \dots & 0 \\ 0 & A_3 & C_3 & B_3 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & 0 \\ 0 & 0 & 0 & 0 & A_n & C_n \end{pmatrix}.$$

Теоретическая часть

- Получаем $A_i x_{i-1} + C_i x_i + B_i x_{i+1} = b_i$ (1)
- используем метод прогонки, исходя из следующего рекуррентного соотношения: $x_{i-1} = \alpha_{i-1} x_i + \beta_{i-1}$, (2)
- получаем: $\alpha_2 = \frac{-B_1}{C_1}, \beta_2 = \frac{b_1}{C_1};$ (3) $\alpha_{i+1} = \frac{-B_i}{A_i \alpha_i + C_i}, \beta_{i+1} = \frac{b_i - A_i \beta_i}{A_i \alpha_i + C_i};$ (4)
- Эти формулы представляют собой прямой проход метода. Обратный проход: $x_n = \frac{b_n - A_n \beta_n}{A_n \alpha_n + C_n};$ (5)
- Остальные x_i находим из формулы (2).
- Для применимости метода прогонки достаточно, чтобы матрица A была с диагональным преобладанием.

Алгоритм.

- 1. Вводим $str/stlb$ – количество строк/столбцов, A – элементы расширенной матрицы
- 2. Проверяем матрицу на диагональное преобладание
- 3. Если матрица с диагональным преобладанием тогда п.4, иначе п.8
- 4. Выполняем прямой ход метода (формулы (3), (4)): $c[1]:=A[1,2]/A[1,1]$;
 $d[1]:=A[1,stlb]/A[1,1]$;
- 5. $c[i]:= (-A[i,i+1])/(A[i,i-1]*c[i-1]+A[i,i])$;
- 6. $d[i]:= (A[i,stlb]-A[i,i-1]*d[i-1])/(A[i,i-1]*c[i-1]+A[i,i])$
- 7. Далее обратный ход (формулы (2), (5)):
- 8. $x[str]:=(A[str,stlb]-A[str,str-1]*d[str-1])/(A[str,str]+A[str,str-1]*c[str-1])$;
- 9. $x[i]:=c[i]*x[i+1]+d[i]$;
- 10. Выводим x ;
- 11. Проверки на невязку;
- 12. Заканчиваем алгоритм.
- 13. В программе: $A[i,i+1] = B_i$, $A[i,i] = C_i$, $A[i,i-1] = A_i$, $A[i,stlb] = b_i$, $d[i] = ?_i$, $c[i] = ?_i$, $str = n$.
- 14. Описание входной информации: Str ($Stlb$) – количество строк (столбцов) в расширенной матрице, $A [i, j]$ – матрица A (i – строки, j – столбцы)

Метод прогонки.

- Если матрица системы является разреженной, то есть содержит большое число нулевых элементов, то применяют еще одну модификацию метода Гаусса - метод прогонки. Рассмотрим систему уравнений с трехдиагональной матрицей:

$$\begin{cases} b_1x_1 + c_1x_2 & = d_1 \\ a_2x_1 + b_2x_2 + c_2x_3 & = d_2 \\ \dots & \dots \\ a_ix_{i-1} + b_ix_i + c_ix_{i+1} & = d_i \\ \dots & \dots \\ a_mx_{m-2} + b_{m-1}x_{m-1} + c_{m-1}x_m & = d_{m-1} \\ a_mx_{m-1} + b_mx_m & = d_m \end{cases}$$

Преобразуем первое уравнение системы к виду

- $x_1 = \alpha_1x_2 + \beta_1$, где $\alpha_1 = -c_1/b_1$ $\beta_1 = d_1/b_1$
- Подставим полученное выражение во второе уравнение системы и преобразуем его к виду
- $x_2 = \alpha_2x_3 + \beta_2$ И т.д.

○ На i -ом шаге уравнение преобразуется к виду

○ $x_i = \alpha_i x_{i+1} + \beta_i$, где $\alpha_i = -c_i / (b_i + a_i \alpha_{i-1})$ $\beta_i = (d_i - a_i \beta_{i-1}) / (b_i + a_i \alpha_{i-1})$

○ На m -ом шаге подстановка в последнее уравнение выражения $x_{m-1} = \alpha_{m-1} x_m + \beta_{m-1}$ дает возможность определить значение $x_m = \beta_m = (d_m - a_m \beta_{m-1}) / (b_m + a_m \alpha_{m-1})$:

○ . Значения остальных неизвестных

○ находятся по формулам:

$$x_i = \alpha_i x_{i+1} + \beta_i$$

○ , $i = m-1, m-2, \dots, 1$.

Метод прогонки.

- Для решения систем вида $Ax = F$ или, что то же самое,
- используем метод прогонки, основанный на предположении, что искомые неизвестные связаны рекуррентным соотношением:

- $x_i = \alpha_{i+1}x_{i+1} + \beta_{i+1}$, где $i = n-1, n-2, \dots, 1$ (2)

- Используя это соотношение, выразим x_{i-1} и x_i через x_{i+1} и подставим в уравнение (1):

где $(A_i\alpha_i\alpha_{i+1} + C_i\alpha_{i+1} + B_i)x_{i+1} + A_i\alpha_i\beta_{i+1} + A_i\beta_i + C_i\beta_{i+1} - F_i = 0$ выполняется независимо от решения, если потребовать

- $$\begin{cases} A_i\alpha_i\alpha_{i+1} + C_i\alpha_{i+1} + B_i = 0 \\ A_i\alpha_i\beta_{i+1} + A_i\beta_i + C_i\beta_{i+1} - F_i = 0 \end{cases}$$

- Из первого уравнения $\alpha_{i+1} = \frac{-B_i}{A_i\alpha_i + C_i}$ и $\beta_{i+1} = \frac{F_i - A_i\beta_i}{A_i\alpha_i + C_i}$

$$\alpha_2 = \frac{-B_1}{C_1} \quad \beta_2 = \frac{F_1}{C_1}$$

Метод прогонки.

- После нахождения прогоночных коэффициентов α и β , используя уравнение (2), получим решение системы. При этом,

$$x_i = \alpha_{i+1}x_{i+1} + \beta_{i+1}, \quad i = n-1..1$$

$$x_n = \frac{F_n - A_n\beta_n}{C_n + A_n\alpha_n}$$

- Другим способом объяснения существа метода прогонки, более близким к терминологии конечно-разностных методов и объясняющим происхождение его названия, является следующий: преобразуем уравнение (1) к эквивалентному ему уравнению

- $A'x = F'$ (1')
- с наддиагональной матрицей

$$A' = \begin{pmatrix} C'_1 & B_1 & 0 & 0 & \dots & 0 & 0 \\ 0 & C'_2 & B_2 & 0 & \dots & 0 & 0 \\ 0 & 0 & C'_3 & B_3 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & C'_{n-1} & B_{n-1} \\ 0 & 0 & 0 & 0 & \dots & 0 & C'_n \end{pmatrix}$$

- Вычисления проводятся в два этапа. На первом этапе вычисляются компоненты матрицы C'_i и вектора F'_i , начиная с $i=2$ до $i=n$

$$C'_1 = C_1;$$

$$C'_i = \frac{C_i - A_i B_{i-1}}{C'_{i-1}};$$

$$F'_1 = F_1;$$

$$F'_i = \frac{F_i - F_{i-1} B_{i-1}}{C'_{i-1}}.$$

- На втором этапе, для $i=n, n-1, \dots, 1$ вычисляется решение: $x_n = \frac{F'_n}{C'_n}; x_i = \frac{F'_i - B_i x_{i+1}}{C'_i}.$

Такая схема вычисления объясняет также английский термин этого метода «shuttle».

- Для применимости формул метода прогонки достаточно свойства диагонального преобладания у матрицы A .
- Описание выходной информации: x – матрица-ответ

Спасибо за внимание!