ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 2005
Seria: BUDOWNICTWO z.104 Nr kol.1695

Marcin DETKA*
Politechnika Świętokrzyska

ALGORYTM BUDOWY "WĘZEŁ PO WĘŹLE" RÓWNAŃ MES W OBLICZENIACH RÓWNOLEGŁYCH PROBLEMÓW DWUWYMIAROWYCH

Streszczenie. W pracy wykorzystano metodę *Free Mesh Method* (FMM) [4,5,6] do rozwiązania dużego dwuwymiarowego problemu eliptycznego w środowisku klastra obliczeniowego, z wykorzystaniem systemu komunikatów MPI oraz pakietu do obliczeń równoległych PETSC. Algorytm FMM zmodyfikowano dla uniknięcia niejednoznaczności przy konstruowaniu lokalnej siatki elementów skończonych wokół węzła. Przykłady potwierdziły skuteczność poprawionego algorytmu i jego efektywność w obliczeniach równoległych.

NODE BY NODE ALGORITHM OF FEM IN PARALLEL COMPUTATIONS OF 2D PROBLEMS

Summary. In the paper, the *Free Mesh Method* (FMM), [4,5,6], is applied to the analysis of the large scale two-dimensional elliptic problems. The MPI environment and PETSC library is used. The original FMM algorithm has bee modified in order to avoid some inconsistency in generation of the mesh elements, around the nodes. The examples confirmed efficiency of the proposed method in the parallel computations.

1. Wprowadzenie

Analiza metodą elementów skończonych (MES) złożonych konstrukcji wymaga efektywnych procedur agregacji równań elementów w system równań globalnych [9]. W przypadku obliczeń sekwencyjnych metodą standardową jest metoda bezpośrednia, wymagająca zdefiniowania macierzy topologii. Macierz ta określa węzły, które należą do każdego z elementów siatki. Taka metoda jest bardzo prosta, lecz czasochłonna i w programach profesjonalnych stosuje się metody bardziej zaawansowane. Jedna z takich

*Opiekun naukowy: Prof. dr hab. inż. Czesław Cichoń

metod, specjalnie efektywna w przypadku obliczeń równoległych, może wynikać z interpretacji globalnego układu równań MES problemów mechaniki ciała stałego, gdzie macierz sztywności $\boldsymbol{K}$ jest miarą sztywności konstrukcji. Oznacza to, że w przypadku dyskretyzacji konstrukcji liniowymi elementami trójkątnymi K_{ij} jest reakcją w węźle i wywołaną przez przemieszczenie węzła j równe jeden. Aproksymacja nieznanych funkcji problemu jest aproksymacją interpolacyjną. Oznacza to, że dla i-tego równania elementy macierzy $K_{ij}\neq 0$ są tylko dla węzłów j, $j=1,2,3,...N$, gdzie N jest liczbą stopni swobody konstrukcji, które należą do elementów skończonych zawierających węzeł i. W rezultacie, globalny układ równań MES może być skonstruowany "węzeł po węźle" (ang. *node by node*), co wymaga budowy tylko lokalnych siatek elementów skończonych wokół każdego węzła obszaru rozwiązania. W taki sposób może być zinterpretowana jedna z metod bezsiatkowych, rozwijana w pracach Yagawy i wsp. [4,5,6] pod nazwą *Free Mesh Method* (FMM). W pracy zaimplementowano FMM do budowy równań MES rozwiązywania dwuwymiarowych problemów eliptycznych, w środowisku klastra obliczeniowego z wykorzystaniem systemu komunikatów MPI [7] oraz pakietu do obliczeń równoległych PETSC [8].

2. Model numeryczny MES

Rozważymy problem eliptyczny opisany przez równanie różniczkowe z warunkami brzegowymi [1]:

$$\begin{aligned} \mathcal{L}u \equiv \nabla\cdot(\alpha(\mathbf{x})\nabla u) &= F(\mathbf{x}) && x\in\Omega \\ u &= 0 && x\in\Gamma_u \\ \frac{\partial u}{\partial n} &= 0 && x\in\Gamma_n \end{aligned} \tag{1}$$

gdzie $\Omega\in\Re^d$ $(d=2,3)$ oraz n wyznacza kierunek normalny na brzegu Γ_n.

Sformułowaniem równoważnym dla (1) jest sformułowanie wariacyjne (Galerkina), które ma postać:

należy znaleźć $u\in \mathrm{H}_0^1(\Omega;\Gamma_u)$, *takie że*

$$a(u,v)=F(v) \quad \textit{dla każdego } v\in \mathrm{H}_0^1(\Omega;\Gamma_u), \tag{2}$$

gdzie:

$$a(u,v)=\int_\Omega \alpha(\mathbf{x})\nabla u\nabla v\,d\mathbf{x}, \quad F(v)=\int_\Omega F(\mathbf{x})v\,d\mathbf{x} \tag{3}$$

$\mathrm{H}_0^1(\Omega;\Gamma_u)$ *jest przestrzenią Hermite'a, zawierającą funkcje z pierwszymi pochodnymi całkowalnymi z kwadratem i zanikającymi na brzegu* Γ_u.

Jest udowodnione, że (2) ma rozwiązanie jednoznaczne, jeśli $\alpha(x)$ jest ściśle dodatnią funkcją skalarną i F jest całkowalne z kwadratem. Jedną z metod rozwiązania jest metoda elementów skończonych (MES) i ma ona następujące sformułowanie.

W pierwszym kroku dyskretyzacji obszary Ω elementami skończonymi (na przykład trójkątnymi w $\Re^2$ i czworościennymi w $\Re^3$). W następnym kroku dla takiego problemu dyskretnego rozwiązujemy zadanie:

należy znaleźć $u_h \in V_h$ *, takie że*

$$a(u_h, v_h) = F(v_h) \quad \textit{dla każdego} \quad v_h \in V_h \tag{4}$$

gdzie V^h *jest skończenie wymiarową podprzestrzenią przestrzeni* $H_0^1(\Omega;\Gamma_u)$, *zawierającą ciągłe funkcje liniowe w każdym z elementów skończonych obszaru zdyskretyzowanego.*

Wielkość rozwiązania dyskretnego w węzłach siatki skończenie elementowej obliczamy rozwiązując liniowy układ równań algebraicznych:

$$\mathbf{Au} = \mathbf{f} \tag{5}$$

gdzie **A** jest macierzą symetryczną i dodatnio określoną, **f** jest wektorem prawej strony, a w wektorze **u** są zawarte wartości węzłowe funkcji u. Macierz $\mathbf{A} = (a_{ij})^k$ i wektor $\mathbf{f} = (f_i)^k$ obliczamy na podstawie równania (4), otrzymując wzory:

$$a_{ij} = a(N_i, N_j), \quad f_j = f(N_j) \tag{6}$$

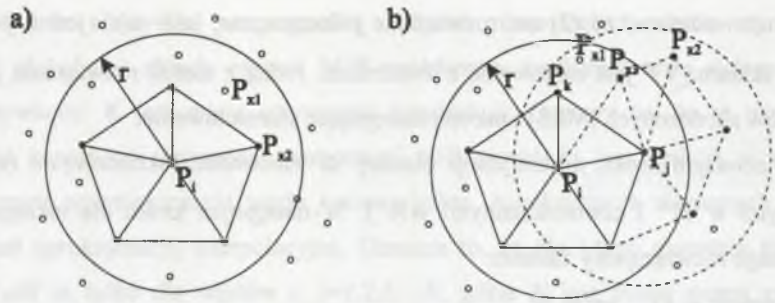
gdzie $N_i, i = 1,2,...,k$ są liniowymi funkcjami kształtu spełniającymi warunek delty Kroneckera $N_i(\mathbf{x}_j) = \delta_{ij}$, w którym $\mathbf{x}_j$ jest współrzędną j węzła siatki. Przez $(\bullet,\bullet)$ oznaczono iloczyn skalarny funkcji w $\Re^k$.

Rozwiązanie wystarczająco dokładne problemu (2) wymaga na ogół rozwiązywania bardzo dużego układu równań (5), co jest przedmiotem dalszej części pracy.

3. Implementacja FMM

3.1. Krótki opis FMM

FMM jest metodą analizy numerycznej MES, w której globalna siatka elementów skończonych reprezentowana jest poprzez lokalne siatki, zbudowane niezależnie wokół każdego z węzłów siatki.



Rys. 1. a) Obszar o promieniu r wokół węzła centralnego P_i; b) Naruszenie konsystentności siatki
Fig. 1. a) Area of influence around central node P_i; b) Violation in selection of nodes

Obliczenia elementów globalnej macierzy sztywności **K** są wykonywane w dwóch krokach:

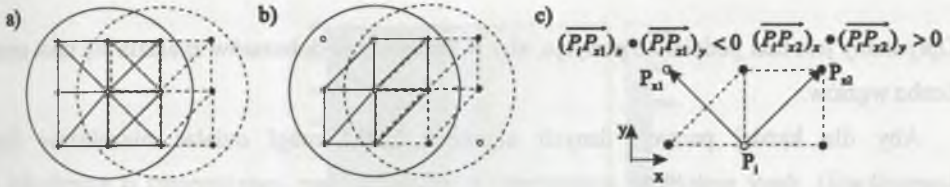
1. Budowa siatki skończenie elementowej wokół węzła centralnego P_i.
2. Obliczenie elementów i-tego wiersza macierzy **K**.

Lokalną siatkę elementów skończonych budujemy wybierając węzły satelitarne P_{xi} z obszaru o promieniu r wokół węzła P_i, rys. 1a. Algorytm budowy lokalnej siatki elementów nie powinien dopuszczać do powstania siatki niekonsystentnej, w której elementy wygenerowane niezależnie dla dwóch węzłów centralnych P_i i P_j nachodzą na siebie, rys. 1b.

3.2. Szczególny przypadek siatki regularnej

Oryginalny algorytm eliminacji węzłów w procesie budowy lokalnej siatki zawodzi, jeśli siatka węzłów jest regularna, co zilustrowano na rys. 2a. Poprawne rozwiązanie jest pokazane na rys. 2b. Otrzymano je dokonując modyfikacji algorytmu w sposób przedstawiony na rys.3. Na rys. 2c $(\overrightarrow{P_iP_x})_x$ i $(\overrightarrow{P_iP_x})_y$ są współrzędnymi wektora o początku w węźle centralnym P_i. Jak widać, niekonsystentność jest usunięta, jeśli P_{xn} jest wybrany w taki sposób, że: $\min_n(P_iP_{xn}) > P_jP_k$ *or* $(\min_n(P_iP_{xn}) = P_jP_k$ *and* $(\overrightarrow{P_iP_{xn}})_x \bullet (\overrightarrow{P_iP_{xn}})_y < 0$ *or* $(\overrightarrow{P_iP_{xn}})_x = 0)$.

Reguła ta ma zastosowanie nie tylko do siatki regularnej, ale w każdym przypadku, gdy zachodzi równość $\min_n(P_iP_{xn}) = P_jP_k$, rys. 1b.



Rys. 2. Generacja siatki regularnej: (a) nieprawidłowa; (b) prawidłowa; (c) interpretacja warunku algorytmu poprawnego dla siatki regularnej
Fig. 2. Generation of the regular mesh nodes: (a) improper; (b) proper selection; (c) interpretation of the proper condition in algorithm

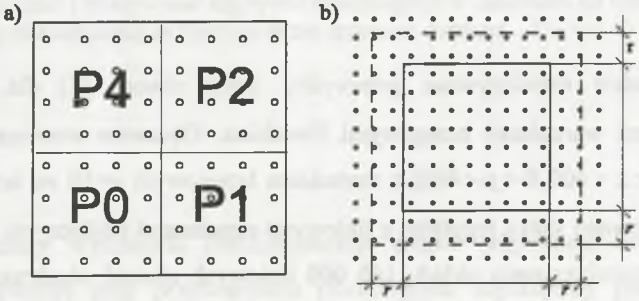
jeśli znajduje się punkt w $\Delta PjPk$
eliminuj $\Delta PiPjPk$
w przeciwnym razie
jeśli odcinek $PiPx$ przecina odcinek $PjPk$
jeśli $(\min_n(P_iP_{xn}) > P_jP_k \ or \ (\min_n(P_iP_{xn}) = P_jP_k \ and \ (\overrightarrow{P_iP_{xn}})_x \bullet (\overrightarrow{P_iP_{xn}})_y < 0 \ or \ (\overrightarrow{P_iP_{xn}})_x = 0))$
eliminuj wszystkie P_{xn}
w przeciwnym razie
eliminuj $\Delta PiPjPk$}

Rys. 3. Poprawione reguły eliminacji przy wyborze elementów lokalnych
Fig. 3. Improved elimination rules for selection of the local elements

4. Obliczenia równoległe

4.1. Implementacja algorytmu do obliczeń równoległych

W algorytmie równoległego obliczania macierzy **K** (oraz wektora prawej strony w układzie równań MES) wykorzystano dekompozycję obszaru rozwiązania na N podobszarów. Węzły należące do danego podobszaru stanowią partycję danych dla danego procesora, rys. 4a.



Rys. 4. a) Podział obszaru na partycje z równą ilością węzłów; b) Tworzenie partycji danych
Fig. 4. a) Domain decomposition with the equal number of nodes; b) Selection of domain partition

Optymalny jest taki podział na partycje, aby w każdym z podobszarów znalazła się taka sama liczba węzłów.

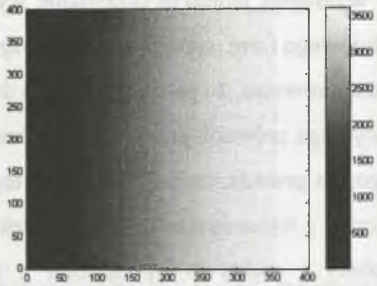
Aby dla każdej partycji danych algorytm FMM mógł działać niezależnie (bez komunikacji), dany podobszar rozszerzano o zakładkę (*ang. overlapping*) o szerokości r, rys.4b, analogicznie do metody Schwarza równoległego rozwiązywania liniowych układów równań algebraicznych [1].

Prezentowany algorytm został zaimplementowany w środowisku systemu operacyjnego LINUX z wykorzystaniem kompilatora języka C GCC. Jako podstawowe narzędzie do komunikacji procesów wybrano system komunikatów MPI [7]. System ten ponadto wspiera pakiet PETSC [8], który posłużył do wszelkich operacji związanych z równoległymi obliczeniami macierzowymi. Do równoległego rozwiązania układu równań liniowych wybrano obiekt KSP (Krylov Subspace Procedures), który jest komponentem systemu PETSC, zwierającym zestaw procedur do rozwiązywania układów równań liniowych metodami iteracyjnymi. W przykładach globalny układ równań MES rozwiązywano metodą gradientu sprzężonego z wykorzystaniem metody addytywnej Schwarza do budowy macierzy wstępnego uwarunkowania.

Wszystkie testy wydajności algorytmu przeprowadzono na klastrze pracującym pod kontrolą systemu operacyjnego LINUX (dystrybucja Mandrake 9.0 (CLIC)), składającego się z dziewięciu jednakowych węzłów. Każdy węzeł klastra wyposażony jest w dwa procesory Xenon 2.4 GHz 1GB pamięci operacyjnej RAM oraz 160GB dysk IDE ATA. Ponadto, każdy węzeł zawiera wbudowane 2 karty sieciowe 100 Mb oraz 1Gb. Podczas testów węzły klastra były połączone wykorzystując topologię magistrali, za pomocą kart 1Gb, wpiętych do przełącznika HP ProCurve. Podczas testów klaster nie wykonywał żadnych innych zadań obliczeniowych.

4.2. Przykład

W przykładzie rozwiązywano jednorodny układ równań (1) dla $\alpha(\mathbf{x}) = const$, z niejednorodnymi warunkami brzegowymi Dirichleta. Obszarem rozwiązania był kwadrat $\Omega = \{(x,y): 0 < x < 400, 0 < y < 400\}$ z warunkiem brzegowym u=10 na brzegu x=0. Obszar ten zdyskretyzowano siatką regularną z liniowymi elementami trójkątnymi, co prowadziło do konieczności rozwiązywania układu 160 000 liniowych równań algebraicznych. Na rys. 5 przedstawiono graficzną interpretacją równoległego rozwiązania.



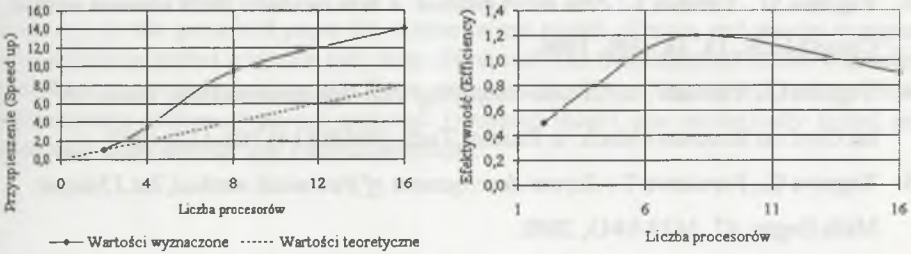
Rys. 5. Graficzna interpretacja wyników obliczeń
Fig. 5. Graphical visualization of computation results

Rozwiązanie przeprowadzono na 2,4,8 oraz 16 procesorach. Wyniki pomiarów i obliczeń zestawiono w tabl. 1 i na rys. 6.

Tablica 1

Czasy obliczeń równoległych

	Liczba procesorów	2	4	8	16
Czas	Dekompozycji	1,1	0,9	0,9	1,5
	Agregacji	3579,0	950,8	282,1	154,9
	Rozwiązania układu równań	333,7	200,4	128,5	120,0
	Całkowity	3913,8	1152,1	411,5	276,4
Przyspieszenie (Speed-up)		1,0	3,4	9,5	14.2
Efektywność (Efficiency)		0,5	0,8	1,2	0,9



Rys. 6. Przyspieszenie i efektywność algorytmu równoległego w zależności od liczby procesorów
Fig. 6. Speed-up and efficiency in function of the processor numbers

5. Wnioski

Porównując czasy wykonania poszczególnych etapów algorytmu, stwierdzono, że największą efektywność przy przetwarzaniu równoległym uzyskujemy podczas budowy siatek lokalnych i agregacji macierzy **K**. oraz rozwiązywania układu równań liniowych.

Szczegółowej weryfikacji wymagają jednakże otrzymane wyniki przyspieszenia algorytmu, lepsze od przyspieszenia liniowego (*ang. superlinear speed-up*).

Uzyskane wyniki pokazują również, że proces akwizycji danych i dekompozycji na z góry założone podobszary nie wymaga zrównoleglenia, bo w szczególnych przypadkach prowadzi to do wydłużenia krótkiego, co prawda, czasu wykonania tego etapu. Według opinii autora agregacja macierzy i wektorów z wykorzystaniem FMM i dekompozycji obszaru rozwiązania może być w sposób spójny i efektywny połączona z równoległym rozwiązywaniem globalnego układu równań MES, wykorzystującym istniejące pakiety procedur numerycznych.

LITERATURA

1. Chan T.F.,Go S.,Zikatanow L.: *Lecture notes on multilewel method for elliptic problems on unstructured grids*. Prepared for the lecture course "28-th Computational Fluid Mechanics", 3-7 March, 1997, von Karman, Institute for Fluid Dynamics, Belgium.
2. Karbowski A.: *Obliczenia równoległe i rozproszone,* Oficyna Wydawnicza Politechniki Warszawskiej, 2001.
3. Yagawa, G., Yamada T.: *Free mesh method: A new meshless finite element method*, Comp.Mech., 18, 383-386, 1996.
4. Yagawa G., Yamada T.: *Parallel computing with free mesh method*, Trans. 14th Int.Conf.on Structural Mech. in Reactor Tech. (SMiRT14) 101-112, 1997.
5. Yagawa G., Furukawa T.: *Recent development of free mesh method*, Int.J.Numer. Meth.Engrg. 47, 1419-1443, 2000.
6. Yagawa Lab., *Free Mesh Methods*, http://garlic.q.t.u-tokyo.ac.jp/research/What-is_FMM/fmm-e.html
7. MPI Mpich Home Page, http://www-unix.mcs.anl.gov/mpi/mpich/
8. PETSC (*Portable, Extensible Toolkit for Scientific Computation*) Home Page, http://www.mcs.anl.gov/petsc/
9. Zienkiewicz O. C., Taylor R. L: *The finite element method, fifth edition*, Butterworth Heinemann 2000.

Recenzent: Dr hab. inż. Bogumił Wrana, prof. Politechniki Krakowskiej