

1. OPIS PROBLEMU

Rozwiązywanym zagadnieniem jest problem rozwiązania złożonego procesu wymiany ciepła podczas nagrzewania materiału. W sprawozdaniu zawarto matematyczny opis rozwiązywanego równania oraz opis implementacji programu wykorzystującego metodę elementów skończonych, który symuluje zmiany pola temperatury materiału w czasie.

2. OPIS RÓWNAŃ

Zjawiska cieplne zachodzące w stanie nieustalonym opisuje równanie:

$$\operatorname{div}(k(t)\operatorname{grad}(t))+Q=c\rho\frac{\partial t}{\partial \tau}, \quad (2.1)$$

które można zapisać w postaci różniczkowej:

$$\frac{\partial}{\partial x}(k_x(t))\frac{\partial t}{\partial x}+\frac{\partial}{\partial y}(k_y(t))\frac{\partial t}{\partial y}+\frac{\partial}{\partial z}(k_z(t))\frac{\partial t}{\partial z}+(Q-c\rho\frac{\partial t}{\partial \tau})=0, \quad (2.2)$$

gdzie $k_x(t)$, $k_y(t)$, $k_z(t)$ – anizotropowe współczynniki przewodzenia ciepła zależne od temperatury, natomiast Q – prędkość generowania ciepła.

Rozwiązanie równania stanu ustalonego sprowadza się do minimalizacji funkcjonału o postaci:

$$J=\int \frac{1}{2}(k_x(t)\frac{\partial t}{\partial x}+k_y(t)\frac{\partial t}{\partial y}+k_z(t)\frac{\partial t}{\partial z}-2Qt)dV, \quad (2.3)$$

Ponieważ rozpatrywany materiał ma izotropowe własności (nie zależą od kierunku w materiale), współczynniki $k_x(t) = k_y(t) = k_z(t)$, zatem funkcjonał (2.3) można zapisać jako:

$$J=\int \frac{k(t)}{2}\left(\left(\frac{\partial t}{\partial x}\right)^2+\left(\frac{\partial t}{\partial y}\right)^2+\left(\frac{\partial t}{\partial z}\right)^2-Qt\right)dV, \quad (2.4)$$

2.1 Warunek brzegowy

W opisywanym problemie występuje warunek brzegowy konwekcji. Ponieważ bezpośrednio wprowadzenie warunku brzegowego do funkcjonału (3) nie jest możliwe, prawo konwekcji zostaje dodane do funkcjonału (3) w postaci całki:

$$\int_S \frac{\alpha}{2}(t-t_\infty)dS+\int_S qtdS, \quad (2.5)$$

gdzie S jest powierzchnią, na której zostają nałożone warunki brzegowe, a α jest współczynnikiem konwekcyjnej wymiany ciepła.

W efekcie otrzymane zostaje równanie:

$$J=\int_V \frac{k(t)}{2}\left(\left(\frac{\partial t}{\partial x}\right)^2+\left(\frac{\partial t}{\partial y}\right)^2+\left(\frac{\partial t}{\partial z}\right)^2-Qt\right)dV+\int_S \frac{\alpha}{2}(t-t_\infty)dS+\int_S qtdS, \quad (2.6)$$

2.2 Dyskretyzacja problemu

Dyskretyzacja problemu polega na podzielenie badanego obszaru na elementy, w których będzie interpolowana wartość temperatury.

W każdym elemencie temperatura może zostać przedstawiona jako funkcja wartości węzłowych:

$$t = \sum_{i=1}^N N_i t_i = \{N\}^T \{t\} \quad (2.7),$$

gdzie $\{N\}$ to macierzowa postać wartości funkcji kształtu. Opis funkcji kształtu w elemencie czterowęzłowym został przedstawiony w rozdziale 3.

Po wprowadzeniu zależności (2.7) do funkcjonału 2.5 otrzymano równanie:

$$J = \int_V \frac{k}{2} \left(\left(\frac{\partial \{N\}}{\partial x} \right)^T \{t\} \right)^2 + \left(\left(\frac{\partial \{N\}}{\partial y} \right)^T \{t\} \right)^2 + \left(\left(\frac{\partial \{N\}}{\partial z} \right)^T \{t\} \right)^2 dV - Q \{N\}^T \{t\} + \int_S \frac{\alpha}{2} (\{N\}^T \{t\} - t_\infty)^2 dS + \int q \{N\}^T \{t\} dS, \quad (2.8)$$

Ostateczne rozwiązanie można otrzymać rozwiązując układ równań:

$$\frac{\partial J}{\partial \{t\}} = \int_V k \left(\left\{ \frac{\partial \{N\}}{\partial x} \right\} \left\{ \frac{\partial \{N\}}{\partial x} \right\}^T + \left\{ \frac{\partial \{N\}}{\partial y} \right\} \left\{ \frac{\partial \{N\}}{\partial y} \right\}^T + \left\{ \frac{\partial \{N\}}{\partial z} \right\} \left\{ \frac{\partial \{N\}}{\partial z} \right\}^T \right) \{t\} - Q \{N\} dV + \int_S \alpha (\{N\}^T \{t\} - t_\infty) \{N\} dS + \int q \{N\} dS = 0 \quad (2.9)$$

Równanie w formie macierzowej można zapisać jako:

$$[H] \{t\} + \{P\} = 0, \quad (2.10)$$

Rozwiązywanym zagadnieniem jest równanie stanu nieustalonego opisane równaniem (2.2). Ponieważ w określonej chwili czasu pochodne temperatury mogą być traktowane jako funkcje zmiennych kierunku x, y, z , przekształcone równanie (2.2) można zapisać w formie macierzowej w postaci:

$$[H] \{t\} + [C] \frac{\partial}{\partial \tau} \{t\} + \{P\} = 0, \quad (2.11)$$

Wektor temperatury t jest zależny od czasu (pole temperatury w badanym materiale jest zależne od czasu). W przedziale czasu $\Delta \tau$ wektor zostaje wyznaczony równaniem:

$$\{t\} = \{N_0, N_1\} \begin{pmatrix} \{t_0\} \\ \{t_1\} \end{pmatrix}, \quad (2.12)$$

gdzie $\{N_0\}, \{N_1\}$ są funkcjami kształtu zależnymi od czasu, a $\{t_1\}$ jest wektorem temperatur w węzłach po czasie $\Delta \tau$. Funkcje kształtu przyjmują postać:

$$N_0 = \frac{\Delta \tau - \tau}{\Delta \tau}, N_1 = \frac{\tau}{\Delta \tau}, \quad (2.13)$$

Po wprowadzeniu zależności (2.12) oraz (2.13) do równania (2.11), oraz przyjmując wektor $\{t\}$ przy macierzy $[H]$ jako wektor temperatury po czasie $\{\Delta \tau - dT\}$ równy $\{t_1\}$, otrzymano jawny schemat wyznaczenia temperatury $\{t_1\}$ w przedziale czasu $\Delta \tau$ (gdzie dT oznacza krok czasowy)

$$([H] + \frac{[C]}{\Delta \tau}) \{t_1\} - (\frac{[C]}{\Delta \tau}) \{t_0\} + \{P\} = 0, \quad (2.14)$$

Wyznaczenie wektora temperatury $\{t_1\}$ po czasie $\Delta \tau$ wymaga rozwiązania równania (2.14).

2.3 Opis rozwiązywanego równania w formie macierzowej

2.3.1 Macierz H jest opisana zależnością:

$$[H] = \int_V k(t) \left(\left\{ \frac{\partial \{N\}}{\partial x} \right\} \left\{ \frac{\partial \{N\}}{\partial x} \right\}^T + \left\{ \frac{\partial \{N\}}{\partial y} \right\} \left\{ \frac{\partial \{N\}}{\partial y} \right\}^T + \left\{ \frac{\partial \{N\}}{\partial z} \right\} \left\{ \frac{\partial \{N\}}{\partial z} \right\}^T \right) dV + \int_S \alpha \{N\} \{N^T\} dS$$

Macierz H opisuje transport ciepła w materiale.

2.3.2 Wektor P ma postać:

$$\{P\} = - \int_S \alpha \{N\} t_\infty dS - \int_V Q \{N\} dV + \int_S q \{N\} dS$$

Wektor P opisuje warunek brzegowy konwekcji z uwzględnieniem temperatury otoczenia.

2.3.3 Macierz C:

$$[C] = \int_V c \rho \{N\} \{N\}^T dV$$

Macierz C opisuje wymianę ciepła w procesie nieustalonym w danej chwili czasu.

2.4 Opis metody całkowania Gaussa

Całkowanie metodą Gaussa jest numeryczną metodą wyznaczenia wartości całki oznaczonej. Całkowanie w tej metodzie odbywa się w przedziale $<-1; 1>$. Metoda polega na zsumowaniu wartości w punktach całkowania pomnożonych przez odpowiadające im współczynniki wag. Całkowanie metodą Gaussa można zapisać w postaci:

$$\int_{-1}^1 f(\xi) d\xi = \sum_{i=1}^n w_i \cdot f(\xi_i) \quad , \quad (2.4.1)$$

gdzie n oznacza liczbę punktów całkowania, w_i jest wagą dla danego punktu całkowania. Suma wag w przedziale całkowania jest zawsze równa 2.

Schematy całkowania Gaussa zostały opracowane oraz tabelaryzowane. Noszą nazwę kwadratur Gaussa.

W poniższej tabeli przedstawiono węzły i współczynniki wagowe kwadratur Gaussa dla schematów dla $N = 2, 3$:

N	k	Węzły x_k	Współczynniki A_k
2	0 ; 1	$\pm \sqrt{\frac{1}{3}}$	1
3	0 ; 2	$\pm \sqrt{\frac{3}{5}}$	$\pm \frac{5}{9}$
	1	0	$\pm \frac{8}{9}$

W przypadku całkowania 2d, należy przenieść punkty całkowania z układu jednowymiarowego do dwuwymiarowego. Współczynnik wagowy w takim przypadku zostaje wyznaczony jako iloczyn wag odpowiednich dwóch punktów całkowania.

3. OPIS METOD UŻYTYCH DO OBLICZENIA RÓWNANIA DYSKRETNEGO

3.1 Funkcje kształtu

Funkcje kształtu są prostymi funkcjami służącymi do interpolowania wartości punkcie wewnątrz elementu. Dla dwuwymiarowego, czterowęzłowego elementu skończonego funkcje kształtu są postaci:

$$\begin{aligned} N_1 &= 0.25 \cdot (1 - \xi)(1 - \eta) \\ N_2 &= 0.25 \cdot (1 + \xi)(1 - \eta) \\ N_3 &= 0.25 \cdot (1 + \xi)(1 + \eta) \\ N_4 &= 0.25 \cdot (1 - \xi)(1 + \eta) \end{aligned} \quad , \quad (3.1.1)$$

gdzie ξ oraz η są współrzędnymi punktu, w którym interpolowana zostaje wartość.

Wyznaczono pochodne funkcji kształtu, które również zostaną wykorzystane w obliczeniach:

$$\begin{aligned} \frac{\partial N_1}{\partial \xi} &= -0.25 \cdot (1 - \eta), \quad \frac{\partial N_1}{\partial \eta} = -0.25 \cdot (1 - \xi) \\ \frac{\partial N_2}{\partial \xi} &= 0.25 \cdot (1 - \eta), \quad \frac{\partial N_2}{\partial \eta} = -0.25 \cdot (1 + \xi) \\ \frac{\partial N_3}{\partial \xi} &= 0.25 \cdot (1 + \eta), \quad \frac{\partial N_3}{\partial \eta} = 0.25 \cdot (1 + \xi) \\ \frac{\partial N_4}{\partial \xi} &= -0.25 \cdot (1 + \eta), \quad \frac{\partial N_4}{\partial \eta} = 0.25 \cdot (1 - \xi) \end{aligned} \quad , \quad (3.1.2)$$

3.2 Macierz Jakobiego oraz Jakobian

Ponieważ w równaniu (2.9) konieczne jest obliczenie pochodnych funkcji kształtu względem x i y konieczne jest przejście z wartości w układzie lokalnym elementu (ξ, η) do układu globalnego. Aby wyznaczyć wartość pochodnej funkcji, której zmienna jest zależna od innej zmiennej, należy skorzystać z wzoru obliczania wzoru pochodnej funkcji złożonej: $(f(g(x)))' = f'(g(x)) \cdot g'(x)$. Wynika z tego, że obliczenie pochodnych funkcji kształtu w układzie globalnym będzie wymagało rozwiązania układu równań:

$$\begin{bmatrix} \frac{\partial N_i}{\partial x} \\ \frac{\partial N_i}{\partial y} \end{bmatrix} = \frac{1}{\det[J]} \begin{bmatrix} \frac{\partial y}{\partial \eta} & -\frac{\partial y}{\partial \xi} \\ -\frac{\partial x}{\partial \eta} & \frac{\partial x}{\partial \xi} \end{bmatrix} \begin{bmatrix} \frac{\partial N_i}{\partial \xi} \\ \frac{\partial N_i}{\partial \eta} \end{bmatrix} \quad , \quad (3.1.3)$$

Powyższe sformułowanie nazywa się Jakobianem przekształcenia, który jest obliczany w każdym punkcie całkowania.

4. ZAIMPLEMENTOWANY PROGRAM

4.1 Główne założenia programu

Zaimplementowany program służy do symulacji procesu wymiany ciepła podczas nagrzewania materiału, w którym występuje warunek brzegowy konwekcji. Program wykorzystuje metodę elementów skończonych celem wyznaczenia pola temperatur w każdym węźle po zadanym czasie Δt . Na badany materiał zostaje nałożona siatka. W praktyce oznacza to wygenerowanie siatki na podstawie danych wejściowych. W każdym z węzłów siatki zostaje interpolowana temperatura z wykorzystaniem funkcji kształtu. Celem wyznaczenia wektora temperatur w węzłach program stosuje równanie (2.14). Wyznaczenie macierzy H , C , oraz wektora P wymaga całkowania

numerycznego. Wykorzystaną metodą całkowania jest kwadratura Gaussa, opisana w rozdziale 2.4. W programie zaimplementowano całkowanie w schemacie 2 oraz 3 punktowym. Program został napisany z wykorzystaniem języka programowania C++. Do wykonywania obliczeń wykorzystano bibliotekę *cmath*.

4.2 Dane wejściowe i wyjściowe programu

Dane wejściowe programu symulującego:

Wielkość	Symbol	Jednostka
Czas symulacji	τ	s
Krok czasowy	$d\tau$	s
Współczynnik przewodzenia	k	$\frac{W}{m \cdot K}$
Współczynnik konwekcji	α	$\frac{W}{m^2 \cdot K}$
Temperatura otoczenia	t_{ot}	$^{\circ}C$
Temperatura początkowa materiału	t_0	$^{\circ}C$
Gęstość	ρ	$\frac{kg}{m^3}$
Ciepło właściwe	c	$\frac{J}{kg \cdot C}$
Liczba węzłów	-	-
Liczba elementów	-	-
Współrzędne węzłów	-	-
Id węzłów w elemencie	-	-
Obecność warunków brzegowych w węzłach	-	-

Danymi wyjściowymi programu jest wektor temperatur węzłowych obliczany w każdym kroku czasowym oraz znaleziona wartość minimalnej i maksymalnej temperatury w węzłach.

4.3 Szczegółowy opis działania programu

Program pobiera dane wejściowe z pliku. Na podstawie otrzymanych danych tworzona jest siatka, która składa się z elementów i węzłów. Każdy element zawiera: lokalną macierz $H + H_{bc}$, lokalny wektor P , macierz lokalną C oraz identyfikatory czterech węzłów (wykorzystywane elementy to zawsze elementy czterowęzłowe). Węzeł zawiera współrzędne x oraz y w układzie globalnym oraz flagę krawędzi (jest to zmienna, która przechowuje informację o tym, czy węzeł znajduje się na krawędzi materiału). Ta flaga jest wykorzystywana przy algorytmach obliczających warunki brzegowe. Siatka, jej elementy i węzły są w programie klasami.

Po wczytaniu danych program oblicza kolejno lokalne macierze H , H_{bc} , C oraz wektory P . Następnie następuje agregacja macierzy lokalnych do macierzy globalnych. Następnie w każdym kroku (iteracji) programu zostaje wyznaczony wektor temperatur zgodnie ze schematem (2.14). W tym celu zostają obliczone elementy składowe równania.

Macierz **[H]**:

$$[H] = [H + H_{bc}] + \frac{[C]}{d\tau}, \quad (4.3.1)$$

gdzie macierze $[H]$, $[H_{bc}]$ oraz $[C]$ są macierzami globalnymi po agregacji.

Wektor **{P}**:

$$\{\mathbf{P}\} = \{\mathbf{P}\} + \left\{ \frac{[C]}{d\tau} \right\} \cdot \{T_0\} \quad , \quad (4.3.2)$$

gdzie wektor $\{\mathbf{P}\}$ jest wektorem globalnym po agregacji, a $\{T_0\}$ jest wektorem przechowującym temperatury w węzłach z poprzedniej iteracji (pole temperatury po czasie $\Delta\tau - d\tau$).

Ponieważ wektor $\{\mathbf{P}\}$ w równaniu (2.14) występuje z ujemnym znakiem, w każdej iteracji programu rozwiązany zostaje układ równań w postaci:

$$[H + H_{bc}] + \left\{ \frac{[C]}{d\tau} \right\} = \{\mathbf{P}\} + \left\{ \frac{[C]}{d\tau} \right\} \cdot \{T_0\} \quad , \quad (4.3.3)$$

Układ równań (3.3.3) zostaje rozwiązany numerycznie, z wykorzystaniem metody dokładnej eliminacji Gaussa. Metoda ta została szczegółowo opisana poniżej.

4.4 Szczegółowy opis metod użytych w programie

W poniższym podrozdziale znajduje się szczegółowy opis wszystkich procedur wymienionych w podrozdziale 4.3, które są konieczne do rozwiązywania układu równań (4.3.3).

4.4.1 Obliczanie macierzy $[H]$ dla elementów siatki

W symulacji 2d macierz H przyjmuje postać:

$$[H] = \int_V k(t) \left(\left\{ \frac{\partial \{N\}}{\partial x} \right\} \left\{ \frac{\partial \{N\}}{\partial x} \right\}^T + \left\{ \frac{\partial \{N\}}{\partial y} \right\} \left\{ \frac{\partial \{N\}}{\partial y} \right\}^T \right) dV \quad , \quad (4.4.1)$$

Procedura obliczania macierzy lokalnej H dla każdego elementu składa się z kilku etapów. Poniższa procedura zostaje wykorzystana dla każdego z punktów całkowania w układzie 2d.

1) Całkowanie macierzy $\left\{ \frac{\partial \{N\}}{\partial x} \right\} \left\{ \frac{\partial \{N\}}{\partial x} \right\}^T$ oraz $\left\{ \frac{\partial \{N\}}{\partial y} \right\} \left\{ \frac{\partial \{N\}}{\partial y} \right\}^T$ zostaje zrealizowane poprzez pomnożenie wektorów zawierających wartości funkcji kształtu w punkcie przez wartości odwróconego Jakobianu (zgodnie z twierdzeniem 3.1.3), który jest obliczany dla każdego elementu na podstawie współrzędnych węzłów elementu. Dzięki wykorzystaniu Jakobianu otrzymane zostają wartości pochodnych funkcji kształtu w układzie globalnym.

2) Otrzymana macierze z wartościami pochodnych ∂x oraz ∂y zostają pomnożone przez odpowiedni współczynnik wagowy, dla schematu całkowania dwu lub trzy – punktowego, w układzie 2d.

3) Wartości macierzy z pochodnymi ∂x oraz ∂y zostają zsumowane. Otrzymana macierz zostaje pomnożona przez współczynnik przewodzenia k , a następnie przez wyznacznik macierzy Jakobiego.

4) Otrzymane macierze dla każdego punktu całkowania zostają zsumowane i zapisane do lokalnej macierzy H w każdym elemencie siatki.

Ponieważ wartości pochodnych funkcji kształtu w punktach całkowania są identyczne dla każdego wywołania metody obliczającej macierz H (3.1.2), wartości te zostają przechowane w strukturze `el_4_2d`. Struktura ta przechowuje macierze z wartościami pochodnych funkcji kształtu $\partial \xi$ oraz $\partial \eta$ dla punktów kwadratury Gaussa w schemacie dwu oraz trzy – punktowym. Instancja struktury zostaje przekazana do funkcji jako parametr.

4.4.2 Obliczanie macierzy $[H_{bc}]$ dla elementów siatki

W symulacji 2d macierz H_{bc} przyjmuje postać:

$$\int_S \alpha \{N\} \{N^T\} dS, \quad (4.4.2)$$

Ponieważ macierz H_{bc} jest jedną z macierzy, które dotyczą warunku brzegowego konwekcji, konieczne jest transponowanie punktów całkowania w schematach dwu i trzypunktowym na krawędzie rozpatrywanego elementu w układzie 2d. Analogicznie jak w przypadku obliczania macierzy H , wartości funkcji kształtu w punktach całkowania pozostają za każdym razem takie same, zatem zostają przechowane w strukturze *el_4_2d_HBC*, która zostaje przekazana do funkcji jako parametr.

Poniższa procedura obliczania macierzy H_{bc} zostaje wykorzystana dla każdej krawędzi elementu:

- 1) Zostaje sprawdzony warunek obecności warunku brzegowego na rozpatrywanej krawędzi elementu. Dalsze etapy metody zostaną wykonane tylko w przypadku, gdy flaga obecności warunku brzegowego na obu węzłach tworzących krawędź jest równa 1.
- 2) Całkowanie Gaussa zostaje zrealizowane przez obliczenie wektorów zawierających wartości funkcji kształtu w odpowiednich punktach całkowania. Następnie wektory zostają pomnożone przez wektory transponowane.
- 3) Otrzymane w punkcie 2) macierze zostają pomnożone przez odpowiednie współczynniki wagowe schematu całkowania w układzie 1d (całkowanie realizowane na krawędzi – w przestrzeni jednowymiarowej).
- 4) Obliczony zostaje Jakobian przekształcenia. Ponieważ rozpatrywana powierzchnia jest w układzie 1d (pojedyncza krawędź elementu), a długość przedziału całkowania w kwadraturze Gaussa jest równa 2, Jakobianem przekształcenia układu lokalnego do układu globalnego będzie długość krawędzi podzielona na 2.
- 5) Macierze z każdego punktu zostają zsumowane, a następnie pomnożone przez współczynnik konwekcji α oraz Jakobian.
- 6) Macierze H_{bc} z każdej krawędzi zostają zsumowane i zapisane we wszystkich elementach w siatce. W przypadku, gdy w elemencie nie występuje warunek brzegowy (element nie ma węzłów znajdujących się na krawędzi materiału), macierz H_{bc} zostaje uzupełniona wartościami 0.

4.4.3 Obliczanie wektora $\{P\}$ dla elementów siatki

Wektor P przyjmuje postać:

$$[P] = \int_S \alpha \{N\} t_{ot} dS, \quad (4.4.3)$$

gdzie α – współczynnik konwekcji, t_{ot} – temperatura otoczenia

Ponieważ wektor P dotyczy warunku brzegowego konwekcji, podobnie jak w macierzy HBC może zostać obliczony tylko na krawędziach zewnętrznych rozpatrywanego materiału. Dlatego punkty całkowania ponownie zostają odpowiednio przesunięte na krawędzie elementu. Wartości funkcji kształtu w odpowiednich punktach całkowania zostają przechowane w odpowiednich strukturach by uniknąć ich wielokrotnego liczenia, analogicznie jak przy obliczaniu macierzy H_{bc} .

Poniższa procedura obliczania lokalnego wektora P zostaje wykorzystana dla każdej krawędzi elementu:

- 1) Sprawdzona zostaje obecność warunku brzegowego na krawędzi, podobnie jak w algorytmie obliczania macierzy H_{bc}
- 2) Zrealizowane zostaje całkowanie Gaussa. Wartości funkcji kształtu w punktach całkowania znajdujących się na danej krawędzi zostają zapisane w wektorach. Następnie każdy z otrzymanych wektorów zostaje pomnożony przez wartość temperatury zewnętrznej oraz odpowiednie współczynniki wagowe w układzie 1d.
- 3) Wektory zostają zsumowane do jednego wektora. Otrzymany wektor zostaje pomnożony przez wartość współczynnika konwekcyjnej wymiany ciepła oraz Jakobian 1d (podobnie jak przy macierzy H_{bc} wartość Jakobianu jest połową długości rozpatrywanej krawędzi).
- 4) Otrzymany wektor zostaje zapisany do lokalnego wektora P w każdym elemencie siatki.

4.4.4 Obliczanie macierzy [C] dla elementów siatki

Macierz C jest postaci:

$$[C] = \int_V \rho c_p (\{N\} \{N\}^T) dV, \quad (4.4.4)$$

gdzie ρ oznacza gęstość materiału, a c_p ciepło właściwe.

Poniższa procedura wyznaczenia macierzy C dla wszystkich elementów w siatce zostaje zastosowana w każdym punkcie całkowania w elemencie:

- 1) Całkowanie Gaussa odbywa się poprzez obliczenie wartości funkcji kształtu w punkcie całkowania. Podobnie jak w przypadku macierzy H, wartości zostają przechowane w strukturze, by uniknąć wielokrotnego obliczania tych samych wartości.
- 2) Otrzymany wektor zostaje pomnożony przez wektor transponowany, zgodnie z wzorem (4.4.4)
- 3) Macierz, która jest wynikiem wykonania punktu 2) zostaje pomnożona przez odpowiedni współczynnik wagowy punktu całkowania w układzie 2d.
- 4) Dalej macierz zostaje pomnożona przez wartości ciepła właściwego oraz gęstości materiału.
- 5) Macierze z każdego punktu całkowania zostają zsumowane. Finalna macierz C zostaje zapisana w elemencie.

4.5 Opis metod agregacji

Za pomocą metod opisanych w podrozdziale 4.4 otrzymano lokalne macierze H, H_{bc} , C oraz lokalne wektory P w każdym elemencie siatki. Ponieważ wykorzystywanymi elementami w rozpatrywanym problemie są elementy czterowęzłowe, każdy element zawiera informacje o transporcie ciepła dla czterech węzłów. By móc wyznaczyć globalne pole temperatur, konieczne jest otrzymanie macierzy oraz wektorów globalnych, które będą opisywać transport ciepła oraz warunek brzegowy na całej rozpatrywanej powierzchni.

Agregacja 2d macierzy lokalnych do macierzy globalnej polega na zsumowaniu w tablicy globalnej wyrazów macierzy lokalnej odpowiadającym danym węzłom w siatce.

Poniżej zawarto opis agregacji dwuwymiarowej zastosowanej w programie do agregacji lokalnych macierzy C do macierzy globalnej C oraz macierzy lokalnych H i H_{bc} do globalnej macierzy H.

Opis procedury agregacji 2d macierzy:

- 1) Utworzona zostaje globalna tablica o rozmiarze [liczba węzłów] x [liczba węzłów].

- 2) Do wartości tablicy globalnej w indeksie odpowiadającym identyfikatorom węzłów w elemencie zostają dodane wartości macierzy lokalnej rozpatrywanego elementu.
- 3) Operacja 2) zostaje powtórzona dla każdego elementu siatki.

Przykład fragmentu kodu realizującego agregację 2d (dla macierzy lokalnych C):

```
for(int i = 0; i < GRID.elm_number; i++){
    for(int j = 0; j < 4; j++){
        for(int k = 0; k < 4; k++){
            result[elements[i].id[j]][elements[i].id[k]] += elements[i].C_mat[j][k];
        }
    }
}
```

Powyższą procedurę powtórzone także dla macierzy H oraz H_{bc} .

Agregacja 2d wektora P przebiega w analogiczny sposób do agregacji 2d macierzy, jednak wynikiem jest globalny wektor P, nie macierz. Poniżej przedstawiono fragment funkcji agregującej wektor P:

```
for(int i = 0; i < GRID.elm_number; i++){
    if(elements[i].P_vector != NULL){
        for(int j = 0; j < 4; j++){
            result[elements[i].id[j]] += elements[i].P_vector[j];
        }
    }
}
```

4.6 Symulacja procesu wymiany ciepła

Wykorzystując metody opisane w rozdziale 4.4 oraz 4.5 otrzymano globalne macierze H, C oraz wektor P. Są to struktury, które opisują transport ciepła na całej powierzchni rozpatrywanego materiału. Można przeprowadzić symulację wymiany ciepła w materiale rozwiązując równanie (2.14), mające postać:

$$\left([H] + \frac{[C]}{\Delta \tau}\right) \{t_1\} - \left(\frac{[C]}{\Delta \tau}\right) \{t_0\} + \{P\} = 0$$

4.6.1 Zasady symulacji

Symulacja polega na wyznaczeniu wektora temperatur $\{t_1\}$ co krok czasowy $d\tau$, przez określony czas trwania symulacji. Po każdym kroku czasowym wyznaczona zostaje minimalna i maksymalna temperatura w materiale.

4.6.2 Opis algorytmu do przeprowadzenia symulacji

Główną funkcją w programie jest algorytm, który po otrzymaniu danych wejściowych opisanych w rozdziale 4.2 przeprowadza symulację opisaną w podrozdziale 4.6.1. Funkcja przyjmuje jako parametr liczbę punktów całkowania (2 lub 3 – schemat dwu lub trzy-punktowy całkowania). Następnie na podstawie danych wejściowych tworzona jest siatka. Wyznaczone zostają lokalne

macierze H , H_{bc} , C oraz wektory P , które następnie zostają agregowane. W głównej pętli programu powtarzana jest sekwencja:

- 1) Obliczany zostaje wektor $\{\frac{[C]}{d\tau}\} \cdot t$, gdzie $\{\frac{[C]}{d\tau}\}$ jest macierzą globalną C podzieloną przez wartość kroku czasowego (dzięki temu w każdym kroku można wyznaczyć proces wymiany ciepła tak, jakby proces był stacjonarny). Wektor t jest wektorem temperatur, który jest wyznaczany w każdej iteracji (w pierwszej iteracji wektor t jest równy wektorowi t_0 temperatury początkowej materiału).
- 2) Wektor otrzymany w operacji 1) zostaje dodany do wektora globalnego P : $\{P\} = (\frac{[C]}{\Delta\tau})\{t_0\} + \{P\}$
- 3) Zostaje utworzona macierz współczynników układu równań. Macierzą kwadratową jest macierz globalna siatki (agregowana macierz $H + H_{bc}$). Wektorem wyrazów wolnych w układzie równań jest wektor P otrzymany w kroku 2).
- 4) Otrzymany układ równań, zapisany w macierzy poszerzonej o wektor wyrazów wolnych utworzonej w operacji 3), zostaje rozwiązany z wykorzystaniem metody Gaussa, opisanej w rozdziale 4.7.
- 5) Rozwiązaniem układu równań są temperatury węzłowe, które zostają zapisane do wektora t . Wyznaczone zostają maksymalne i minimalne wartości temperatury w materiale. W następnej iteracji otrzymany wektor zostanie wykorzystany w punkcie 1).

Kroki 1) – 5) zostają wykonane w pętli ilość razy równą czasowi symulacji podzielonemu na przyjęty krok czasowy

4.7 Metoda rozwiązywania układu równań

Metodą wykorzystaną do rozwiązywania układu równań (2.14) jest metoda Gaussa, która wykorzystuje operacje elementarne na wierszach macierzy celem otrzymania macierzy trójkątnej. Po otrzymaniu macierzy trójkątnej, rozwiązania mogą zostać w łatwy sposób wyznaczone sekwencyjnie. Metoda Gaussa jest metodą dokładną.

W programie metoda Gaussa przyjmuje jako dane wejściowe macierz współczynników poszerzoną o wektor wyrazów wolnych. Funkcja zwraca wektor rozwiązań układu równań.

4.8 Pobieranie danych wejściowych z pliku źródłowego

Parametry symulacji są pobierane z pliku źródłowego, który zostaje umieszczony w podkatalogu programu */input*. Utworzono metodę *read_file* która uzupełnia strukturę *init_data* wartościami z pliku. Struktura *init_data* przechowująca parametry symulacji zostaje następnie przekazana jako argument funkcji przeprowadzającej symulację.

4.8.1 Struktura pliku z danymi wejściowymi

Plik zaczyna się od opisu parametrów symulacji. Po opisie atrybutu znajduje się jego wartość, na przykład:

`SimulationTime 500`

Po zebraniu danych dotyczących czasu symulacji, kroku czasowego oraz parametrów badanego materiału i temperatury otoczenia wczytana zostaje siatka. Podane zostają kolejno: liczba węzłów, liczba elementów, a ponieważ w wykorzystywanej strukturze siatki węzły są przechowywane w tablicy 2d, również ilość węzłów w wysokości i szerokości siatki. Dalej zostają podane współrzędne

węzłów siatki oraz identyfikatory węzłów dla każdego elementu siatki. Na końcu pliku znajdują się identyfikatory węzłów, na których występuje warunek brzegowy.

4.8.2 Metoda wczytywania danych

Do wczytania danych wykorzystano metody z biblioteki *fstream*. Znaki zostają wczytane metodą *read()* a następnie zapisane w strukturze *InitData*. Ze względu na przechowanie węzłów w tablicy 2d w siatce, identyfikatory węzłów z warunkiem brzegowym zostają posortowane i sekwencyjnie zapisane w odpowiednich węzłach siatki.

5. PRZEPROWADZONA SYMULACJA

Przeprowadzono cztery symulacje nieustalonej wymiany ciepła. Parametry materiału w każdej próbie pozostają niezmiennie. Różnice między próbami dotyczą kształtu rozpatrywanego materiału oraz elementów w siatce.

5.1 Parametry materiału rozpatrywanego w symulacji

Parametr	Wartość	Parametr	Wartość
Współczynnik przewodzenia ciepła	25 $\frac{W}{m \cdot K}$	Początkowa temperatura materiału	100 °C
Współczynnik konwekcji	300 $\frac{W}{m^2 \cdot K}$	Gęstość	7800 $\frac{kg}{m^3}$
Temperatura otoczenia	1200 °C	Ciepło właściwe	700 $\frac{J}{kg \cdot C}$

6. WYNIKI SYMULACJI

Zebrano najniższą i najwyższą temperaturę dla czterech symulacji w każdym kroku czasowym.

Symulacja 1

Czas symulacji: 500 [s]

Krok czasowy symulacji: 50 [s]

Liczba węzłów w siatce: 16

Liczba elementów w siatce: 9

Kształt próbki: Kwadrat

Kształt elementów: Prostokątne

WYNIKI DLA SCHEMATU CAŁKOWANIA DWUPUNKTOWEGO:

	T_{min}	T_{max}
Iteracja: 0, czas: 50s:	110.038	; 365.815
Iteracja: 1, czas: 100s:	168.837	; 502.592
Iteracja: 2, czas: 150s:	242.801	; 587.373
Iteracja: 3, czas: 200s:	318.614	; 649.387
Iteracja: 4, czas: 250s:	391.256	; 700.068
Iteracja: 5, czas: 300s:	459.037	; 744.063
Iteracja: 6, czas: 350s:	521.586	; 783.383
Iteracja: 7, czas: 400s:	579.034	; 818.992
Iteracja: 8, czas: 450s:	631.689	; 851.431
Iteracja: 9, czas: 500s:	679.907	; 881.058

WYNIKI DLA SCHEMATU CAŁKOWANIA TRÓJPUNKTOWEGO:

	T_{min}	T_{max}
Iteracja: 0, czas: 50s:	110.038	; 365.815
Iteracja: 1, czas: 100s:	168.837	; 502.592

Iteracja: 2, czas: 150s: 242.801 ; 587.373
 Iteracja: 3, czas: 200s: 318.615 ; 649.387
 Iteracja: 4, czas: 250s: 391.256 ; 700.068
 Iteracja: 5, czas: 300s: 459.037 ; 744.063
 Iteracja: 6, czas: 350s: 521.586 ; 783.383
 Iteracja: 7, czas: 400s: 579.034 ; 818.992
 Iteracja: 8, czas: 450s: 631.689 ; 851.431
 Iteracja: 9, czas: 500s: 679.908 ; 881.058

Symulacja 2

Czas symulacji: 500 [s]

Krok czasowy symulacji: 50 [s]

Liczba węzłów w siatce: 16

Liczba elementów w siatce: 9

Kształt próbki: Kwadrat

Kształt elementów: Czworokąty

WYNIKI DLA SCHEMATU CAŁKOWANIA DWUPUNKTOWEGO:

	T_{min}	T_{max}
Iteracja: 0, czas: 50s:	106.537	362.729
Iteracja: 1, czas: 100s:	168.369	517.288
Iteracja: 2, czas: 150s:	245.433	618.095
Iteracja: 3, czas: 200s:	323.311	690.518
Iteracja: 4, czas: 250s:	392.724	746.706
Iteracja: 5, czas: 300s:	458.547	792.755
Iteracja: 6, czas: 350s:	520.124	831.912
Iteracja: 7, czas: 400s:	577.26	866.028
Iteracja: 8, czas: 450s:	630.003	896.23
Iteracja: 9, czas: 500s:	678.528	923.251

WYNIKI DLA SCHEMATU CAŁKOWANIA TRÓJPUNKTOWEGO:

	T_{min}	T_{max}
Iteracja: 0, czas: 50s:	106.344	362.632
Iteracja: 1, czas: 100s:	167.682	517.188
Iteracja: 2, czas: 150s:	244.406	617.993
Iteracja: 3, czas: 200s:	322.003	690.409
Iteracja: 4, czas: 250s:	392.336	746.568
Iteracja: 5, czas: 300s:	457.698	792.56
Iteracja: 6, czas: 350s:	518.924	831.64
Iteracja: 7, czas: 400s:	575.801	865.671
Iteracja: 8, czas: 450s:	628.356	895.789
Iteracja: 9, czas: 500s:	676.747	922.733

SYMULACJA 3

Czas symulacji: 20 [s]

Krok czasowy symulacji: 1 [s]

Liczba węzłów w siatce: 961

Liczba elementów w siatce: 900

Kształt próbki: Kwadrat

Kształt elementów: Prostokątne

WYNIKI DLA SCHEMATU CAŁKOWANIA DWUPUNKTOWEGO:

	T_{min}	T_{max}
Iteracja: 0, czas: 1s:	100	149.557

Iteracja: 1, czas: 2s:	100	; 177.445
Iteracja: 2, czas: 3s:	100	; 197.267
Iteracja: 3, czas: 4s:	100	; 213.153
Iteracja: 4, czas: 5s:	100	; 226.683
Iteracja: 5, czas: 6s:	100	; 238.607
Iteracja: 6, czas: 7s:	100	; 249.347
Iteracja: 7, czas: 8s:	100	; 259.165
Iteracja: 8, czas: 9s:	100	; 268.241
Iteracja: 9, czas: 10s:	100	; 276.701
Iteracja: 10, czas: 11s:	100.001	; 284.641
Iteracja: 11, czas: 12s:	100.002	; 292.134
Iteracja: 12, czas: 13s:	100.003	; 299.237
Iteracja: 13, czas: 14s:	100.005	; 305.997
Iteracja: 14, czas: 15s:	100.009	; 312.451
Iteracja: 15, czas: 16s:	100.014	; 318.631
Iteracja: 16, czas: 17s:	100.021	; 324.564
Iteracja: 17, czas: 18s:	100.032	; 330.271
Iteracja: 18, czas: 19s:	100.046	; 335.772
Iteracja: 19, czas: 20s:	100.064	; 341.085

WYNIKI DLA SCHEMATU CAŁKOWANIA TRÓJPUNKTOWEGO:

	T_{min}	T_{max}
Iteracja: 0, czas: 1s:	100	; 149.557
Iteracja: 1, czas: 2s:	100	; 177.445
Iteracja: 2, czas: 3s:	100	; 197.267
Iteracja: 3, czas: 4s:	100	; 213.153
Iteracja: 4, czas: 5s:	100	; 226.683
Iteracja: 5, czas: 6s:	100	; 238.607
Iteracja: 6, czas: 7s:	100	; 249.347
Iteracja: 7, czas: 8s:	100	; 259.165
Iteracja: 8, czas: 9s:	100	; 268.241
Iteracja: 9, czas: 10s:	100	; 276.701
Iteracja: 10, czas: 11s:	100.001	; 284.641
Iteracja: 11, czas: 12s:	100.002	; 292.134
Iteracja: 12, czas: 13s:	100.003	; 299.237
Iteracja: 13, czas: 14s:	100.005	; 305.997
Iteracja: 14, czas: 15s:	100.009	; 312.451
Iteracja: 15, czas: 16s:	100.014	; 318.631
Iteracja: 16, czas: 17s:	100.021	; 324.564
Iteracja: 17, czas: 18s:	100.032	; 330.271
Iteracja: 18, czas: 19s:	100.046	; 335.772
Iteracja: 19, czas: 20s:	100.064	; 341.085

SYMULACJA 4

Czas symulacji: 20 [s]

Krok czasowy symulacji: 1 [s]

Liczba węzłów w siatce: 961

Liczba elementów w siatce: 900

Kształt próbki: Trapez

Kształt elementów: Trapezy

WYNIKI DLA SCHEMATU CAŁKOWANIA DWUPUNKTOWEGO:

	T_{min}	T_{max}
Iteracja: 0, czas: 1s:	100	; 165.561
Iteracja: 1, czas: 2s:	100	; 204.815

Iteracja: 2, czas: 3s:	100	; 233.059
Iteracja: 3, czas: 4s:	100	; 255.577
Iteracja: 4, czas: 5s:	100	; 274.587
Iteracja: 5, czas: 6s:	100.001	; 291.198
Iteracja: 6, czas: 7s:	100.002	; 306.042
Iteracja: 7, czas: 8s:	100.004	; 319.519
Iteracja: 8, czas: 9s:	100.007	; 331.898
Iteracja: 9, czas: 10s:	100.013	; 343.372
Iteracja: 10, czas: 11s:	100.023	; 354.084
Iteracja: 11, czas: 12s:	100.038	; 364.142
Iteracja: 12, czas: 13s:	100.059	; 373.633
Iteracja: 13, czas: 14s:	100.089	; 382.626
Iteracja: 14, czas: 15s:	100.128	; 391.177
Iteracja: 15, czas: 16s:	100.179	; 399.333
Iteracja: 16, czas: 17s:	100.243	; 407.132
Iteracja: 17, czas: 18s:	100.323	; 414.608
Iteracja: 18, czas: 19s:	100.42	; 421.79
Iteracja: 19, czas: 20s:	100.535	; 428.702

WYNIKI DLA SCHEMATU CAŁKOWANIA TRÓJPUNKTOWEGO:

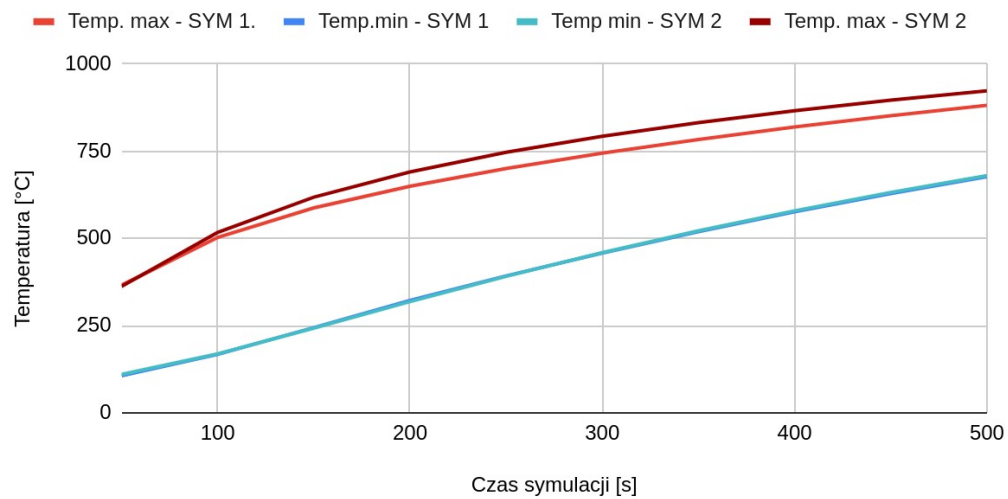
Iteracja: 0, czas: 1s:	100	; 165.561
Iteracja: 1, czas: 2s:	100	; 204.815
Iteracja: 2, czas: 3s:	100	; 233.059
Iteracja: 3, czas: 4s:	100	; 255.577
Iteracja: 4, czas: 5s:	100	; 274.587
Iteracja: 5, czas: 6s:	100.001	; 291.198
Iteracja: 6, czas: 7s:	100.002	; 306.042
Iteracja: 7, czas: 8s:	100.004	; 319.519
Iteracja: 8, czas: 9s:	100.007	; 331.898
Iteracja: 9, czas: 10s:	100.013	; 343.372
Iteracja: 10, czas: 11s:	100.023	; 354.084
Iteracja: 11, czas: 12s:	100.038	; 364.142
Iteracja: 12, czas: 13s:	100.059	; 373.633
Iteracja: 13, czas: 14s:	100.089	; 382.626
Iteracja: 14, czas: 15s:	100.128	; 391.177
Iteracja: 15, czas: 16s:	100.179	; 399.333
Iteracja: 16, czas: 17s:	100.243	; 407.132
Iteracja: 17, czas: 18s:	100.323	; 414.608
Iteracja: 18, czas: 19s:	100.42	; 421.79
Iteracja: 19, czas: 20s:	100.535	; 428.702

7. ANALIZA WYNIKÓW I WNIOSKI

- Równanie Fouriera opisujące nieustalony proces wymiany ciepła może być w konkretnej chwili czasu rozpatrzone tak, jakby dotyczyło procesu ustalonego.
- Pomimo badania tej samej próbki w symulacji 1 oraz 2 uzyskane wyniki zauważalnie się różnią. Świadczy to o fakcie, że przyjęty model siatki i rodzaj elementów mają znaczny wpływ na uzyskiwane wyniki. MES jest metodą, w której dobór odpowiednich parametrów siatki i metod badania mają kluczowy wpływ na wynik.
- Różnice w schemacie dwu i trójpunktowym zauważono jedynie w przypadku symulacji 2, w której elementy miały najbardziej skomplikowane kształty.
- Elementy skrajne badanych próbek nagrzewają się zdecydowanie szybciej niż elementy wewnętrzne. W symulacjach 3 oraz 4 wartości węzły wewnętrzne bardzo nieznacznie zwiększyły swoją temperaturę, podczas gdy węzły mające bezpośredni kontakt z otoczeniem zwiększyły swoją temperaturę kilkakrotnie.
- Dla prób symulacji 1, 2, 3 oraz 4 wygenerowano wykresy temperatury minimalnej i maksymalnej w zależności od czasu symulacji:

Wartości temperatury

Symulacja 1 i 2

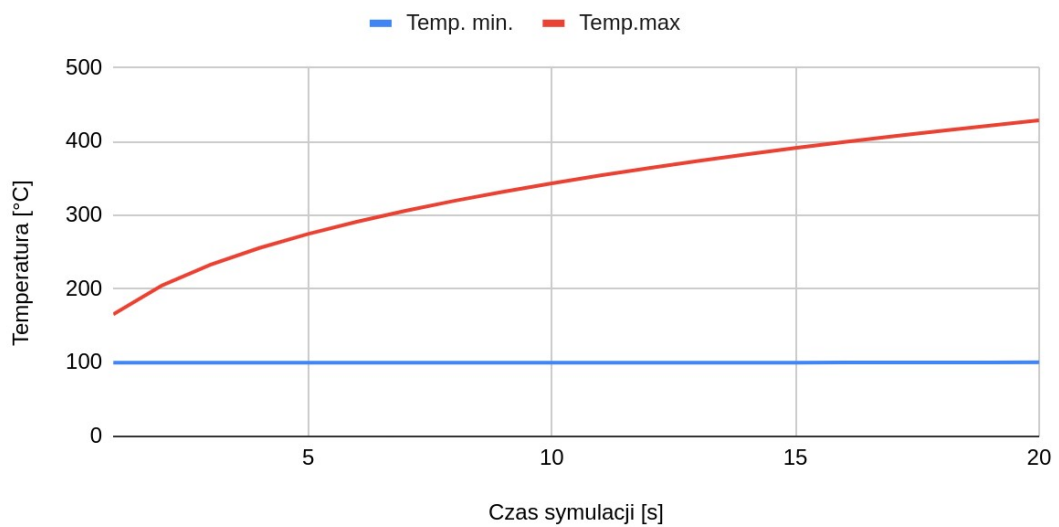


Wykres (1). Temperatury dla symulacji 1 oraz 2

Wykres (2). Temperatury dla symulacji 3

Wartości temperatury

Symulacja 4



Wykres (3). Temperatury dla symulacji 4

- Analizując wykres (1) zauważono znacznie mniejszą różnicę w obliczaniu wartości minimalnych temperatur między symulacjami 1 oraz 2 niż w przypadku wartości maksymalnych.
- W każdym z przypadków symulacji zauważono logarytmiczny przyrost temperatury maksymalnej (temperatura rośnie coraz wolniej).