

Praca przejściowa

Piotr Tarasiewicz

nr albumu 271278

Zastosowanie Metody Chimery w Metodzie Siatkowej Boltzmannna

0. Spis treści

1. Wstęp	3
2. Geometria.....	4
3. Chimera	6
4. Interpolacja.....	8
5. Wyniki	11
6. Podsumowanie	14
7. Przypisy	15

1. Wstęp

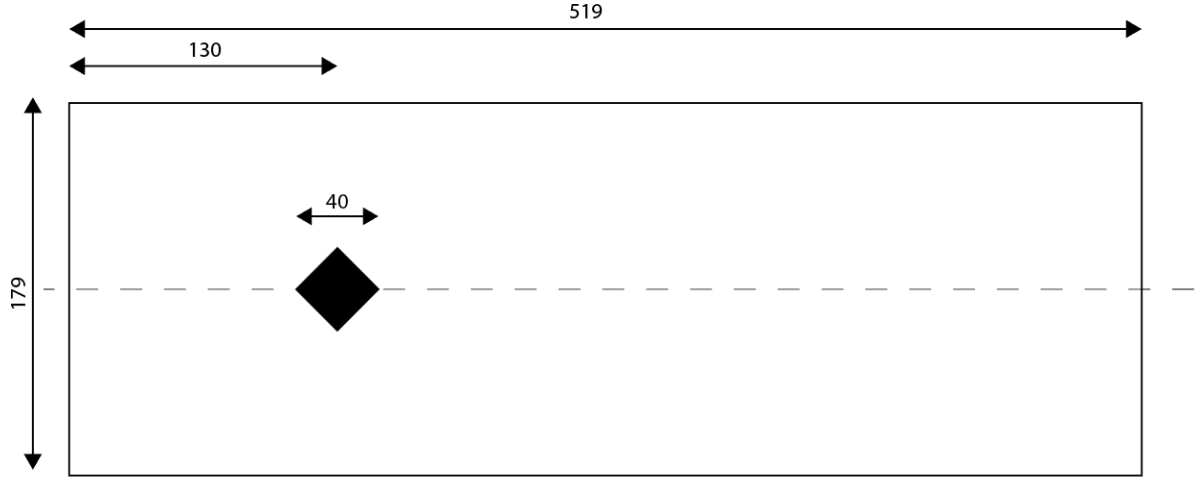
Metoda Chimera, lub inaczej metoda nakładających się siatek, jest metodą pozwalającą na przeprowadzanie symulacji za pomocą więcej niż jednej siatki. Jest to bardzo przydatne wówczas, gdy geometria problemu jest skomplikowana lub występują w niej ruchome elementy i bardzo ciężko jest modelować całą geometrię za pomocą jednej tylko siatki.

W poniższej pracy podjąłem próbę zastosowania metody Chimery w Metodzie Siatkowej Boltzmann (LBM). Omówione będą metody tworzenia nowych siatek, interpolacji oraz wyniki przeprowadzonych symulacji. Wszystkie symulacje zostały przeprowadzone za pomocą metody D2Q9 LBM.

Jako podstawę do implementacji Metody Chimery został użyty program stworzony przez zespół FlowKit Ltd w Lozannie [1] udostępnionego w serwisie PALABOS [2]. Do porównania wyników został użyty program TCLB rozwijany na Politechnice Warszawskiej [3].

2. Geometria

Jako problem został użyty przepływ płynu przez kanał 2D z symetrycznie ustawioną przeszkodą w kształcie diamentu [Rysunek 1]. Wymiary przedstawione na rysunku są podane w jednostkach siatki użytej do przeprowadzania symulacji LBM.



Rysunek 1

$$\begin{aligned} n_x &= 520 \\ l_x &= 519 [lu] \\ n_y &= 180 \\ l_y &= 179 [lu] \\ d &= 40 [lu] \\ p &= 130 [lu] \end{aligned}$$

Na wlocie do kanału została zadana równomierna prędkość $u_{in} = 0.01 \left[\frac{lu}{it} \right]$.

Przyjęto lepkość $\nu = 0.02 \left[\frac{(lu)^2}{it} \right]$.

Przyjmując za wymiar charakterystyczny szerokość diamentu otrzymujemy dla tego przepływu liczbę Reynoldsa:

$$Re = \frac{u_{in} \cdot d}{\nu} = 20. \quad (1)$$

Warunek wlotowy był realizowany za pomocą następujących równań użytych w LBM:

$$u(x = 0) := u_{in}, \quad (2)$$

$$f_{i=1,5,8}(x = 0) := f_{i=1,5,8}^{eq}(x = 0). \quad (3)$$

Warunek wylotowy został zrealizowany następująco:

$$f_{i=3,6,7}(x = lx) := f_{i=3,6,7}(x = lx - 1). \quad (4)$$

Widocznym problemem jest reprezentacja przeszkody na siatce, ponieważ brzegi przeszkody bieżą pod kątem do osi siatki. W związku z widoczna będzie niska rozdzielczość przeszkody [Rysunek 2]. Jest to jeden z problemów używanych w LBM siatek kartezjańskich, które można rozwiązać z pomocą Metody Chimery.



Rysunek 2

Położenie przeszkody jest definiowane za pomocą jej środka (c_x, c_y) oraz szerokości (tudzież wysokości) d . Ponieważ siatka zawiera w osi y parzystą liczbę punktów, środek symetrycznie ustawionej przeszkody, jak również jej rogi, nie wypadają w punktach siatki lecz pomiędzy nimi. Jako należące do przeszkody zaznaczane są te punkty siatki (x, y) , które spełniają równanie:

$$|x - c_x| + |y - c_y| < \frac{d}{2}. \quad (5)$$

Na brzegu przeszkody został zastosowany warunek brzegowy typu *bounce back* [4].

3. Chimera

Do opisu Chimery została przyjęta następująca konwencja: jedna z siatek jest wyróżnioną siatką główną, oznaczoną jako L_0 , określającą globalny układ współrzędnych. Wszystkie pozostałe siatki, nazywane siatkami pomocniczymi, są opisane w odniesieniu do siatki głównej. Siatki pomocnicze oznaczone są jako L_i , $i = 1, 2, \dots, n_L$, gdzie n_L oznacza liczbę użytych siatek pomocniczych. Siatki są ułożone według hierarchii. Jeśli dany punkt leży wewnątrz siatek L_i i L_j oraz $i < j$, to w tym miejscu siatka L_i , a obliczenia są przeprowadzane na siatce L_j .

Siatkę pomocniczą L_i oprócz wymiarów samej siatki opisują dodatkowo następujące parametry:

- wektor \mathbf{r}_i , opisujący położenie początku siatki, czyli punkt (0,0);
- macierz rotacji \mathbf{R}_i , opisująca obrót wektorów z L_i do L_0 ;
- skala s_i , opisująca skalę zagęszczenia L_i względem L_0 .

Każdy wektor położenia $\mathbf{s}^{(i)}$ opisany we współrzędnych siatki L_i można przedstawić we współrzędnych L_0 jako:

$$\mathbf{s}^{(0)} = \mathbf{r}_i + \mathbf{R}_i \mathbf{s}^{(i)}. \quad (6)$$

Oraz chcąc przejść w drugą stronę:

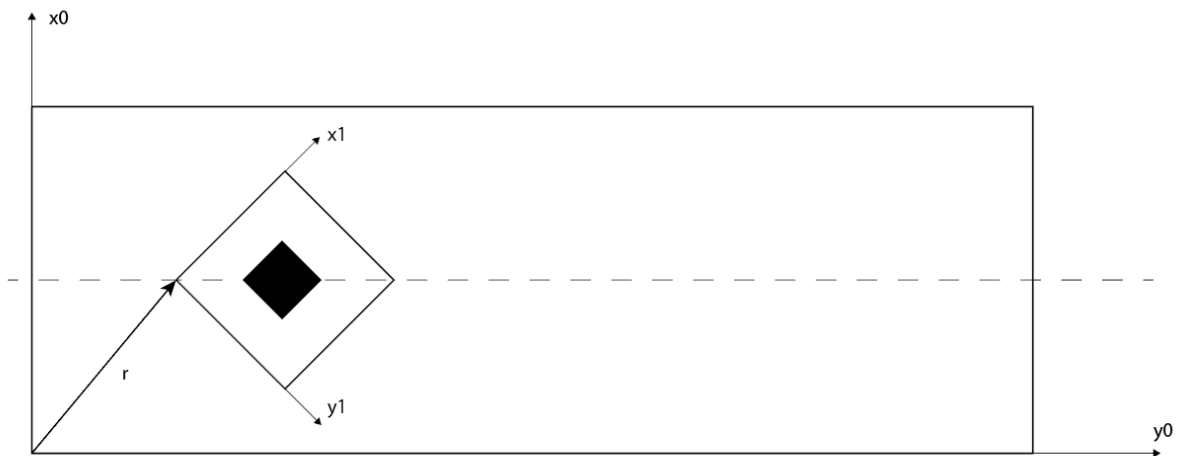
$$\mathbf{s}^{(i)} = \mathbf{R}_i^T (\mathbf{s}^{(0)} - \mathbf{r}_i). \quad (7)$$

Podobnie dla wektorów prędkości zachodzi:

$$\mathbf{u}^{(0)} = \mathbf{R}_i \mathbf{u}^{(i)}, \quad (8)$$

$$\mathbf{u}^{(i)} = \mathbf{R}_i^T \mathbf{u}^{(0)}. \quad (9)$$

Do rozpatrywanego problemu użyta została jedna siatka pomocnicza L_1 otaczająca przeszkodę w taki sposób, żeby osie siatki były równoległe do brzegów przeszkody [Rysunek 3].



Rysunek 3

Parametry siatki L_1 :

$$n_{x1} = 320,$$

$$n_{y1} = 320,$$

$$s = 4,$$

$$\mathbf{r} = [130 - 20\sqrt{2}, 89.5],$$

$$\mathbf{R} = \begin{bmatrix} \cos(\frac{\pi}{4}) & \sin(\frac{\pi}{4}) \\ -\sin(\frac{\pi}{4}) & \cos(\frac{\pi}{4}) \end{bmatrix}.$$

Po zdefiniowaniu siatek na wszystkie z nich nanoszona jest geometria. Ważne jest, aby na każdą siatkę geometria była nanoszona oddzielnie, a nie przenoszona z siatki głównej na pomocnicze. W tym drugim przypadku, jeśli jakiś element byłby niedokładnie reprezentowany na siatce głównej, to te niedokładności byłyby przenoszone na siatki pomocnicze, co przeczyłoby całej idei stosowania Metody Chimery.

Gdy skala siatek jest różna, konieczne jest również przeskalowanie pozostałych parametrów siatki, czyli jej prędkości, gęstości i kroku czasowego. Do testów posłużył przypadek w którym zachowywana jest skala prędkości. W rozpatrywanym przykładzie siatka jest 4 razy gęstsza. Chcemy, żeby parametry fizyczne były na obydwu siatkach jednakowe. Przyjmijmy zatem, że na głównej siatce zachodzi:

$$1 [m] = 1 [lu],$$

$$1 [s] = 1 [it].$$

Uwzględniając wymienione wcześniej czynniki na siatce pomocniczej zachodzi:

$$1 [m] = 4 [lu_1],$$

$$1 \left[\frac{m}{s} \right] = 1 \left[\frac{lu_1}{it_1} \right].$$

Stąd:

$$1 [s] = 4 [it_1],$$

$$1 \left[\frac{m^2}{s} \right] = 4 \left[\frac{(lu_1)^2}{it_1} \right].$$

Zatem przy przyjętych założeniach, w trakcie każdej iteracji siatki głównej należy wykonać 4 iteracje siatki pomocniczej oraz przyjąć na niej 4 razy większą gęstość.

4. Interpolacja

W Metodzie Chimery kluczowym elementem jest interpolacja pomiędzy siatkami. Etap ten jest wykonywany w każdej iteracji i został wprowadzony do zwykłej funkcji iteracyjnej LBM. Iteracje można opisać schematycznie jako algorytm:

Iteracja:

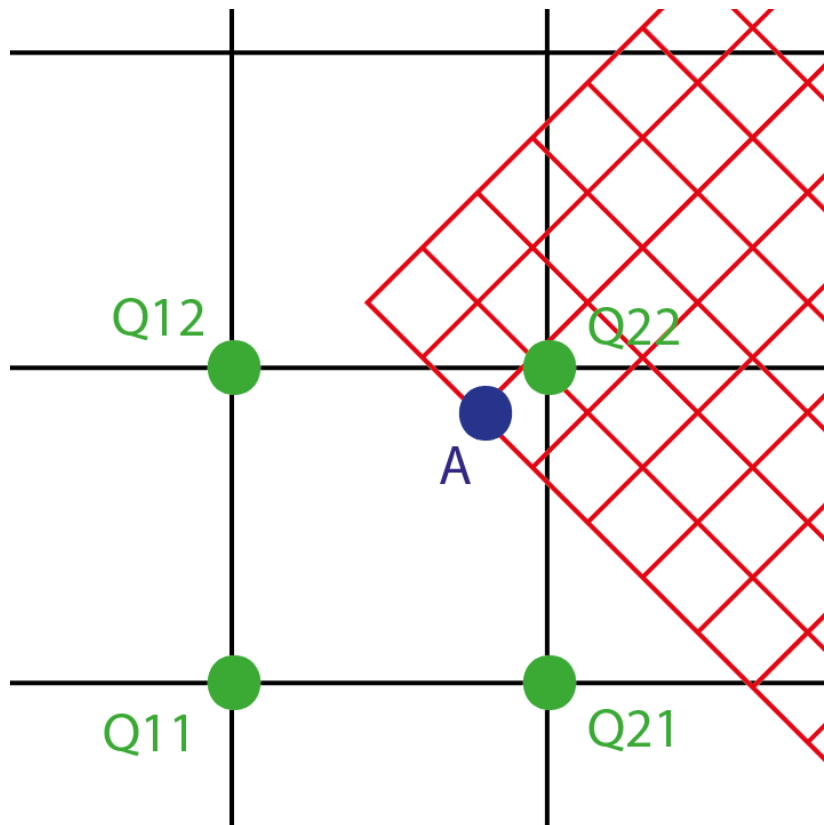
1. Obliczenie wartości ϱ, u korzystając z wartości f .
2. Interpolacja ϱ i u pomiędzy siatkami.
3. Obliczenie f^{eq} .
4. Etap kolizji: $f := f - \omega(f - f^{eq})$.
5. Etap streamingu.

W omawianym problemie mamy do czynienia z dwoma rodzajami interpolacji: z siatki głównej na pomocniczą oraz odwrotnie. Każdy z tych rodzajów zostanie omówiony oddzielnie.

Interpolacja z siatki głównej na pomocniczą

Trudnością w tym etapie jest fakt, że interpolujemy z siatki rzadszej na gęstą, natomiast ułatwieniem, że wiemy dokładnie na jaki obszar chcemy interpolować, to jest na granicę siatki pomocniczej.

Do rozwiązania pierwszego z tych problemów zostanie użyta interpolacja dwuliniowa. Dla każdego z punktów na granicy siatki pomocniczej, jego wartość prędkości i gęstości będzie interpolowana z czterech najbliższych mu punktów siatki głównej [Rysunek 4].



Rysunek 4

Dla każdego punktu A na granicy siatki pomocniczej znamy wektor $s_A^{(1)}$. Następnie korzystając ze wzoru (6) można obliczyć wektor $s_A^{(1)}$, czyli współrzędne A w siatce głównej. Biorąc odpowiednio podłogi i sufity z tych współrzędnych otrzymujemy punkty $Q_{11}, Q_{12}, Q_{21}, Q_{22}$.

Niech $Q_{11} = (x_1, y_1), Q_{22} = (x_2, y_2)$. Wówczas korzystając z interpolacji dwuliniowej otrzymujemy:

$$\varrho(A) = \frac{1}{(x_2 - x_1)(y_2 - y_1)} \begin{bmatrix} x_2 - x & x - x_1 \end{bmatrix} \begin{bmatrix} \varrho(Q_{11}) & \varrho(Q_{12}) \\ \varrho(Q_{21}) & \varrho(Q_{22}) \end{bmatrix} \begin{bmatrix} y_2 - y \\ y - y_1 \end{bmatrix}. \quad (10)$$

Analogicznie obliczamy wartość $u^{(0)}(A)$, którą potem musimy obrócić zgodnie z równaniem (9) otrzymując $u^{(1)}(A)$.

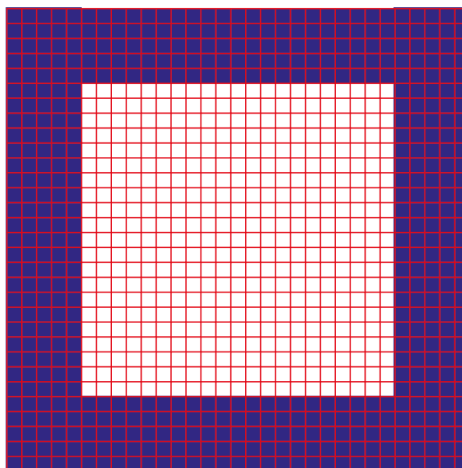
W bardzo mało prawdopodobnym przypadku, że punkt A znajduje się na jednej z linii siatki głównej lub w jednym z jej punktów, używamy odpowiednio interpolacji liniowej lub wartości z tego punktu.

Interpolacja z siatki pomocniczej na główną

Tym razem sytuacja jest odwrotna. Ułatwieniem jest fakt, że interpolujemy z siatki gęstszej na rzadszą, jednak trudnością jest fakt, że nie znamy dokładnego obszaru na który chcemy interpolować.

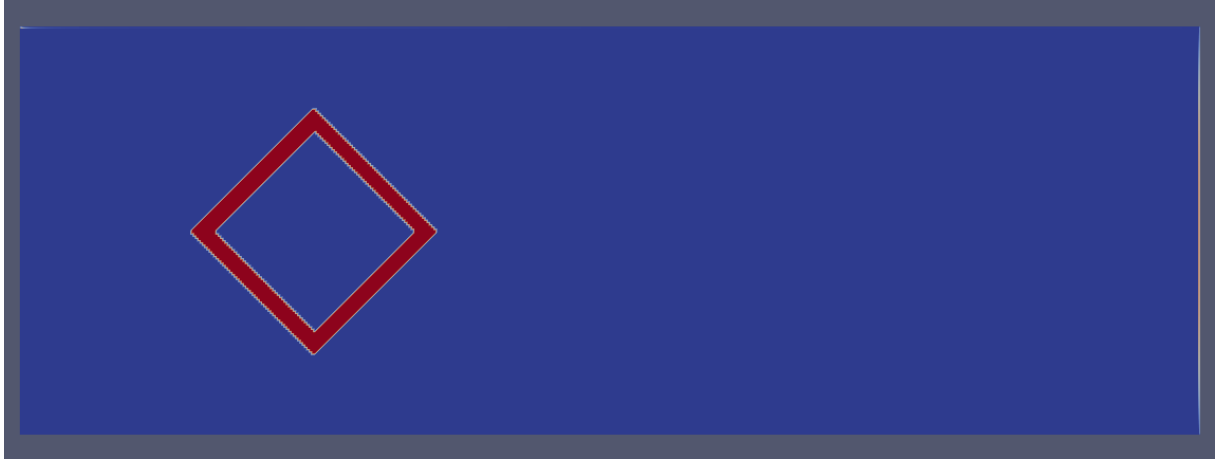
Korzystając z pierwszego ułatwienia do interpolacji została użyta jedynie interpolacja najbliższego punktu, pamiętając oczywiście o odpowiednim obracaniu wektorów prędkości.

Do wyznaczenia obszaru na który interpolujemy została użyta następująca metoda. Na początku wyznaczamy ramkę siatki pomocniczej, która z zapasem będzie stanowiła spójną ramkę po zrzutowaniu jej na siatkę główną [Rysunek 5].



Rysunek 5

Następnie dla każdego z punktów tej ramki (zaznaczonej na niebiesko) wyznaczmy najbliższy mu punkt na siatce głównej. Jeśli ten punkt nie służył wcześniej do interpolacji na siatkę pomocniczą (należy przechowywać flagi, oznaczające, które punkty służyły do której interpolacji), to dodajemy go do obszaru na który będziemy interpolować. W przypadku badanego problemu użyliśmy ramki o grubości 30 i powstał obszar widoczny na rysunku [Rysunek 6]. Jak widać jest to spójna ramka, na którą zostanie przeprowadzona interpolacja z siatki pomocniczej.



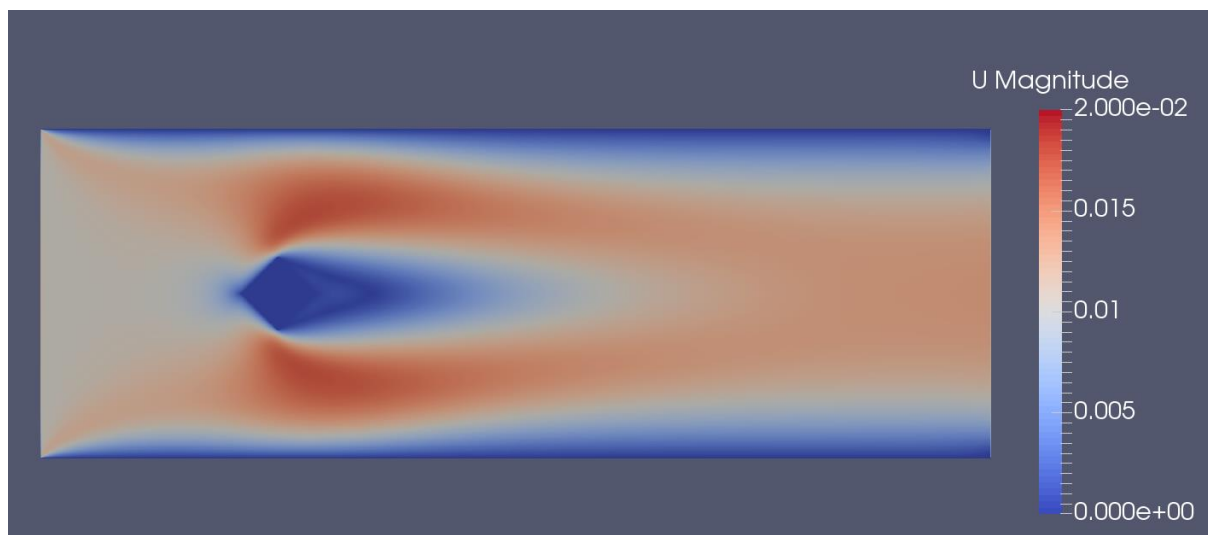
Rysunek 6

Pozostaje do rozwiązania jeszcze jeden problem. Kiedy w 4. etapie iteracji (kolizji) aktualizujemy funkcję f , to w granicznych obszarach każdej siatki w f znajdują się błędne informacje. Dzieje się tak ponieważ w poprzednim etapie streamingu w obszarach granicznych nieznane są wartości, które powinny być streamowane z zewnątrz granicy. Dlatego w tym miejscu w obszarach granicznych przyjmujemy:

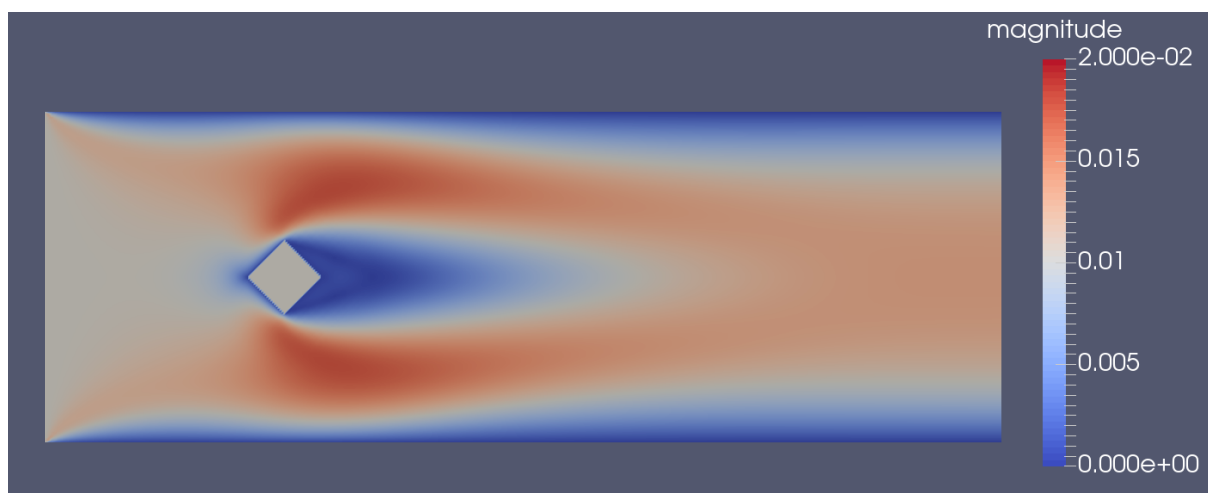
$$f = f^{eq}. \quad (11)$$

5. Wyniki

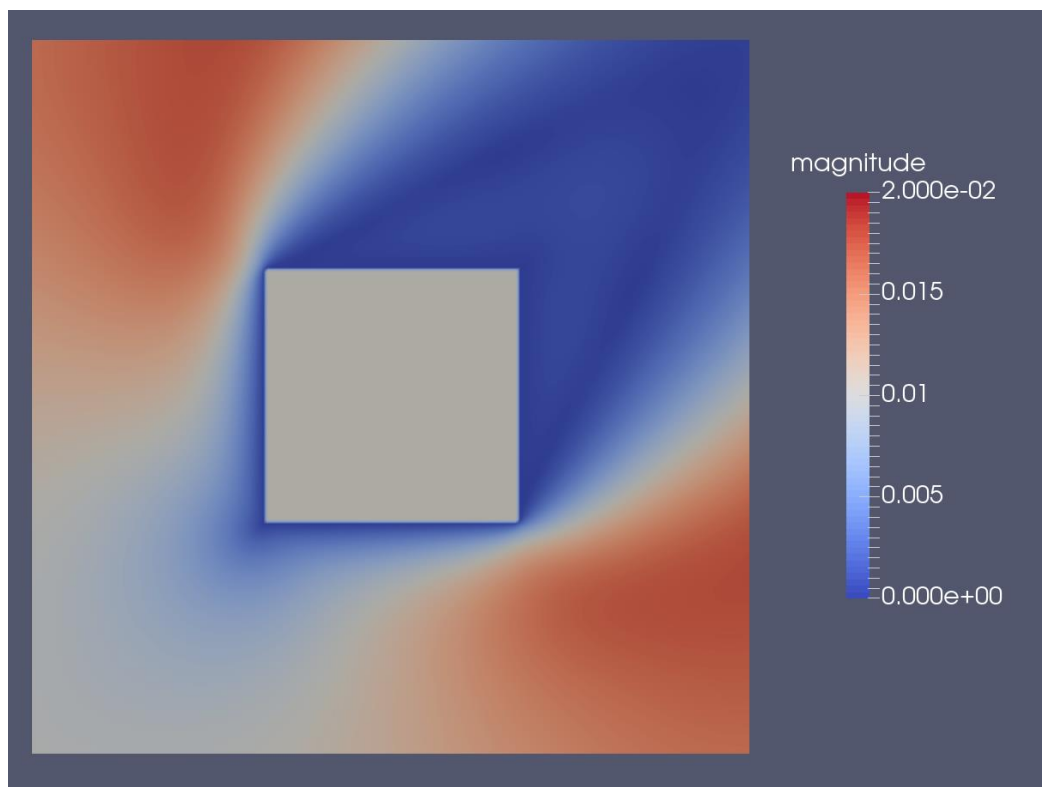
Poniżej zamieszczone są wyniki symulacji przeprowadzonej dla Chimery dla 25000 iteracji siatki głównej, porównane z wynikami uzyskanymi dla zwykłej LBM, dla tego samego problemu, ale na czterokrotnie gęstej siatce, obliczone przy użyciu TCLB. Porównywane są wartości modułu prędkości w przekrojach $x = 130$ (przez środek przeszkody), $x = 150$ (tuż za przeszkodą), $x = 260$ (w połowie kanału). Kolor zielony odpowiada wynikom uzyskanym przy użyciu Metody Chimery.



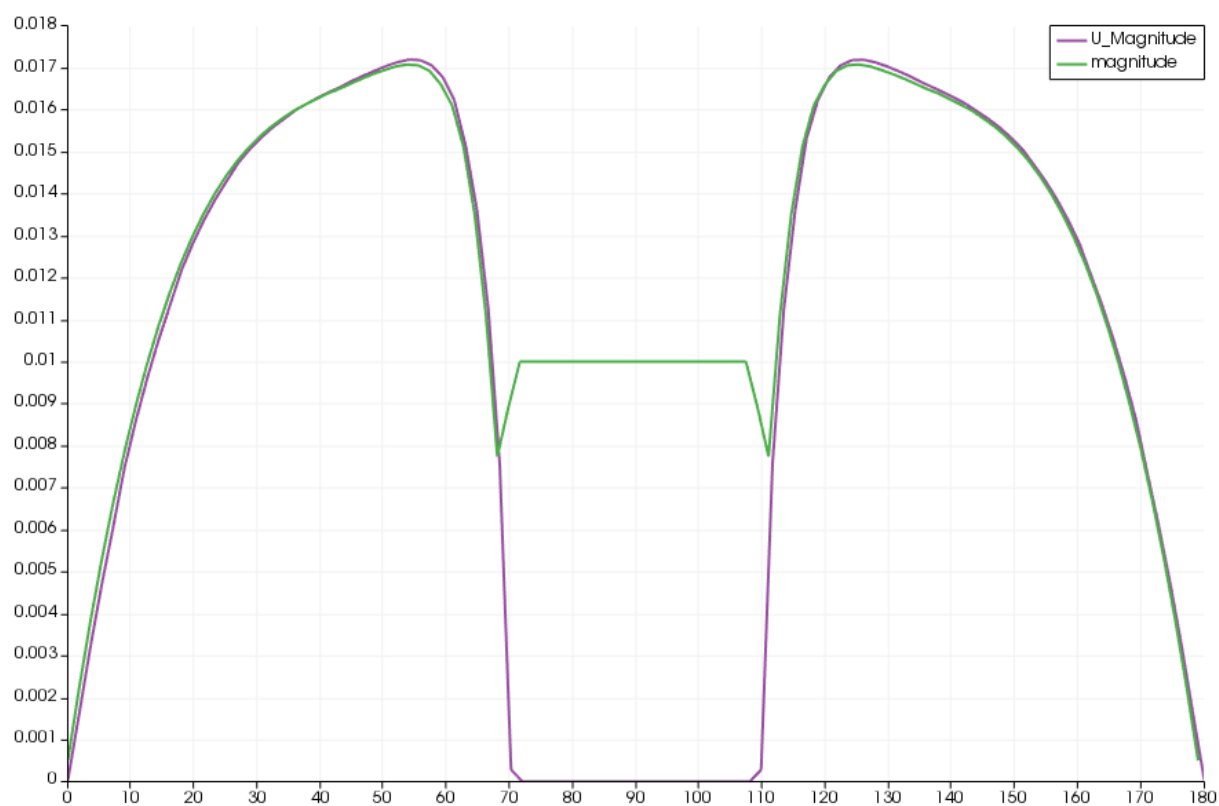
Rysunek 7. Przepływ uzyskany za pomocą TCLB.



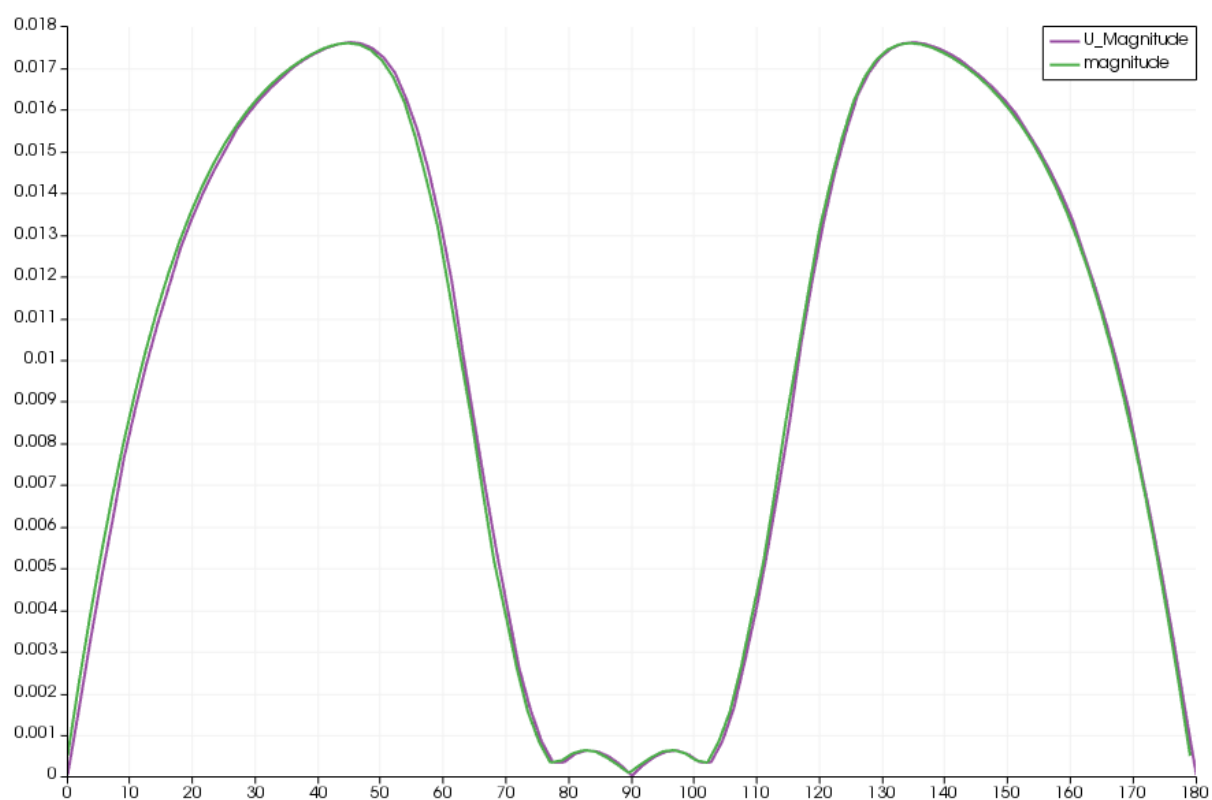
Rysunek 8. Przepływ uzyskany za pomocą Chimery.



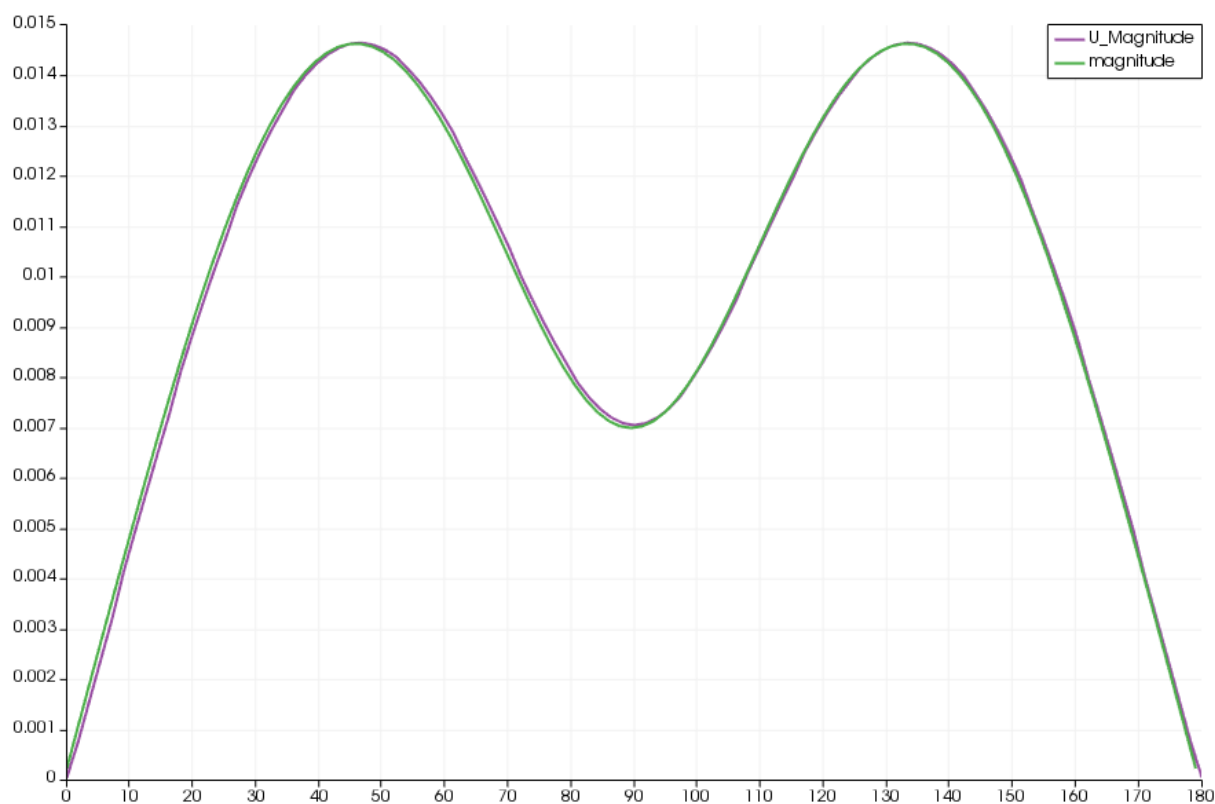
Rysunek 9. Przepływ na siatce pomocniczej.



Rysunek 10. Przekrój przez środek przeszkody.



Rysunek 11. Przekrój za przeszkodą.



Rysunek 12. Przekrój w połowie kanału

6. Podsumowanie

Jak widać na umieszczonych wynikach, Metoda Chimera pozwoliła na uzyskanie bardzo zbliżonych wyników do tych, uzyskanych na gęstej siatce w TCLB. Chimera pozwala jednak na zaoszczędzenie znacznej ilości pamięci i mocy obliczeniowej, ponieważ w badanym przykładzie zawierała jedynie $520 * 180 + 320 * 320 = 196\,000$ punktów. Siatka TCLB zawierała $520 * 4 * 180 * 4 = 1\,497\,600$ punktów, zatem o cały rząd wielkości więcej.

Nie można jednak niestety stwierdzić, że wyniki te są w pełni satysfakcjonujące. W obszarze interpolacji pojawiają się bardzo małe, ale widoczne zniekształcenia, które mogą być wynikiem nie dość dokładnej interpolacji lub niedoskonałego warunku (11).

Można jednak zdecydowanie stwierdzić, że dla tak dużej oszczędności pamięci i mocy obliczeniowej, metoda ta daje bardzo obiecujące rezultaty i warta jest dalszych badań nad tym zagadnieniem.

7. Przypisy

[1] <https://www.flowkit.com/>

[2] <http://wiki.palabos.org/>

[3] <https://github.com/CFD-GO/TCLB>

[4] Yuanxun Bill Bao, Justin Meskas, *Lattice Boltzmann Method for Fluid Simulations*, Courant Institute of Mathematical Sciences, April 2011