

<http://www.uwm.edu.pl/pathfinder/>

## 2<sup>nd</sup> Workshop on Porous Media

Olsztyn, 28-30 czerwca 2018

<http://www.uwm.edu.pl/wpm/>

### Materiały dydaktyczne na 2<sup>nd</sup> Workshop on Porous Media

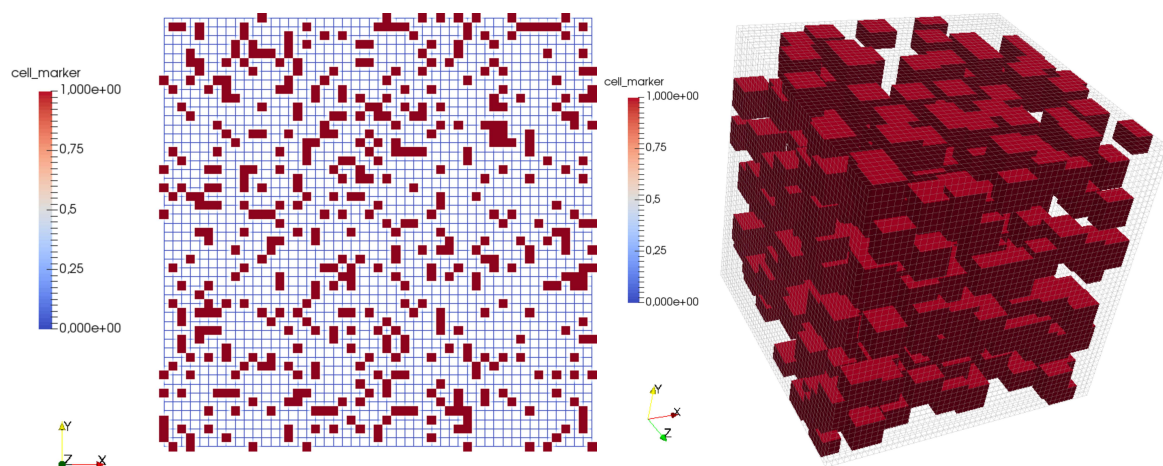
Wojciech Sobieski

Wydział Nauk Technicznych, Uniwersytet Warmińsko-Mazurski w Olsztynie

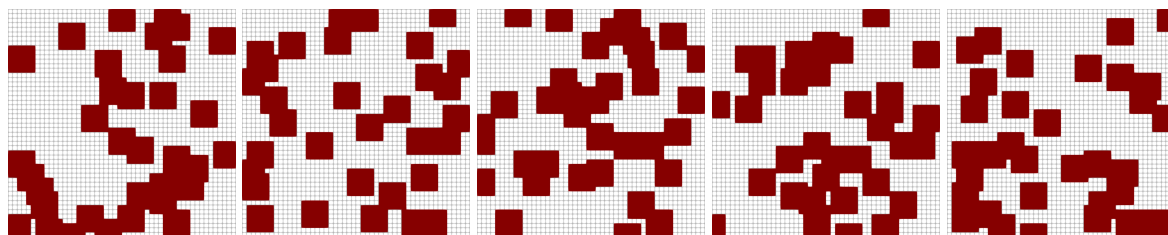
### Wprowadzenie

Pakiet zawiera zbiór czterech programów komputerowych (wraz z kodami źródłowymi napisanymi w języku Fortran 90/95): MATRIX, Refiner, PSA i LBM. Programy działają w systemach operacyjnych Windows oraz UNIX/Linux.

**Program MATRIX** służy do generowania struktur porowych, w przestrzeni dwu- lub trójwymiarowej (Rys. 1), w postaci sieci węzłów, którym przyporządkowuje się stan 0, oznaczający, że węzeł jest dostępny dla gazu sieciowego (jest to część przestrzeni porowej), lub stan 1, co oznacza, że węzeł jest niedostępny dla gazu sieciowego (jest to część szkieletu). Zarówno rozmiar obszaru obliczeniowego (oznaczany zmiennymi  $n_x$ ,  $n_y$  i  $n_z$ ) jak i rozmiar elementu strukturalnego tworzącego szkielet (oznaczony zmiennymi  $s_x$ ,  $s_y$  i  $s_z$ ), mogą być dowolne. Dowolna może być również porowatość (oznaczana jako  $\phi$ ), uzyskiwana przez iteracyjne i losowe dodawanie elementów strukturalnych, o kształcie prostokąta w 2D lub prostopadłościanu w 3D, aż do osiągnięcia wartości żądanej. Każdorazowe uruchomienie programu powoduje uzyskanie innej struktury porowej, nawet przy tych samych ustawieniach programu (Rys. 2).



Rys. 1. Przykłady struktur porowych generowanych programem MATRIX



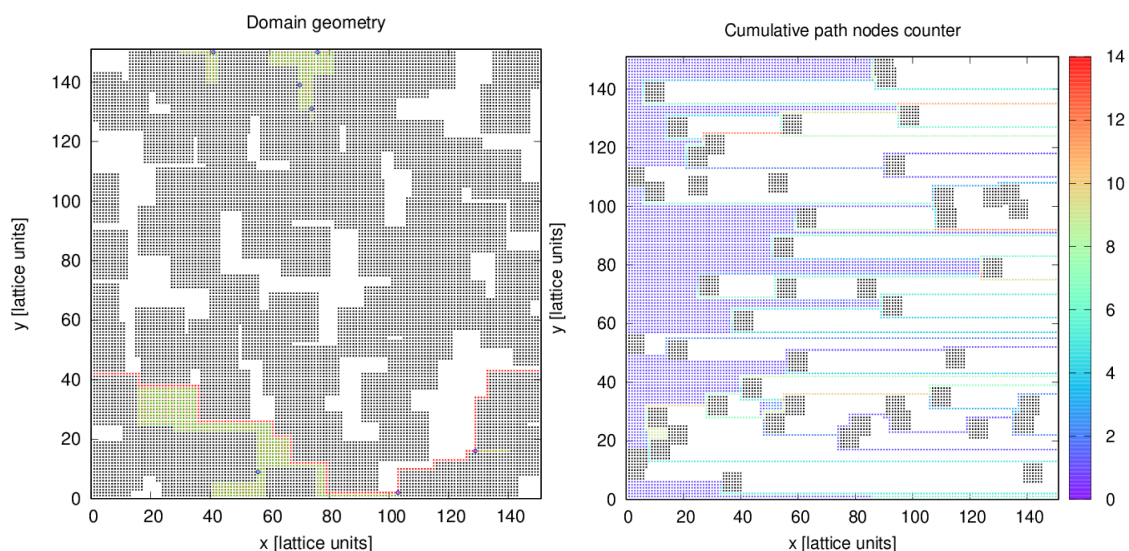
Rys. 2. Przykłady struktur porowych generowanych przy tych samych ustawieniach programu MATRIX

**Program Refiner** służy do zagęszczania sieci węzłów, tak aby przerwy pomiędzy poszczególnymi elementami strukturalnymi nie były zbyt małe. Podczas zagęszczania siatki, każdy pojedynczy węzeł zamieniony sieci zostaje na macierz węzłów o rozmiarach  $n_r \times n_r$  w 2D lub  $n_r \times n_r \times n_r$  w 3D, gdzie  $n_r$  jest tzw. współczynnikiem zagęszczenia sieci.

**Program LBM** służy do przeprowadzania dwuwymiarowych symulacji przepływu gazu Metodą Gazu Sieciowego Boltzmana z wykorzystaniem modelu sieci D2Q9. Ruch gazu inicjowany jest przez jednostkową siłę masową definiowaną oddzielnie dla kierunku X i Y. Na brzegach obszaru zaimplementowane są periodyczne warunki brzegowe. Program działa z wykorzystaniem jednostek sieci i nie bazuje na parametrach fizycznych.

Program LBM powstał na bazie kodu ilustrującego podstawową implementację LBM w języku Fortran, udostępnionego w Internecie przez Mike Sukopa<sup>1</sup>.

**Program PSA** (Path Searching Algorithm) służy do wyszukiwania kanałów porowych przechodzących przez całą długość struktury porowej. Oprócz określania liczby i przebiegu takich kanałów (przykład na Rys. 3, po lewej), program oblicza również sumaryczną liczbę przejść ścieżek przez poszczególne węzły sieci (przykład na Rys. 3, po prawej).



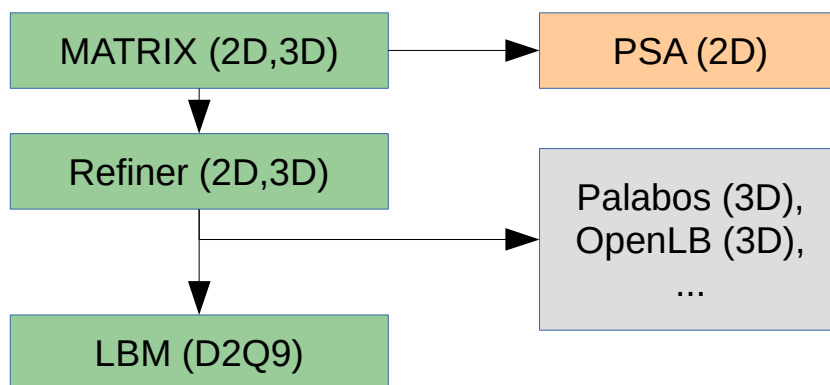
Rys. 3. Przykłady działania algorytmu PSA

**Program run\_all** służy do sekwencyjnego uruchomienia programów MATRIX, Refiner, PSA i LBM. Parametry symulacji definiuje się w kodzie źródłowym programu run\_all, który to należy przed uruchomieniem ponownie skompilować.

1 <http://faculty.fiu.edu/~sukopm/GLY5835/lb.f>

Struktura działań z pakietem jest następująca:

- dla przestrzeni 2D:
  - generacja struktury porowej programem MATRIX;
  - analiza kanałów porowych programem PSA;
  - zagęszczenie siatki programem Refiner;
  - symulacja przepływu gazu sieciowego programem LBM;
- dla przestrzeni 3D:
  - generacja struktury porowej programem MATRIX;
  - zagęszczenie siatki programem Refiner;
  - symulacja przepływu gazu sieciowego programem 3D, np. programem Palabos.



Rys. 4. Struktura połączeń między programami

## Licencja

Warunki korzystania z każdego z elementów projektu PathFinder (w tym programów MATRIX, Refiner, PSA, LBM i run\_all) są następujące:

1. Oprogramowanie jest bezpłatne i może być wykorzystywane w dowolnym celu przez instytucje i osoby prywatne.
2. W przypadku opublikowania jakiegokolwiek materiału, do którego stworzenia przyczyniły się elementy projektu PathFinder, w szczególności artykułu naukowego, jego Autorzy zobowiązani są do przytoczenia odpowiednich artykułów ze strony internetowej projektu PathFinder [1].
3. W przypadku redystrybucji oprogramowania żadna część pakietu instalacyjnego nie może zostać usunięta, w szczególności warunki licencji.
4. Właściciele praw autorskich każdego elementu projektu są Autorzy wskazani w nagłówkach kodu źródłowego lub w dokumentacji.
5. Autorzy projektu nie ponoszą odpowiedzialności za jakiegokolwiek szkody lub straty wynikające z użytkowania oprogramowania i wykorzystania informacji w nim zawartych. Autorzy zapewniają również, że dołożyli wszelkich starań, aby projekt był wolny od błędów i zawierał wiarygodne informacje.

[1] PathFinder: <http://www.uwm.edu.pl/pathfinder/>

## Potrzebne oprogramowanie

Aby korzystać z programów zawartych w pakiecie należy zainstalować i skonfigurować dodatkowe oprogramowanie.

Zalecane oprogramowanie w systemach Windows:

- środowisko Edi<sup>2</sup> (Freeware) – do modyfikacji i kompilacji kodów źródłowych;
- środowisko Gnuplot<sup>3</sup> (GPL) – do wizualizacji wyników z użyciem skryptów PLT (należy skojarzyć pliki o rozszerzeniu PLT z plikiem wykonywalnym wgnuplot.exe, znajdującym się w katalogu BIN tego środowiska);
- środowisko ParaView<sup>4</sup> (GPL) – do wizualizacji wyników z użyciem plików VTK.

Zalecane oprogramowanie w systemach Unix/Linux:

- edytor Kate (GPL) lub inny edytor plików tekstowych – do modyfikacji kodów źródłowych;
- kompilator gfortran (GPL) – do kompilacji kodu źródłowego;
- środowisko Gnuplot (GPL) – do wizualizacji wyników z użyciem skryptów PLT;
- środowisko ParaView (GPL) – do wizualizacji wyników z użyciem plików VTK.

## Kompilacja i uruchamianie

W systemach Windows, i przy użyciu środowiska Edi, kompilacja i uruchamianie kodu odbywa się z poziomu interfejsu programu. Opcjonalnie kompilację można wykonać manualnie po zainstalowaniu i skonfigurowaniu kompilatora gfortran. Kod źródłowy nie był testowany na innych kompilatorach języka Fortran.

W systemach Unix/Linux kompilację dowolnego kodu źródłowego, np. programu LBM, należy wykonać w konsoli systemu operacyjnego z poziomu katalogu, w którym znajduje się dany kod źródłowy, za pomocą następującego polecenia:

```
gfortran lbm.f90 -o lbm.out
```

Skompilowany kod uruchamia się poleceniem:

```
./lbm.out
```

## Geometria przestrzeni obliczeniowej

Geometrię układu – w postaci zapisanej do pliku tekstowego tablicy zer (węzeł wolny) oraz jedynek (węzeł przeszkody) – należy przygotować wcześniej, np. przy użyciu programu MATRIX (Rys. 5). Program MATRIX generuje losowe struktury porowe 2D i 3D o dowolnych rozmiarach na kierunkach X, Y i Z (zmiennie  $n_x$ ,  $n_y$  i  $n_z$ ), dowolnym rozmiarze elementu strukturalnego tworzącego pojedynczą przeszkodę ( $s_x$ ,  $s_y$  i  $s_z$ ) oraz dowolnej porowatości ( $\phi$ ). Zaleca się, aby porowatość struktury porowej była relatywnie duża, tak aby uniknąć sytuacji, w której w głównym kierunku przepływu nie ma kanałów, którymi gaz może się przemieszczać.

2 <http://pracownicy.uwm.edu.pl/wojsob/>

3 <https://sourceforge.net/projects/gnuplot/>

4 <https://www.paraview.org/>

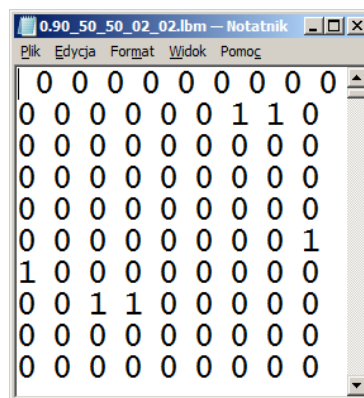
Pliki generowane w programie MATRIX posiadają nazwy tworzone wg następującego kodu (wariant dla struktury 3D; w przypadku geometrii 2D nie ma elementu nr 4 oraz 7):

$\phi\_nx\_ny\_nz\_sx\_sy\_sz.lmb$

gdzie:  $\phi$  – porowatość,  $n_x$ ,  $n_y$ ,  $n_z$  – rozmiary sieci na poszczególnych kierunkach przestrzennych;  $s_x$ ,  $s_y$ ,  $s_z$  – rozmiary elementów tworzących przeszkody na poszczególnych kierunkach sieci.

Istotne jest, że program LBM wykorzystuje tę strukturę do odczytania rozmiarów siatki obliczeniowej.

Należy podkreślić, że generalnie geometrię obszaru obliczeniowego można definiować w dowolny sposób – ważne jest jedynie, aby liczba zer i jedynek w pliku geometrii odpowiadała zadeklarowanej wielkości siatki obliczeniowej, czyli liczbie równej  $n_x \cdot n_y$ . Istotne jest również to, że zera i jedynki powinny być umieszczone w pliku w sposób ciągły – bez umieszczania, np. na końcach kolumn lub wierszy – znaków końca linii.



Rys. 5. Przykład zawartości pliku z geometrią przestrzeni obliczeniowej

Siatkę wygenerowaną programem MATRIX można dodatkowo zagęścić, tak aby zwiększyć szerokość szczelin występujących pomiędzy poszczególnymi przeszkodami. Stopień zagęszczenia sieci (parametr  $n_r$ ) powinien wynosić co najmniej cztery. Dla szczelin o szerokości mniejszej od 4 węzłów, algorytm obliczeniowy nie działa w pełni poprawnie. Po zagęszczeniu sieci węzłów, geometria zapisywana jest w pliku tekstowym o takiej samej nazwie i strukturze, ale z rozszerzeniem „\*.ref”. Po zagęszczeniu rozmiar macierzy wynosi  $(n_x \cdot n_r) \cdot (n_y \cdot n_r)$ .

## Konfiguracja symulacji

Program LBM ma charakter dydaktyczny i zawiera jedynie podstawowe funkcje. Aby skonfigurować model należy edytować tekstowy plik konfiguracyjny „lmb.cfg” i zdefiniować:

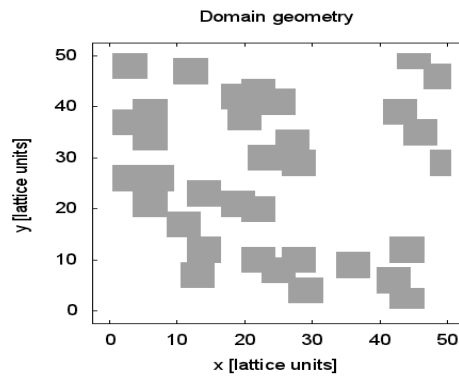
- rodzaj pliku z geometrią: 1 – plik \*.lmb (bez zagęszczenia); 2 – plik \*.ref (z zagęszczoną siatką);
- siłę masową wymuszającą ruch gazu na kierunkach X (zmienna „gx”) oraz Y (zmienna „gy”) – wartość domyślna to 0.0005 na obu kierunkach (zmniejszać dla większych siatek);
- czas relaksacji (zmienna „tr”) – wartość domyślna wynosi 10 (zwiększać dla większych siatek);
- liczbę iteracji – wartość domyślna to 1000;
- mnożnik skali wektorów prędkości (zmienna „vs”) – wartość domyślna to 500;
- użycie trybu LIVE „T” (True) lub „F” (False).

W programie zaimplementowany jest warunek periodyczny na obu kierunkach przestrzeni.

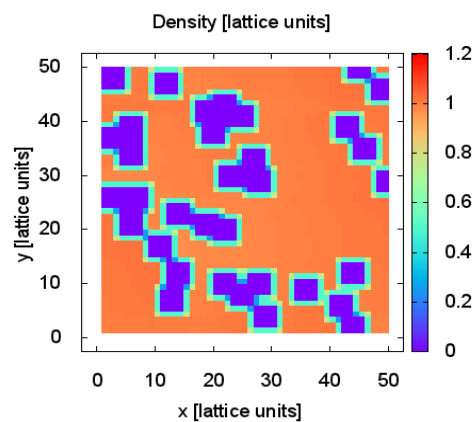


Po przygotowaniu geometrii i skonfigurowaniu modelu można rozpocząć obliczenia.

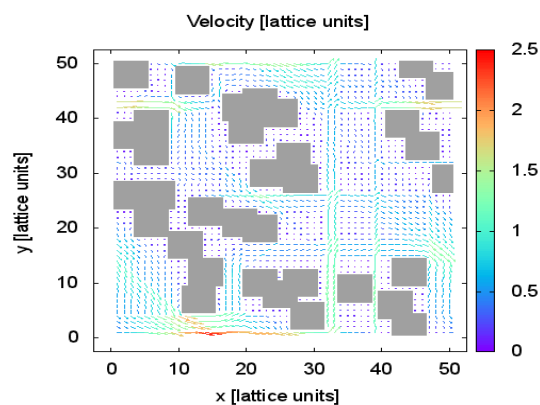
Po zakończeniu symulacji powinien uruchomić się automatycznie skrypt PLT przedstawiający podstawowe wyniki obliczeń (Rys. 6, 7 oraz 8).



Rys. 6. Przykład wizualizacji geometrii obszaru obliczeniowego

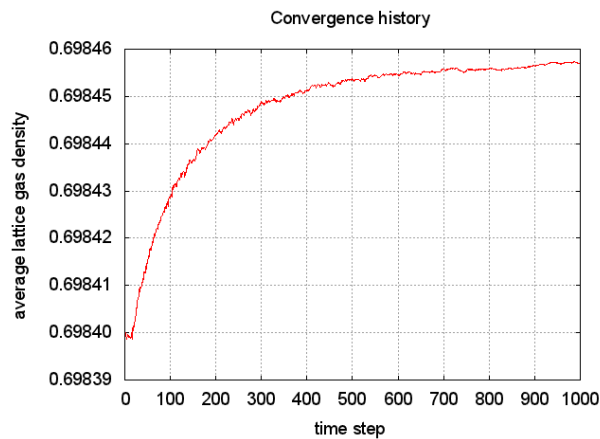


Rys. 7. Przykład wizualizacji rozkładu gęstości gazu sieciowego

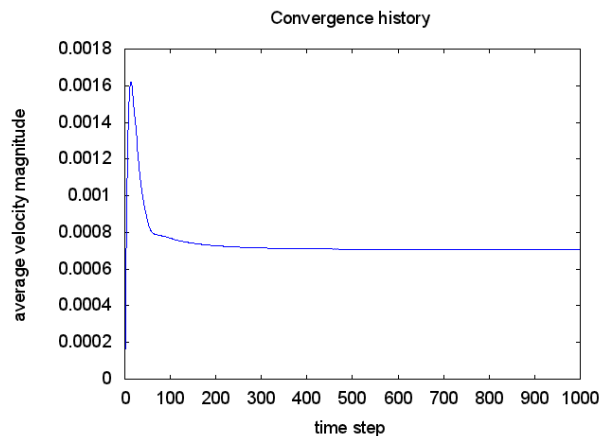


Rys. 8. Przykład wizualizacji pola prędkości (wektory przeskalowane o współczynnik  $vS$  – patrz kod źródłowy)

Po zakończeniu obliczeń powinien się również uruchomić drugi skrypt PLT przedstawiający zmianę średnich wartości gęstości gazu sieciowego (Rys. 9) oraz modułu prędkości (Rys. 10).



Rys. 9. Przykład wizualizacji zmian średniej gęstości gazu sieciowego



Rys. 10. Przykład wizualizacji zmian średniego modułu prędkości

Skrypty PLT można również uruchamiać w późniejszym okresie z poziomu katalogu głównego programu LBM.

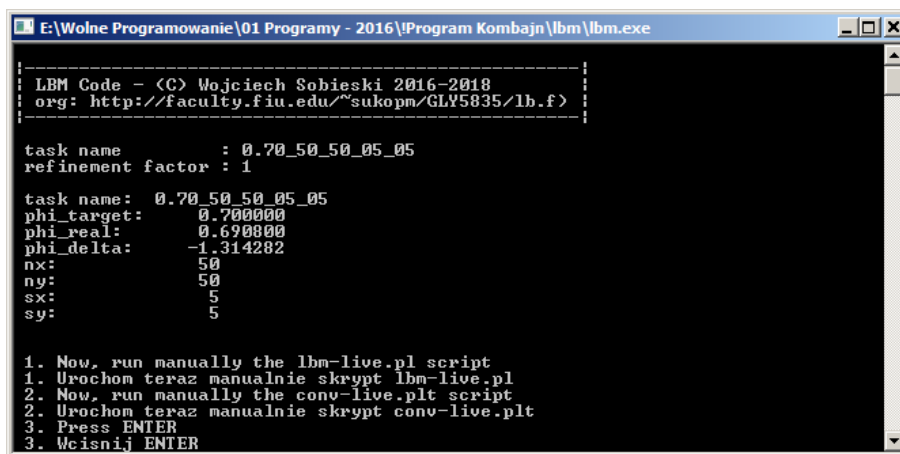
Wszystkie pliki z danymi, skrypty PLT oraz skrypty powłoki systemu operacyjnego generowane są automatycznie podczas pracy programu. Stosowne procedury zawarte są w plikach „\*.inc”.

## Tryb LIVE

Oprócz trybu normalnego, opisanego wyżej, symulacje można uruchomić w tzw. trybie LIVE, pozwalającym na obserwację przebiegu obliczeń. W tym celu należy wykonać następujące działania:

- zmienić w pliku „lbm.cfg” wartość zmiennej „live” z „F” (False) na „T” (True);
- zapisać zmiany w pliku konfiguracyjnym programu LBM;
- uruchomić program i postępować wg wskazówek pojawiających się na ekranie (Rys. 11).

Generalnie chodzi o to, że w trybie LIVE odpowiednie skrypty PLT (jeden z wizualizacją pola prędkości i jeden z wizualizacją procesu zbieżności) muszą być uruchomione przed rozpoczęciem się obliczeń. W trybie LIVE można uruchomić tylko jeden skrypt – np. ten z wizualizacją procesu zbieżności obliczeń.



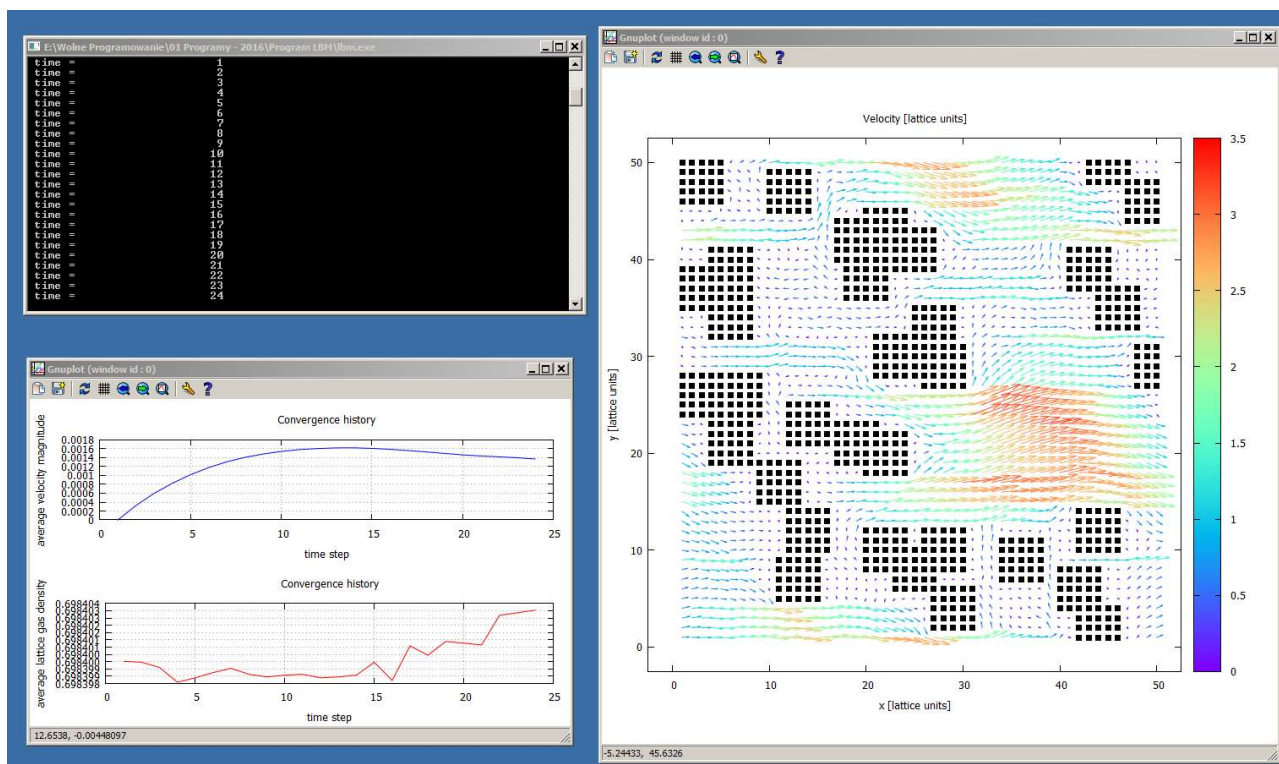
Rys. 11. Komunikaty wyświetlane w trybie LIVE

Jakość wizualizacji w trybie LIVE zależy od mocy komputera oraz jakości karty graficznej. Dla większych siatek mogą wystąpić problemy z odświeżaniem rysunków (szczególnie tego z polem prędkości).

Ze względu na problem współużytkowania tych samych plików przez dwie aplikacje (program LBM oraz środowisko Gnuplot), w niektórych systemach operacyjnych tryb LIVE może nie działać poprawnie.

Czas symulacji w trybie LIVE jest znacznie dłuższy niż w trybie normalnym, ze względu na konieczność zapisywania plików z wynikami obliczeń w każdym kroku czasowym. W trybie normalnym czynność ta jest robiona raz, po zakończeniu obliczeń.

Przykład efektu działania trybu LIVE widoczny jest na Rys. 12.



Rys. 12. Przykład działania trybu LIVE



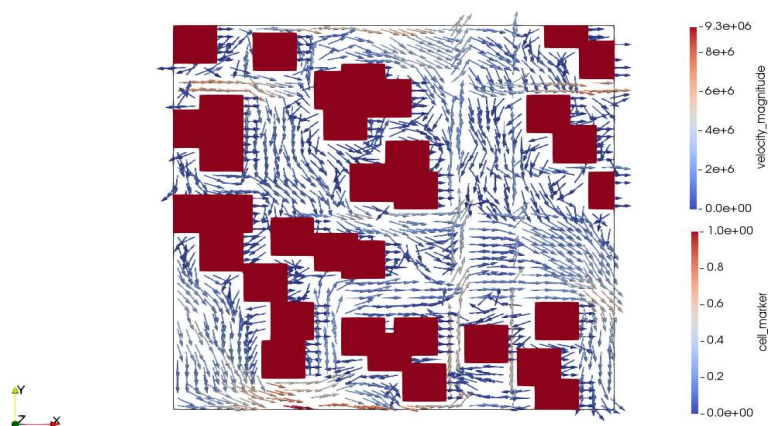
## Wizualizacja w środowisku ParaView

Po zakończeniu głównej pętli obliczeniowej, w programie LBM zapisywane są trzy pliki VTK, bazujące na nieco innej strukturze danych: RECTILINEAR\_GRID (lbm-rg.vtk), STRUCTURED\_GRID (lbm-sg.vtk) oraz STRUCTURED\_POINTS (lbm-sp.vtk).

Aby wykonać podstawową wizualizację w programie ParaView wykonaj następujące działania:

1. Otwórz plik VTK i potwierdź swój wybór klawiszem „Apply”.
2. Opcję „Representation” ustaw na „Outline”.
3. Opcję „Coloring” ustaw na „Solid Color”.
4. Użyj narzędzia „Glyph” (do wizualizacji przeszkód) oraz ustaw:
  - opcję „Glyph Type” na „Box”;
  - opcję „Scale Mode” na „scalar”;
  - opcję „Scale Factor” na 1 (lub inną wartość jeśli zajdzie potrzeba);
  - opcję „Glyph Mode” na „All Points”.
5. Potwierdź swoje wybory klawiszem Apply.
6. Użyj narzędzia „Glyph” (do wizualizacji pola prędkości) oraz ustaw:
  - opcję „Glyph Type” na „Arrow”;
  - opcję „Scale Mode” na „off”;
  - opcję „Scale Factor” na 2 (lub inną wartość jeśli zajdzie potrzeba);
  - opcję „Glyph Mode” na „Uniform Spatial Distribution”;
  - opcję „Coloring” na „velocity magnitude”.
7. Potwierdź swoje wybory klawiszem „Apply”.

Przykład wizualizacji wykonanej wg powyższego schematu widoczny jest na Rys. 13.



Rys. 13. Przykład wizualizacji pola prędkości (wektory przeskalowane o współczynnik  $VS$  – patrz kod źródłowy)

## Literatura:

Lista publikacji znajduje się na stronie projektu PathFinder: <http://www.uwm.edu.pl/pathfinder/>.