

Algorytmy Równoległe i Rozproszone

Część I - Wprowadzenie i Sieci porównujące

Łukasz Kuszner
pokój 209, WETI

<http://www.sphere.pl/~kuszner/>
kuszner@sphere.pl

Oficjalna strona wykładu
<http://www.sphere.pl/~kuszner/ARiR/>
Wykład 15 godzin, Projekt 15 godzin

2007/08

Plan wykładu

Literatura

Sieci porównujące

PRAM - przykład

Model rozproszony

Strona główna

Strona tytułowa



Strona 1 z 14

Powrót

Full Screen

Zamknij

Koniec

1. Ogólny plan wykładu

- Wstęp
- Sieci porównujące,
- Układy kombinacyjne,
- Algorytmy w modelu PRAM,
- Algorytmy w modelu rozproszonym.

Strona główna

Strona tytułowa

◀◀ ▶▶

◀ ▶

Strona 2 z 14

Powrót

Full Screen

Zamknij

Koniec

2. Literatura

- T. H. Cormen, C. E. Leiserson and R. L. Rivest, „Introduction to Algorithms”, The MIT Press/McGraw-Hill Company, 1990 (wydanie polskie WNT).
- Raymond Greenlaw, H. James Hoover, Walter L. Ruzzo „Limits to Parallel Computation: P-Completeness Theory”, Oxford University Press, 1998.
- Hagit Attiya „Lecture Notes for Course Distributed Algorithms”, 1994.

- C. Xavier, S. S. Iyengar, „Introduction to Parallel Algorithms”, Wiley-IEEE, 1998.
- Gerard Tel, „Introduction to Distributed Algorithms”, Cambridge University Press, 2nd edition, 2000.
- Shlomi Dolev. Self-Stabilization. MIT Press, 2000.

- J. Jaja, „An Introduction to Parallel Algorithms”, Addison-Wesley, Reading, MA, 1992.
- S.G. Akl, „The Design and Analysis of Parallel Algorithms”, Prentice-Hall, 1989.
- Lecture Notes Repository: [Uniwersytet w Paderborn](#).

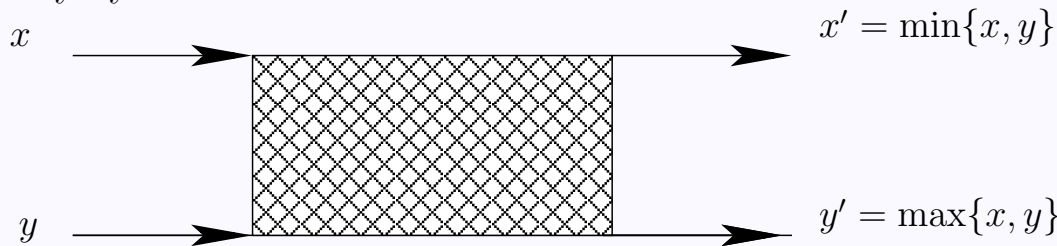
- Ananth Grama, Anshul Gupta, George Karypis, Vipin Kumar „Introduction to Parallel Computing”, Addison Wesley, 2003.
- Hagit Attiya, Jennifer Welch „Distributed Computing: Fundamentals, Simulations, and Advanced Topics”, 2nd Edition.
- Guy E. Blelloch, Bruce M. Maggs: Parallel Algorithms. The Computer Science and Engineering Handbook, 1997: 277-315.

3. Sieci porównujące - przykład

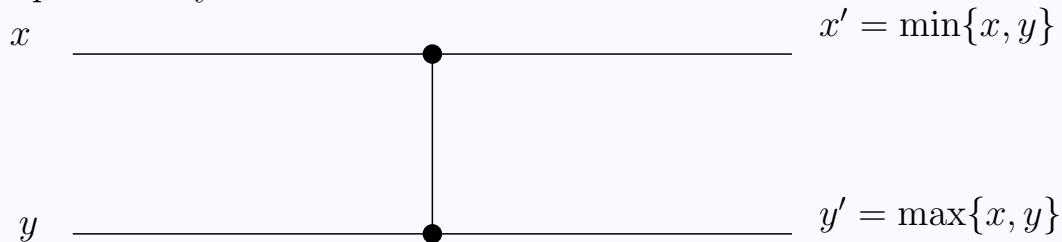
Sieć porównująca składa się z połączonych ze sobą komparatorów.

3.1. Schematy komparatora

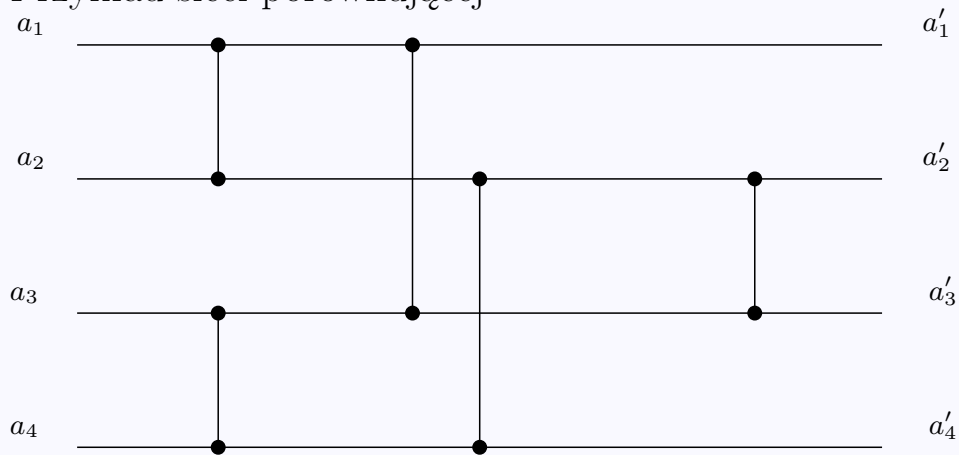
zwykły:



uproszczony:



Przykład sieci porównującej



Plan wykładu

Literatura

Sieci porównujące

PRAM - przykład

Model rozproszony

Strona główna

Strona tytułowa

◀▶

◀▶

Strona 9 z 14

Powrót

Full Screen

Zamknij

Koniec

Ćwiczenie 1

Sprawdź, że sieć porównująca z poprzedniego slajdu jest siecią sortującą.

4. PRAM - przykład

4.1. Iloczyn skalarny

We: Tablice współrzędnych $a[1 : n]$ i $b[1 : n]$

Wy: Liczba będąca iloczynem skalarnym wektorów a i b .

Algorytm 1: Iloczyn skalarny

```
1: for  $i = 1$  to  $n$  in parallel do  
2:    $c_i = a_i * b_i$   
3: end for  
4:  $p = n/2$   
5: while  $p > 0$  do  
6:   for  $i = 1$  to  $p$  in parallel do  
7:      $c_i = c_i + c_{i+p}$   
8:   end for  
9:    $p = p/2$   
10: end while
```

Strona główna

Strona tytułowa

◀▶

◀▶

Strona 10 z 14

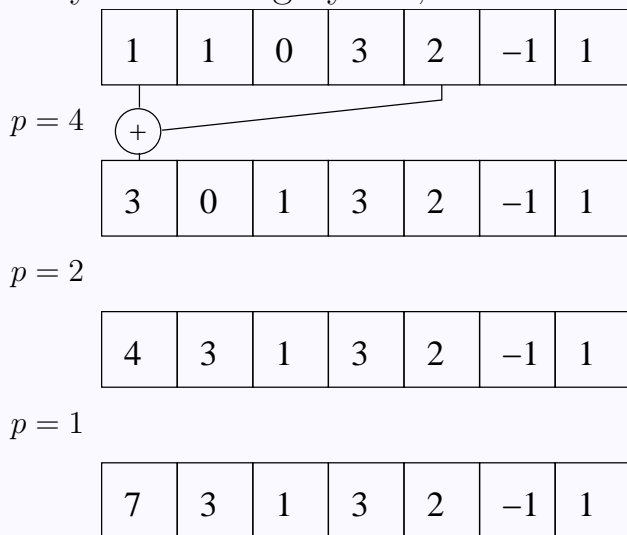
Powrót

Full Screen

Zamknij

Koniec

Przykład dla algorytmu, dla $n = 7$.



Strona główna

Strona tytułowa

◀▶

◀▶

Strona 11 z 14

Powrót

Full Screen

Zamknij

Koniec

5. Model rozproszony - przykład

5.1. Rozproszony algorytm dla problemu: „wybór lidera”

Przypuśćmy, że mamy rozproszony system asynchroniczny, w którym jednostki obliczeniowe tworzą pierścień.

Problem polega na skonstruowaniu algorytmu, który pozwoli wyróżnić jeden z procesorów. Oczekujemy, że po zakończeniu działania algorytmu dokładnie jeden procesor zasygnalizuje „jestem liderem”, a wszystkie pozostałe „nie jestem liderem”.

Przypuśćmy, że każdy z procesorów ma unikalny identyfikator. Procesory tworzą pierścień, możemy więc mówić o lewym i prawym sąsiedzie. Inicjalnie każdy z procesorów wysyła wiadomość do swojego lewego sąsiada, w której zapisuje swój identyfikator.

Następnie każdy procesor odbierając wiadomość od prawego sąsiada przesyła ją dalej, jeśli zapisany na niej identyfikator jest większy od jego własnego identyfikatora i kasuje, jeśli jest mniejszy.

Procesor, który otrzymał z powrotem swój własny identyfikator ogłasza się liderem i wysyła wiadomość kończącą działanie algorytmu.

[Strona główna](#)[Strona tytułowa](#)[◀◀](#)[▶▶](#)[◀](#)[▶](#)[Strona 13 z 14](#)[Powrót](#)[Full Screen](#)[Zamknij](#)[Koniec](#)

Ćwiczenie 2

Uzasadnij, że liczba wiadomości wysłanych w trakcie działania algorytmu jest $O(n^2)$.