Харківський національний університет імені В.Н. Каразіна

Факультет математики і інформатики

Кафедра прикладної математики

Кваліфікаційна робота

Бакалавра
на тему «*Дослідження впливу граничних умов на стаціонарну лінійну теплопровідність у плоскій стінці*»

Виконав: студент
групи МП41 IV курсу
спеціальність 113 – прикладна математика освітньо-професійна програма
«Прикладна математика»

Токар К.О.

Науковий керівник: доктор техн. наук, професор кафедри прикладної математики

Ромашов Ю.В.

Рецензент: доктор техн. наук, професор, завідувач кафедри комп'ютерно-інтегрованих технологій, автоматизації та робототехніки Харківського національного університету радіоелектроніки

Невлюдов І.Ш.

Харків — 2024 рік

Зміст

Анотація

Токар К.О. Дослідження впливу граничних умов на стаціонарну лінійну теплопровідність у плоскій стінці.
У роботі розглядається стаціонарна задача теплопровідності в плоскій стінці, знаходження теплового поля та теплового потоку. Розглядаємо задачу з граничними умовами першого, другого та/або третього роду.

Tokar K.O. Study of the influence of boundary conditions on stationary linear thermal conductivity in a flat wall.
The paper considers the stationary problem of thermal conductivity in a flat wall, finding the thermal field and heat flow. We consider a problem with boundary conditions of the first, second and/or third kind.

Вступ та постановка задачі

Актуальність данної теми: Математична постановка задач теплопровідності є важливою для різних досліджень, пов'язаних з передачею тепла.. Застосування аналітичних методів дослідження дозволяє отримувати аналітичні розв'язки для спрощених випадків, що сприяє кращому розумінню фізичних процесів та перевірці чисельних методів. Ця тема має широкі застосування, включаючи проектування теплообмінних пристроїв, розрахунок температурного режиму в електроніці, моделювання геотермальних систем, вивчення теплопередачі в матеріалах та структурах, а також розрахунок температурного поля в атмосфері. Зрозуміння та математична постановка задач теплопровідності є важливими для багатьох наукових і технологічних досліджень, спрямованих на вирішення проблем, пов'язаних з передачею тепла.

Диференціальне рівняння теплопровідності дає залежність між температурою, часом та координатами елементарного об'єму.

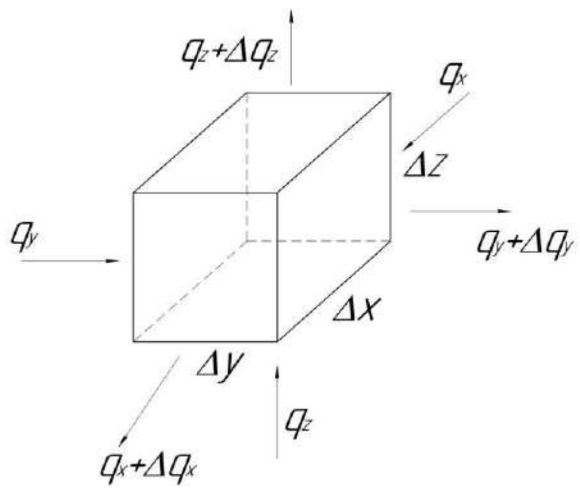

Рисунок 0.1 - Вплив потоків теплоти на тіло див. [2], розділ 2.3

На рисунку 0.1 ми бачимо, як тепловий потік взаємодіє з тілом, в нашому випадку це прямокутний паралелепіпед. Для теплового потоку q_x, якщо $q_x > q_x + \Delta q_x$, то прямокутний паралелепіпед буде нагріватися. [1]

В такому випадку

$$q_x dydz - q_{x+dx} dydz = c\rho \frac{\partial T}{\partial t} dxdydz. \quad (0.1)$$

За допомогою ряда Тейлора ми можемо розкласти q_{x+dx} та обмежити двома першими членами ряда, то:

$$q_{x+dx} \approx q_x + \frac{\partial q_x}{\partial x} dx$$

Якщо ми скористаємось рівнянням (0.1) отримаємо наступне:

$$-\frac{\partial q_x}{\partial x} dxdydz = c\rho \frac{\partial T}{\partial \tau} dxdydz$$

Із закону теплопровідності Фур'є:

$$q_x = -\lambda \frac{\partial T}{\partial x}, \quad (0.2)$$

де λ – коефіцієнт теплопровідності [3]

Тепер скористуємось рівнянням теплопровідності (0.2) та отримаємо

$$c\rho \frac{\partial T}{\partial t} = \lambda \frac{\partial^2 T}{\partial x^2} \quad (0.3)$$

Ми будемо вирішувати задачі за допомогою стаціонарної теплопровідності, бо це полегшить нам вирішення не лінійних задач:

$$T = T(x), \;\; \mathrm{q} = \mathrm{q(x)}$$

$$\frac{\partial q_x}{\partial x} = 0, \;\; q_x = -\lambda \frac{\partial T}{\partial x}$$

Та отримаємо систему

$$\begin{cases} c\rho \frac{\partial T}{\partial t} = -\frac{\partial q_x}{\partial x} \\ q_x = -\lambda \frac{\partial T}{\partial x} \end{cases} \quad (0.4)$$

1 розділ

Математична постановка задачі теплопровідності

1.1 Рівняння теплопровідності

Припустимо, що задані значення в вузлах. Ми зможемо за допомогою лінійної інтерполяції знайти приблизні значення між цих вузлів. Але проблема стоїть в тому, що, якщо мені треба порахувати похідну цієї формули, то отримаємо некоректну операцію, бо малі відхилення в вузлах приведуть до значних відхилень.

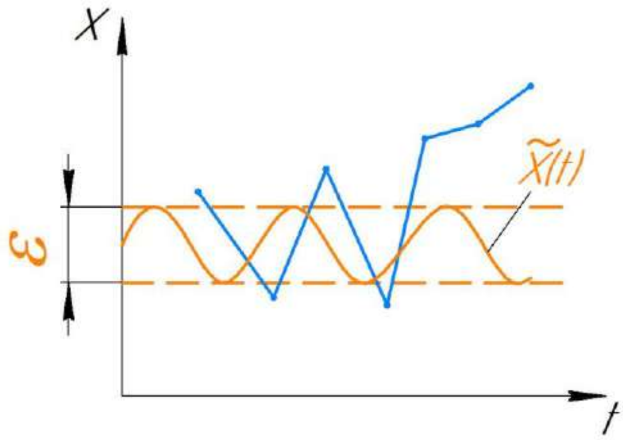

Рисунок 1.1 – Відхилення в вузлах

Ми кажемо, що $\frac{dx}{dt} = 0$, а тепер, ми знайшли наближене – позначено помаранчевим кольором на графіку (рисунок 1.1), позначимо як $\tilde{x}(t)$, та в нас $\tilde{x}(t) \approx x(t)$.

$$|\tilde{x}(t) - x(t)| < \varepsilon$$

Тепер розглядаємо $\frac{d\tilde{x}}{dt}$.

$$\frac{d\tilde{x}}{dt} \neq 0, \quad \left|\frac{d\tilde{x}}{dt}\right| \gg \varepsilon$$

В такому випадку в нас похідна може бути, як додатною, так і від'ємною, а десь може йти на нескінченність.

$\hat{x}(t)$ – лінійна інтерполяція, на (рисунок 1.1) позначено синім кольором.

Тепер розглянемо такий графік (рисунок 1.2), де зеленим кольором позначені значення λ з одного довідника, а синім – з іншого. Помаранчевим кольором позначено те, як воно є насправді.

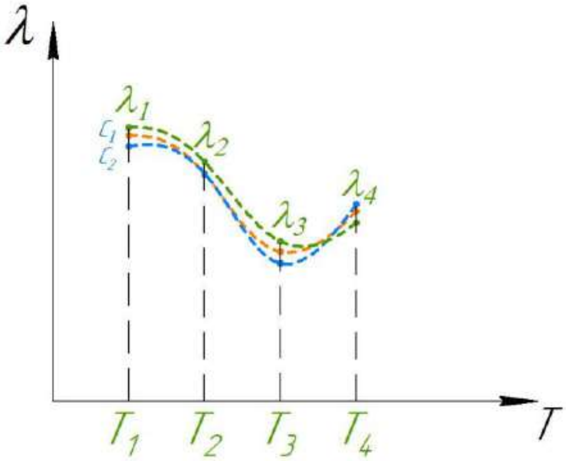

Рисунок 1.2 – Розбіжність значень λ

В різних підручниках при одному T можуть бути різні λ, але вони будуть приблизно однакові. А тепер, якщо нам потрібно буде взяти похідну $\frac{d\lambda}{dT}$, то вона може змінитись суттєво.

Також, причина по якій не варто робити сплайн-інтерполяцію це можливість того, що моя залежність не обов'язково має безперервні похідні.

Тепер виведемо ще одне рівняння

$$\frac{dq}{dx} = 0, \qquad q = -\lambda \frac{dT}{dx}$$

Підставимо q в $\frac{dq}{dx} = 0$, отримаємо

$$\frac{d}{dx}(-\lambda \frac{dT}{dx}) = 0$$

$$\frac{d\lambda}{dx}\frac{dT}{dx} + \lambda \frac{d^2T}{dx^2} = 0$$

Тепер скористаємось рівнянням $\frac{d\lambda}{dx} = \frac{d\lambda}{dT} * \frac{dT}{dx}$

$$\lambda \frac{d^2T}{dx^2} + \frac{d\lambda}{dT}(\frac{dT}{dx})^2 = 0 \qquad (1.1)$$

І тут в нам виходить не лінійне рівняння, де немає точних даних о $\lambda(t)$. Саме тому ми будемо вирішувати систему (0.4), а не рівняння (1.1).

1.2 Граничі умови

Випадок 1:

Гранична умова першого роду визначає температуру на границі області. $T(a) = T_a, \quad T(b) = T_b, \qquad$ де $\quad T_a, T_b -$ відомі.

Це означає, що температура на границі задана і є постійною. Такі умови використовуються, коли температура поверхні тіла відома і підтримується на постійному рівні. [5]

Випадок 2:

Гранична умова другого роду визначає тепловий потік на границі області.

$T(a) = T_a, \quad q_x(b) = q_b, \quad$ де $\quad T_a, q_b -$ відомі.

Це означає, що відомий тепловий потік через границю. Такі умови використовуються, коли заданий тепловий потік на поверхні тіла. [5]

Випадок 3:

Гранична умова третього роду є комбінацією граничних умов першого та другого роду і визначає теплопередачу на границі області.

$$q_x(a) = \alpha_a(T_a - T(a))$$

$$q_x(b) = \alpha_b(T(b) - T_b)$$

$$\begin{cases} \alpha_a T(a) + q_x(a) = \alpha_a T_a \\ \alpha_b T(b) + q_x(b) = \alpha_b T_b \end{cases}$$

Це означає, що тепловий потік через границю залежить від різниці температур між поверхнею та навколишнім середовищем.

α_a, α_b в нас це коефіцієнт теплопередачі, а T_a, T_b це температура середовища. [5]

1.3 Формулювання задачі для узагальненого виду стаціонарної теплопровідності.

Задача теплопровідності у плоскій стінці є однією з класичних задач теорії теплопровідності, яка описує розподіл температури в матеріалі залежно від часу та просторових координат. Розглянемо детальну постановку задачі, включаючи необхідні рівняння та граничні умови.

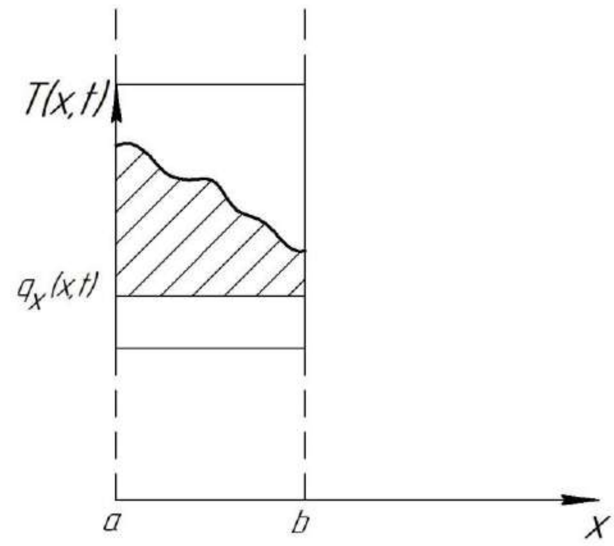

Рисунок 1.3 – Плоска стінка

Розглянемо плоску стінку (рисунок 1.3), розташовану вздовж осі x від a до b. Температура в будь-якій точці стінки в момент часу t позначається як $\mathrm{T}(x,t)$. Необхідно знайти розподіл температури $\mathrm{T}(x,t)$ у стінці, враховуючи диференціальне рівняння теплопровідності та граничні умови. [4]

У випадку не стаціонарної задачі, де є залежність від часу, використовується рівняння теплопровідності з часом:

$$c\rho\frac{\partial T}{\partial t} = \lambda\frac{\partial^2 T}{\partial x^2}$$

Але я буду розглядати стаціонарну задачу, у випадку стаціонарної задачі, такої, де температурне поле не змінюється з часом, рівняння теплопровідності виглядає таким чином:

$$\frac{\partial^2 T}{\partial x^2} = 0$$

2 Розділ

Метод вирішення

2.1 Сітка кінцевої різниці

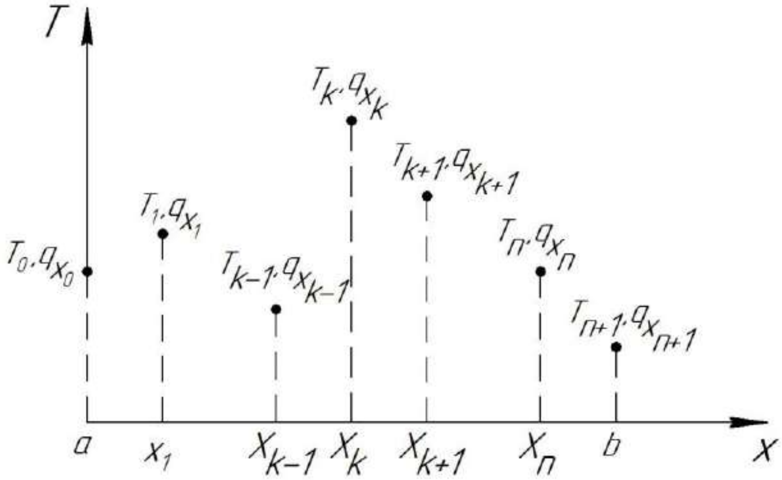

Рисунок 1.4 – Залежність температури та теплового потоку від координати

Область $[a, b]$ поділена на $\mathrm{n}+1$ рівномірний відрізок довжиною $\Delta\mathrm{x}$

$$\Delta\mathrm{x} = \frac{b-a}{n+1},\ \text{де } n - \text{число внутрішніх вузлів сітки},\ n \geq 1$$

Координати вузлів сітки визначаються як:

$$x_k = a + k\Delta x,\ k = 0,1,2, \ldots, n, n+1$$

$$\mathrm{T_k} = T(\mathrm{x_k}),\ \ \mathrm{q_{x,k}} = \mathrm{q_x}(\mathrm{x_k}),\ k = 0,1,2, \ldots, n, n+1$$

$$\begin{cases} \dfrac{\partial \mathrm{q_x}}{\partial \mathrm{x}} = 0 \\ \mathrm{q_x} + \lambda \dfrac{\partial \mathrm{T}}{\partial \mathrm{x}} = 0 \end{cases}$$

$$\Delta\mathrm{x} - \text{крок сітки}$$

$$\mathrm{x_k} - \text{вузли сітки}$$

$$\mathrm{T_k} = T(\mathrm{x_k})\ - \text{вузлові значення температури}$$

$$\mathrm{q_{x,k}} = \mathrm{q_x}(\mathrm{x_k}) - \text{вузлові значення теплового потоку}$$

$$\frac{\partial \mathrm{q_{x,k}}}{\partial \mathrm{x}} = \frac{\mathrm{q_{x,k+1}} - \mathrm{q_{x,k-1}}}{2\Delta x}; \quad \frac{\partial T_k}{\partial \mathrm{x}} = \frac{T_{k+1} - T_{k-1}}{2\Delta x},\ k = 1,2,3, \dots, n$$

$$\begin{cases} \dfrac{\mathrm{q_{x,k+1}} - \mathrm{q_{x,k-1}}}{2\Delta x} = 0 \\ \mathrm{q_x} + \lambda \dfrac{T_{k+1} - T_{k-1}}{2\Delta x} = 0 \end{cases} (1) \Rightarrow \begin{cases} \mathrm{q_{x,k+1}} - \mathrm{q_{x,k-1}} = 0 \\ T_{k+1} - T_{k-1} + \mathrm{q_x} \dfrac{2\Delta x}{\lambda} = 0 \end{cases}, k = 1,2,3, \dots, n \ [6]$$

2.2 Обчислення рівнянь.

Виразимо $\frac{\partial T_k}{\partial x}$ через температури $T_k, T_{k\pm1}, T_{k\pm2}$ з коефіцієнтами a, b, c.

$$\frac{\partial T_k}{\partial x} = aT_k + bT_{k\pm1} + cT_{k\pm2}$$

Розкладемо $T_{k\pm1}$ та $T_{k\pm2}$ в ряд Тейлора

$$T_{k\pm1} = T_k \pm T_k{}'\Delta x + \frac{1}{2}T_k{}''\Delta x^2 \pm \frac{1}{6}T_k{}'''\Delta x^3 + \cdots$$

$$T_{k\pm2} = T_k \pm 2 * T_k{}'\Delta x + 4 * \frac{1}{2}T_k{}''\Delta x^2 \pm 8 * \frac{1}{6}T_k{}'''\Delta x^3 + \cdots$$

Підставимо розклад $T_{k\pm1}$ та $T_{k\pm2}$ у диференціальне рівняння

$$\frac{\partial T_k}{\partial x} = aT_k + b\left(T_k \pm T_k{}'\Delta x + \frac{1}{2}T_k{}''\Delta x^2 \pm \frac{1}{6}T_k{}'''\Delta x^3 + \cdots\right) +$$

$$+ c\left(T_k \pm 2 * T_k{}'\Delta x + 4 * \frac{1}{2}T_k{}''\Delta x^2 \pm 8 * \frac{1}{6}T_k{}'''\Delta x^3 + \cdots\right)$$

Об'єднуємо всі терміни, зводимо подібні та спрощуємо вираз, отримаємо:

$$(a + b + c)T_k \pm (b + 2c)T_k{}'\Delta x + \frac{1}{2}(b + 4c)T_k{}''\Delta x^2 \pm \frac{1}{6}(b + 8c)T_k{}'''\Delta x^3 + \cdots$$

Вирішуємо систему рівнянь для знаходження значень коефіцієнтів a, b та c.

$$\begin{cases} a + b + c = 0 \\ b + 2c = \pm\dfrac{1}{\Delta x} \\ b + 4c = 0 \end{cases} \Rightarrow \begin{cases} a + b + c = 0 \\ b + 2c = \pm\dfrac{1}{\Delta x} \\ b = -4c \end{cases} \Rightarrow \begin{cases} a - 4c + c = 0 \\ -4c + 2c = \pm\dfrac{1}{\Delta x} \\ b = -4c \end{cases} \Rightarrow$$

$$\Rightarrow \begin{cases} a = 3\mathrm{c} \\ c = \mp\dfrac{1}{2\Delta x} \\ b = -4c \end{cases} \Rightarrow \begin{cases} a = \mp\dfrac{3}{2\Delta x} \\ b = \pm\dfrac{2}{\Delta x} \\ c = \mp\dfrac{1}{2\Delta x} \end{cases}$$

Отримані данні підставимо в рівняння, там отримаємо:

$$\frac{dT_k}{dx} = \frac{-3T_k + 4T_{k+1} - T_{k+2}}{2\Delta x}$$

$$\frac{dT_k}{dx} = \frac{3T_k - 4T_{k-1} + T_{k-2}}{2\Delta x}$$

Рівняння балансу теплових потоків для внутрішніх вузлів.

$$-\mathrm{q}_{\mathrm{x,k-1}} + \mathrm{q}_{\mathrm{x,k+1}} = 0, \quad k = 1,2,3,\dots,n$$

Рівняння для вузла при $k = 0$, крайова умова.

$$-3T_0 + 4T_1 - T_2 + \mathrm{q}_{\mathrm{x},0}\frac{2\Delta x}{\lambda} = 0$$

Рівняння для усіх внутрішніх вузлів сітки при $k = 1,2 \dots n$.

$$-T_{k-1} + T_{k+1} + \mathrm{q}_{\mathrm{x},k}\frac{2\Delta x}{\lambda} = 0$$

Ще одне рівняння для вузла при $k = n + 1$, крайова умова.

$$3T_{n+1} - 4T_n + T_{n-1} + \mathrm{q}_{\mathrm{x},n+1}\frac{2\Delta x}{\lambda} = 0$$

На даний момент ми маємо 2n+2 рівняння та 2n+4 невідомі, тому нам треба знайти ще 2 рівняння, ці рівняння візьмемо з граничних умов.

2.3 Дискретна форма запису рівнянь та граничних умов

Вектор T містить температури в вузлах сітки, вектор q містить теплові потоки в вузлах сікти.

$$\mathrm{T} = (\mathrm{T}_0, \mathrm{T}_1, \mathrm{T}_2, \dots, \mathrm{T}_n, \mathrm{T}_{n+1})^{\mathrm{T}} \qquad \mathrm{q} = (\mathrm{q}_{\mathrm{x}_0}, \mathrm{q}_{\mathrm{x}_1}, \mathrm{q}_{\mathrm{x}_2}, \dots, \mathrm{q}_{\mathrm{x}_n}, \mathrm{q}_{\mathrm{x}_{n+1}})^{\mathrm{T}}$$

$$\begin{pmatrix} -3\ 4\ -1\ 0\ 0 \dots 0\ 0\ 0 \\ -1\ 0\ \ 1\ 0\ 0 \dots 0\ 0\ 0 \\ 1\ -1\ \ 0\ 1\ 0 \dots 0\ 0\ 0 \\ \vdots \\ 0\ 0\ 0\ 0\ 0 \dots -1\ \ 0\ \ 1 \\ 0\ 0\ 0\ 0\ 0 \dots 0\ 3\ -4\ 1 \end{pmatrix} \begin{pmatrix} \mathrm{T}_0 \\ \mathrm{T}_1 \\ \mathrm{T}_2 \\ \cdots \\ \mathrm{T}_{n-1} \\ \mathrm{T}_n \\ \mathrm{T}_{n+1} \end{pmatrix} + \begin{pmatrix} \frac{2\Delta x}{\lambda}\ 0\ 0\ 0\ 0 \dots 0\ 0\ 0 \\ 0\ \ \frac{2\Delta x}{\lambda}\ \ 0\ 0\ 0 \dots 0\ 0\ 0 \\ 0\ \ 0\ \ \frac{2\Delta x}{\lambda}\ 0\ 0 \dots 0\ 0\ 0 \\ \vdots \\ 0\ 0\ 0\ 0\ 0 \dots 0\ \ \frac{2\Delta x}{\lambda}\ \ 0 \\ 0\ 0\ 0\ 0\ 0 \dots 0\ \ 0\ \ \frac{2\Delta x}{\lambda} \end{pmatrix} \begin{pmatrix} \mathrm{q}_{x,0} \\ \mathrm{q}_{x,1} \\ \mathrm{q}_{x,2} \\ \cdots \\ \mathrm{q}_{x,n-1} \\ \mathrm{q}_{x,n} \\ \mathrm{q}_{x,n+1} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \cdots \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad (2.1)$$

$$\mathrm{D} \qquad \mathrm{T} \qquad \frac{2\Delta x}{\lambda} \qquad \mathrm{q} \qquad 0$$

Тепер запишемо рівняння (2.1) у вигляді:

$$\mathrm{DT} + \frac{2\Delta x}{\lambda} q = 0$$

Виразимо звідси q.

$$q = -\frac{\lambda}{2\Delta x} \mathrm{DT}$$

Гранична умова, $k = 0$: $\qquad A_a T_0 + B_a q_{x,0} = C_a$

$k = 1,2,3, \dots, \mathrm{n}$: $\qquad \mathrm{q}_{x,k+1} - \mathrm{q}_{x,k-1} = 0$

Гранична умова, $k = n + 1$: $\qquad A_b T_{n+1} + B_b q_{x,n+1} = C_b$

$$\begin{pmatrix} A_a 0\ 0\ 0\ 0 \dots 0\ 0\ 0 \\ 0\ 0\ 0\ 0\ 0 \dots 0\ 0\ 0 \\ 0\ 0\ 0\ 0\ 0 \dots 0\ 0\ 0 \\ \vdots \\ 0\ 0\ 0\ 0\ 0 \dots 0\ 0\ 0 \\ 0\ 0\ 0\ 0\ 0 \dots 0\ 0\ A_b \end{pmatrix} \begin{pmatrix} T_0 \\ T_1 \\ T_2 \\ \dots \\ T_{n-1} \\ T_n \\ T_{n+1} \end{pmatrix} + \begin{pmatrix} B_a 0\ 0\ 0\ 0 \dots 0\ 0\ 0 \\ -1\ 0\ 1\ 0\ 0 \dots 0\ 0\ 0 \\ 0\ -1\ 0\ 1\ 0 \dots 0\ 0\ 0 \\ \vdots \\ 0\ 0\ 0\ 0\ 0 \dots 1\ 0\ 1 \\ 0\ 0\ 0\ 0\ 0 \dots 0\ 0\ B_b \end{pmatrix} \begin{pmatrix} q_{x,0} \\ q_{x,1} \\ q_{x,2} \\ \dots \\ q_{x,n-1} \\ q_{x,n} \\ q_{x,n+1} \end{pmatrix} = \begin{pmatrix} C_a \\ 0 \\ 0 \\ \dots \\ 0 \\ 0 \\ C_b \end{pmatrix}$$

Матричне представлення:

$$BT + Cq = f$$

Вираз для q:

$$q = -\frac{\lambda}{2\Delta x} DT$$

Матричне представлення з виразом для q:

$$BT + C\left(-\frac{\lambda}{2\Delta x} DT\right) = f$$

Перетворення рівняння:

$$\left(B - \frac{\lambda}{2\Delta x} CD\right) T = f$$

Скорочена формула:

$$AT = f$$

Визначення матриці A:

$$A = B - \frac{\lambda}{2\Delta x} CD$$

Розв’язок для T:

$$T = A^{-1} f$$

Знайшли вектор T, тепер знаємо все що нам необхідно, щоб знайти q, підставляємо в рівняння і знаходимо.

$$q = -\frac{\lambda}{2\Delta \mathrm{x}} \mathrm{DT}$$

3 Розділ

Вплив граничних умов на теплопровідність

3.1 Гранична умова 1го роду

Граничні умови 1-го роду передбачають, що температури на кінцях стінки фіксовані:

$\mathrm{T(a)} = \mathrm{T_a}, \quad \mathrm{T(b)} = \mathrm{T_b}$

Візьмемо такі значення: для f таке, що $\mathrm{T_a} = \mathrm{C_a} = 100, \mathrm{T_b} = \mathrm{C_b} = 200$, для матриці B значення A_a та A_b дорівнюють 1, в матриці С: $B_\mathrm{a} = B_\mathrm{b} = 0$. Також $\lambda = 1, a = 1, b = 2$.

Вектор $T = \begin{pmatrix} 100.00000 \\ 120.833344 \\ 150.00000 \\ 154.166672 \\ 200.00000 \end{pmatrix} \quad q = \begin{pmatrix} -83.3335 \\ -125.0000 \\ -83.33331 \\ -125.0000 \\ -83.33325 \end{pmatrix}$

3.2 Гранична умова 1го та 2го роду

Граничні умови 1-го та 2-го роду включають фіксовану температуру на одному кінці та фіксований тепловий потік на іншому:

$\mathrm{T(a)} = \mathrm{T_a}, \quad \mathrm{q_x(b)} = \mathrm{q_b}$

Візьмемо такі значення: для f таке, що $\mathrm{T_a} = \mathrm{C_a} = 100, \mathrm{C_b} = \mathrm{q_b} * \frac{\Delta x}{\lambda} = 100 * \frac{0.2}{1} = 20$, для матриці B значення $A_\mathrm{a} = 1$ та $A_\mathrm{b} = 0$, в матриці С: $B_\mathrm{a} = 0$, $B_\mathrm{b} = 1$. Також $\lambda = 1, a = 1, b = 2$.

Вектор $T = \begin{pmatrix} 100.00000 \\ 105.00000 \\ 111.99998 \\ 112.99998 \\ 123.99995 \end{pmatrix} \quad q = \begin{pmatrix} -20.0001 \\ -29.99994 \\ -19.99994 \\ -29.99994 \\ -19.99994 \end{pmatrix}$

3.3 Гранична умова 1го та 3го роду

Граничні умови 1-го та 3-го роду включають фіксовану температуру на одному кінці та змішану умову на іншому:

$$T(a) = T_a, \quad \alpha_b T(b) + q_x(b) = \alpha_b T_b$$

Візьмемо такі значення: для f таке, що $T_a = C_a = 100, C_b = \alpha_b * T_b = 2 * 200 = 400$, для матриці B значення $A_a = 1$ та $A_b = \alpha_b = 2$, в матриці С: $B_a = 0$, $B_b = 1$. Також $\lambda = 1, a = 1, b = 2$.

Вектор $T = \begin{pmatrix} 100.00000 \\ 114.70588 \\ 135.29410 \\ 138.23528 \\ 170.58821 \end{pmatrix} \quad q = \begin{pmatrix} -58.82361 \\ -88.23523 \\ -58.82349 \\ -88.23529 \\ -58.82349 \end{pmatrix}$

3.4 Гранична умова 2го та 3го роду

Граничні умови 2-го та 3-го роду включають фіксований тепловий потік на одному кінці та змішану умову на іншому:

$$T(a) = q_a, \quad \alpha_b T(b) + q_x(b) = \alpha_b T_b$$

Візьмемо такі значення: для f таке, що $C_a = q_a * \frac{\Delta x}{\lambda} = 50 * \frac{0.2}{1} = 10$, $C_b = \alpha_b * T_b = 2 * 200 = 400$, для матриці B значення $A_a = 0$ та $A_b = \alpha_b = 2$, в матриці С: $B_a = 1$, $B_b = 1$. Також $\lambda = 1, a = 1, b = 2$.

Вектор $T = \begin{pmatrix} 183.00002 \\ 185.50002 \\ 189.00000 \\ 189.50002 \\ 195.00000 \end{pmatrix} \quad q = \begin{pmatrix} -10.00000 \\ -14.99997 \\ -10.00000 \\ -15.00000 \\ -9.999878 \end{pmatrix}$

4 Розділ

Програма

У додатку представлена програма на fortran 90 яка приймає від користувача значення n, λ, a, b, а також коефіцієнти граничних умов $A_a, A_b, B_a, B_b, C_a, C_b$. Після роботи вона виводить два вектори: T та f. Гранична умова для кожного боку стінки може бути своя.

Висновки

В роботі розглядалася задача теплопровідності в плоскій стінці. Ми зрозуміли, чому випадок з стаціонарною задачею краще ніж з нестаціонарною, це поліпшить обчислювання в нелінійних задачах, а також результати будуть більш точними, бо мені не потрібна похідна коефіцієнта теплопровідності. Також, за допомогою fortran 90 я зміг написати програму, що допоможе знайти температуру та тепловий потік в вузлах стінки, для цього необхідно знати граничні умови на краях цієї стінки, це може бути як гранична умова першого роду, так і другого, так і третього.

Список використаних джерел

1. Vedat S. Arpaci. Conduction Heat Transfer. Addison-Wesley Publishing Company, 1966 - 550 стор.
2. Bergman, Theodore L. Fundamentals of heat and mass transfer. John Wiley & Sons, 2011 – 1070 стор.
3. Hahn, David W., and M. Necati Özisik. Heat conduction. John Wiley & Sons, 2012 – 734 стор.
4. Tritt, Terry M., ed. Thermal conductivity: theory, properties, and applications. Springer Science & Business Media, 2005 – 290 стор.
5. Практикум з тепломасообміну. Стаціонарна теплопровідність без внутрішніх джерел теплоти [Електронний ресурс] : навчальний посібник для студентів спеціальності 144 «Теплоенергетика», освітнього ступеня «бакалавр» / КПІ ім. Ігоря Сікорського ; уклад.: І. Е. Фуртат, Н. О. Притула. – Електронні текстові дані (1 файл: 1,8 Мбайт). – Київ: КПІ ім. Ігоря Сікорського, 2021 – 53 стор.
6. Bejan, Adrian. Convection heat transfer. John Wiley & Sons, 2013 – 685 стор.

Додаток

```fortran
program heat_conduction
    implicit none

    ! Оголошення змінних
    integer :: N, m, i, j, k
    real :: La, Lb, Aa, Ab, Ba, Bb, Ca, Cb, lambda, delta_x, h
    real, allocatable :: D(:,:), A(:,:), B(:,:), C(:,:), f(:), q(:), T(:), At(:,:), TEMP(:, :)

    ! Запит на введення даних користувачем
    print *, 'Enter the number of internal network nodes (n):'
    read *, m
    print *, 'Enter (lambda):'
    read *, lambda
    print *, 'Enter coordinates of the beginning and end of the wall (a, b):'
    read *, La, Lb
    print *, 'Enter the coefficients for the boundary conditions (Aa, Ab, Ba, Bb, Ca, Cb)'
    read *, Aa, Ab, Ba, Bb, Ca, Cb

    ! Рахуємо розмір матриці (N)
    N = m + 2
    ! Рахуємо крок сітки (Δx)
    delta_x = (Lb - La) / N

    ! Виділяємо пам'ять для матриць
    allocate(D(N,N), A(N,N), At(N,N), TEMP(N,N), B(N,N), C(N,N), f(N), q(N), T(N))

    ! Обчислюємо коефіцієнт h
    h = -1*lambda/(2*delta_x)

    ! Заповнюємо матрицю D
    D = 0.0
    ! Заповнюємо дерший ряд
    D(1, 1) = -3
    D(1, 2) = 4
    D(1, 3) = -1
    ! Заповнюємо останній ряд
    D(N, N - 2) = 3
    D(N, N - 1) = -4
    D(N, N) = 1
    !Все що всередині
    do i = 2, N - 1
        do j = 1, N
            if (i == j) then
                D(i, j) = 0
            else if (abs(i - j) == 1) then
                D(i, j) = j - i
            end if
        end do
    end do

    ! Заповнюємо матрицю B
    B = 0.0
    B(1, 1) = Aa
    B(N, N) = Ab

    ! Заповнюємо матрицю C
    C = 0.0
    C(1, 1) = Ba
    C(N, N) = Bb
    do i = 2, N - 1
        do j = 1, N
            if (i == j) then
                C(i, j) = 0
            else if (abs(i - j) == 1) then
                C(i, j) = j - i
            end if
```

```
    end do
end do

! Заповнюємо вектор f
f = 0.0
f(1) = Ca
f(N) = Cb

! Виконання основних обчислень
call multiply_matrices(C, D, TEMP, N)

call multiply_matrix_constant(TEMP, h, TEMP, N)

call subtract_matrices(B, TEMP, A, N)

call solve_linear_system(A, f, T, n)

call multiply_matrix_constant(D, h, TEMP, N)

call multiply_matrix_vector(TEMP, T, q, N)

call display(T, 'T')
call display(q, 'q')

! Звільняємо пам'ять
deallocate(D, A, At, TEMP, B, C, f, q, T)

contains
 !Підпрограма вирішує лінійну систему
 subroutine solve_linear_system(A, f, T, n)
    implicit none
    integer, intent(in) :: n
    real, intent(inout) :: A(n, n)
    real, intent(inout) :: f(n)
    real, intent(out) :: T(n)


    integer :: i, j, k
    real :: factor, s

    ! Прямий хід методу Гауса
    do k = 1, n-1
        do i = k+1, n
            factor = A(i, k) / A(k, k)
            A(i, k:n) = A(i, k:n) - factor * A(k, k:n)
            f(i) = f(i) - factor * f(k)
        end do
    end do

    ! Зворотний хід для знаходження розв'язку
    T(n) = f(n) / A(n, n)
    do i = n-1, 1, -1
        s = f(i)
        do j = i+1, n
            s = s - A(i, j) * T(j)
        end do
        T(i) = s / A(i, i)
    end do
end subroutine solve_linear_system

!Підпрограма множення матриць
subroutine multiply_matrices(A, B, R, n)
    integer, intent(in) :: n
    real, intent(in) :: A(n, n), B(n, n)
    real, intent(out) :: R(n, n)
    integer :: i, j

    R = 0.0

    do i = 1, n
        do j = 1, n
            do k = 1, n
```

```
                    R(i, j) = R(i, j) + A(i, k) * B(k, j)
                end do
            end do
        end do
    end subroutine multiply_matrices

    !Підпрограма множення матриці на вектор
    subroutine multiply_matrix_vector(A, x, y, n)
        integer, intent(in) :: n
        real, intent(in) :: A(n, n), x(n)
        real, intent(out) :: y(n)
        integer :: i, j

        y = 0.0

        do i = 1, n
            do j = 1, n
                y(i) = y(i) + A(i, j) * x(j)
            end do
        end do
    end subroutine multiply_matrix_vector

    !Підпрограма множення матриці на константу
    subroutine multiply_matrix_constant(A, constant, R, n)
        integer, intent(in) :: n
        real, intent(in) :: A(n, n), constant
        real, intent(out) :: R(n, n)
        integer :: i, j

        do i = 1, n
            do j = 1, n
                R(i, j) = A(i, j) * constant
            end do
        end do
    end subroutine multiply_matrix_constant

    !Підпрограма віднімань матриць на вектор
    subroutine subtract_matrices(A, B, R, n)
        integer, intent(in) :: n
        real, intent(in) :: A(n, n), B(n, n)
        real, intent(out) :: R(n, n)
        integer :: i, j

        do i = 1, n
            do j = 1, n
                R(i, j) = A(i, j) - B(i, j)
            end do
        end do
    end subroutine subtract_matrices

    subroutine display(v, text)
        real, intent(in) :: v(n)
        character(len=*), intent(in) :: text
        integer :: i

        print *, 'Vector ' // trim(text) // ':'
        do i = 1, N
            print *, v(i)
        end do
    end subroutine
end program heat_conduction
```