

Prosty model przepływu ciepła z wykorzystaniem standardu MPI do obliczeń równoległych

B. Nowak, M. Wawrzak, M. Lachman

8 lutego 2016

- 1 MPI - Message Passing Interfacei
 - Opis standardu MPI
 - Koncepcja standardu
- 2 Prosty model przepływu ciepła
 - Założenia modelu
- 3 Implementacja i założenia projektu
 - Założenia projektu
 - Implementacja
 - Zawartość projektu
- 4 Podsumowanie

Ogólnie o MPI

MPI (Message Passing Interfacei) to nazwa standardu biblioteki przesyłania komunikatów dla potrzeb programowania równoległego. Pod skrótem MPI kryje się tylko formalna specyfikacja interfejsu, nie jest to nazwa żadnego konkretnego pakietu oprogramowania. Najbardziej znaną implementacją MPI jest MPICH, pochodzący z Argonne National Laboratory i rozwijany przez grupę pracowników działu matematyki i informatyki tej instytucji. Dostępna jest wersja zarówno na platformy UNIX-owe, jak i Windows.

Główne założenia

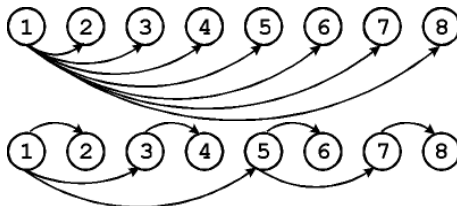
MPI realizuje model przetwarzania współbieżnego zwany MIMD (Multiple Instruction Multiple Data), a dokładniej SPMD (Single Program Multiple Data). Zakłada on, że ten sam kod źródłowy wykonuje się jednocześnie na kilku maszynach i procesy mogą przetwarzać równocześnie różne fragmenty danych, wymieniając informacje przy użyciu komunikatów.

Standard MPI umożliwia:

- Wymianę komunikatów między procesami
- Uzyskiwanie informacji o środowisku
- Kontrolę nad systemem

Komunikacja w MPI

Główną zaletą MPI przy bardziej złożonych schematach wymiany danych jest ukrywanie przed programistą szczegółów implementacyjnych oraz możliwość optymalizacji ścieżki przepływu danych.



Rysunek: Różne formy komunikacji między wątkami. MPI umożliwia dostosowanie przepływu danych użytkownikowi.

Biblioteka MPICH

Do implementacji poszczególnych funkcji standardu MPI posłużyła w naszym projekcie biblioteka MPICH. Kod programu obliczeniowego napisany został w języku C.

Funkcje sterujące przepływem komunikatów:

- MPIRecv - funkcja pozwalająca odbierać kolejno komunikaty
- MPISend - funkcja wywołująca komunikaty do poszczególnych wątków

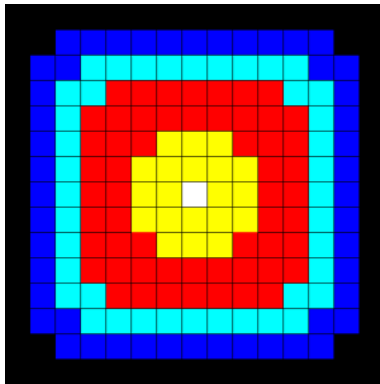
Model przepływu ciepła - założenia

Do prezentacji możliwości wykorzystania biblioteki MPICH wybraliśmy proste zagadnienie symulujące przepływ ciepła na siatce. Zakłada się istnienie dwuwymiarowej siatki która dzieli równomiernie rozpatrywany obszar. Na początku symulacji generujemy początkowy rozkład temperatury na siatce, a następnie obliczane są kolejne temperatury komórek w kolejnych krokach czasowych.

Model przepływu ciepła - założenia

Do prezentacji możliwości wykorzystania biblioteki MPICH wybraliśmy proste zagadnienie symulujące przepływ ciepła na siatce. Zakłada się istnienie dwuwymiarowej siatki która dzieli równomiernie rozpatrywany obszar. Na początku symulacji generujemy początkowy rozkład temperatury na siatce, a następnie obliczane są kolejne temperatury komórek w kolejnych krokach czasowych.

Model przepływu ciepła - założenia



Rysunek: Początkowa siatka temperatur do symulacji.

Model przepływu ciepła - założenia

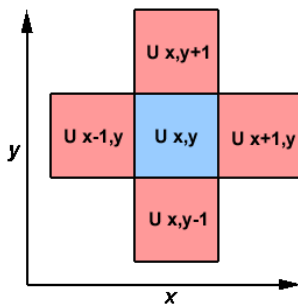
Obliczanie temperatur dla kolejnych elementów siatki następuje zgodnie ze wzorem:

$$U_{x,y} = U_{x,y} + C_x * (U_{x+1,y} + U_{x-1,y} - 2 * U_{x,y}) + C_y * (U_{x,y+1} + U_{x,y-1} - 2 * U_{x,y})$$

C_x oraz C_y to współczynniki określające szybkość z jaką następuje przepływ ciepła od poszczególnych komórek.

Model przepływu ciepła - założenia

W każdym kroku czasowych temperatura każdej komórki obliczana jest na podstawie swoich najbliższych sąsiadów:



Rysunek: Oznaczenia i numeracja komórek siatki.

Model przepływu ciepła - założenia

Trzeba zaznaczyć, że przyjęty model rozchodzenia się ciepła jest bardzo uproszczony i fizycznie nie jest poprawny. Posłużył on jednak do prostego przedstawienia jak działa program oparty na obliczeniach równoległych przy wykorzystaniu biblioteki MPICH.

Założenia projektu

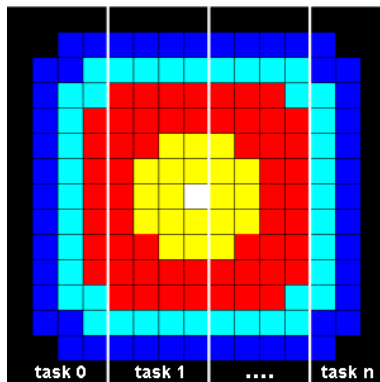
Część obliczeniowa odpowiedzialna za obliczenia równoległe napisana została w języku C i jest realizowana w oparciu o wczesniej przedstawiony model przepływu ciepła. Pierwszym krokiem jaki wykonuje program to inicjalizacja biblioteki i wywołanie wątku głównego, czyli wątku odpowiedzialnego o rozsyłanie komunikatów do poszczególnych wątków pobocznych i zbieranie danych wyjściowych - danych po obliczeniach.

Założenia projektu

Główne założenia dotyczące sposobu rozwiązywania problemu:

- Zostaje zainicjalizowana siatka początkowa, na której brzegach temperatura jest stała i wynosi 0.0
- W zależności od ilości wybranych wątków pobocznych, wątek główny dzieli obszar siatki (danych) na równe części (w miarę możliwości) i wysyła odpowiednie dane do poszczególnych wątków pobocznych (roboczych)

Założenia projektu



Rysunek: W zależności od ilości przyjętych wątków roboczych wątek główny dzieli obszar siatki na odpowiednie podobszary na, których każdy z wątków roboczych będzie działał.

Założenia projektu

Główne założenia dotyczące sposobu rozwiązywania problemu:

- W trakcie obliczeń gdy wątek roboczy napotka komórkę, która do obliczeń wymaga znajomości komórki należącej do innego wątku następuje komunikacja.
- Wymagana jest odpowiednia współpraca między poszczególnymi wątkami, aby odpowiednio zoptymalizować pracę.
- Po zakończonym procesie obliczania wątek roboczy przesyła informacje zwrotną do wątku głównego.
- Wątek główny oczekuje na wyniki od każdego wątku pobocznego. Gdy to nastąpi aktualizuje siatkę temperatur, a proces powtarza się.

Implementacja

```
find out if I am MASTER or WORKER

if I am MASTER
  initialize array
  send each WORKER starting info and subarray
  receive results from each WORKER

else if I am WORKER
  receive from MASTER starting info and subarray

do t = 1, nsteps
  update time
  send neighbors my border info
  receive from neighbors their border info

  update my portion of solution array

end do

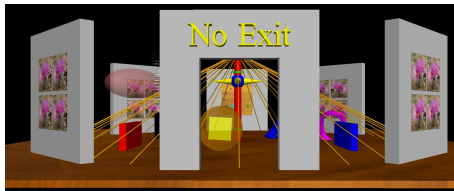
send MASTER results

endif
```

Rysunek: Algorytm postępowania w celu dokonania obliczeń. Czerwony kolor odpowiada części kodu, która odpowiedzialna jest za przetwarzanie równoległe.

Zawartość projektu

Po wykonaniu obliczeń przez program główny zapisywane są dane wynikowe z siatką temperatur dla zadanej ilości kroków. Następnie został stworzony skrypt w pythonie, który wizualizuje otrzymane dane. Korzysta on z biblioteki visual python, która umożliwia graficzne przedstawienie zagadnienia.



Zawartość projektu

W skład projektu wchodzi dwa skrypty w pythonie, główny plik C z częścią obliczeniową i plik Makefile. Plik Makefile oferuje kilka możliwości uruchomienia programu.



Rysunek: Katalog główny projektu. Do projektu został stworzony plik Makefile, który ułatwia uruchamianie i kompilacje programu.

Podsumowanie

Posumowując, udało się nam przeprowadzić symulację rozchodzenia się ciepła, na siatce z wykorzystaniem standardu MPI i bibliotekę go wykorzystującą MPICH.

Podsumowanie

Dziękujemy za uwagę.