



Problemy równoległej  
optymalizacji dyskretnej

**PROBLEMY RÓWNOLEGŁEJ  
OPTYMALIZACJI DYSKRETNEJ**

Redaktorzy:

**Leon Słomiński**

**Ignacy Kaliszewski**

# 1. Wprowadzenie

Niniejsza książka prezentuje wyniki prac prowadzonych w Zakładzie Programowania Matematycznego Instytutu Badań Systemowych PAN w latach 1986-92, w ramach tematu badawczego "Optymalizacja na strukturach kombinatorycznych z uwzględnieniem obliczeń równoległych". Podjęcie tego tematu, którym kierował pierwszy z współredaktorów, poprzedziło roczne wspólne seminarium Zakładu Programowania Matematycznego IBS PAN oraz Pracowni Metod Numerycznych Instytutu Podstaw Informatyki PAN. Niektóre pomysły dyskutowane podczas seminarium znalazły swoje odbicie w treści tej książki.

Celem książki jest przybliżenie polskiemu Czytelnikowi problematyki obliczeń równoległych w zadaniach optymalizacji, a szczególnie w zadaniach optymalizacji dyskretnej oraz uzupełnienie, chociażby częściowe, luki jaka jest w tym zakresie zauważalna w krajowej literaturze. Z publikacji w języku polskim poświęconych w całości lub w części metodom i algorytmom równoległym optymalizacji dyskretnej można wymienić pozycje [1 + 3]. Postawiony cel chcemy osiągnąć przez zaprezentowanie wyboru zadań optymalizacyjnych z zakresu naszych zainteresowań i przedstawienie, na ich przykładzie, problemów związanych z konstrukcją algorytmów równoległych. Problemy te ukazujemy na tle znacznie szerszego wachlarza zagadnień związanych z obliczeniami równoległymi.

Książka ma następujący układ. Rozdział drugi zawiera podstawowe pojęcia związane z obliczeniami równoległymi i z maszynami równoległymi. Przedstawiono w nim najważniejsze modele obliczeń równoległych, typowe architektury maszyn równoległych i ich charakterystyki, a także podstawowe wyniki teorii złożoności dla

algorytmów równoległych. Dopełnieniem tego rozdziału jest omówienie nowatorskiej realizacji sprzętowej - transputera, którego sieci wydają się szczególnie przydatne do rozwiązywania zadań optymalizacji dyskretnej za pomocą ogólnych i wąsko ukierunkowanych algorytmów równoległych. W rozdziale trzecim przedstawiono algorytmy równoległe dla rozwiązywania zadania wyznaczania dendrytu minimaxowego w grafie skierowanym z wagami na łukach. W celu pokazania związku, który zachodzi między złożonością algorytmu i wybranym modelem maszyny równoległej, użyto różnych modeli maszyny i różnych pierwowzorów algorytmów sekwencyjnych. Rozdział czwarty przedstawia algorytmy wyznaczania elementu maksymalnego w skończonym zbiorze wektorów oraz opis ich realizacji na sieci transputerów, wraz z wynikami testów numerycznych. W rozdziale piątym opisano asynchroniczny algorytm równoległy, do rozwiązywania algebraicznego zagadnienia  $k$  - przydziału, przedstawiono realizację tego algorytmu na sieci transputerów oraz wyniki eksperymentu numerycznego. Kolejny, szósty rozdział prezentuje język programowania równoległego MODEST. Jest to język ogólnego zastosowania zawierający mechanizmy do uruchamiania procesów i kontroli przebiegu obliczeń równoległych. Dwa dodatki zawierają: Dodatek 1 - Polsko - angielski słownik nazw i pojęć z zakresu obliczeń równoległych, Dodatek 2 - Listę artykułów i raportów ZPM IBS PAN, które ukazały się w latach 1986 - 1992 i które dotyczą problematyki tej książki.

Warszawa, styczeń 1993.

Leon Słomiński  
Ignacy Kaliszewski

## Literatura

1. Błażewicz J. (1988): *Złożoność obliczeniowa problemów kombinatorycznych*. WNT, Warszawa.
2. Słomiński L., Kaliszewski I. (1988): "Problemy obliczeń równoległych". *Prace IBS PAN*, 168, Warszawa.
3. Sysło M.M. (1985): "Maszyny i algorytmy równoległe". *Raport Instytutu Informatyki Uniwersytetu Wrocławskiego*, 154, Wrocław.

## Dodatek 1.

### POLSKO - ANGIELSKI SŁOWNIK NAZW I POJĘĆ Z DZIEDZINY OBLICZEŃ RÓWNOLEGLYCH.

*Algorytm asynchroniczny - asynchronous algorithm.* - Typ algorytmu dla maszyn MIMD, w którym procesory mogą korzystać z aktualnych danych bez potrzeby określania momentów synchronizacji.

*Algorytm potokowy - pipelined algorithm.* - Jest to programowy odpowiednik architektury potokowej. Algorytm taki podzielony jest na uporządkowany ciąg segmentów w ten sposób, że wyjście jednego segmentu jest wejściem następnego.

*Algorytm równoległy - parallel algorithm.* - Algorytm zaprojektowany do wykonania na maszynie równoległej.

*Algorytm systoliczny - systolic algorithm.* - Algorytm równoległy przeznaczony do realizacji na maszynie systolicznej, charakteryzujący się regularnością i synchronizmem prostych działań wykonywanych na regularnych strumieniach danych.

*Bardzo duży stopień scalenia (BDSS).* - *Very Large Scale of Integration (VLSI).* - Klasa scalonych układów elektronicznych, o liczbie elementów w module przekraczającej 100000.

*Drzewa ortogonalne - orthogonal trees.*

- 1. Sieć utworzona z  $2n$  ( $n=2^q$ ,  $q \geq 1$ ) pełnych drzew binarnych. Liście drzew tworzą tablicę o wymiarze  $n \times n$ ; nad każdym wierszem (każdą kolumną), rozpstarte jest pełne drzewo binarne. W ten sposób każda para drzew ma jeden wspólny liść.
- 2. Architektura maszyny równoległej, której procesory i połączenia między nimi odpowiadają węzłom i połączeniom drzewa ortogonalnego.

*Drzewo binarne - binary tree.*

- 1. Drzewo skierowane, w którym każdy węzeł ma co najwyżej dwa następniki. Drzewo binarne nazywa się pełnym, gdy istnieje liczba naturalna  $q$  taka, że każdy węzeł o głębokości mniejszej od  $q$  ma dwa następniki oraz węzeł o głębokości  $q$  jest liściem. Pełne drzewo binarne ma dokładnie  $2^{q+1} - 1$  węzłów.

- 2. Architektura *maszyny równoległej*, której procesory i połączenia między nimi odpowiadają węzłom i połączeniom drzewa binarnego.

*Drzewo kwadratowe - quadrary tree.*

- 1. *Drzewo skierowane*, w którym każdy węzeł ma nie więcej niż 4 następniki.
- 2. Architektura *maszyny równoległej*, której procesory i połączenia między nimi odpowiadają węzłom i połączeniom drzewa kwadratowego.

*Drzewo skierowane - oriented tree.*

- 1. Acykliczna sieć skierowana spełniająca następujące warunki: istnieje dokładnie jeden węzeł, nazywany korzeniem, do którego nie dochodzi żaden łuk (korzeń nie ma węzłów poprzedzających); istnieje droga (można udowodnić, że jest to droga jedyna), prowadząca od korzenia do każdego węzła; każdy węzeł (z wyjątkiem korzenia) ma dokładnie jeden łuk wchodzący do niego. Jeżeli łuk  $(i, j)$  należy do sieci, to węzeł  $i$  nazywa się poprzednikiem węzła  $j$ , a węzeł  $j$  nazywa się następnikiem węzła  $i$ . Węzeł, który nie ma następników, nazywa się liściem. Głębokością węzła  $i$  w drzewie jest jego odległość (długość drogi mierzona liczbą łuków) od korzenia. Wysokością węzła  $i$  jest maksymalna długość drogi od  $i$  do liścia. Wysokością drzewa jest wysokość jego korzenia.
- 2. Architektura *maszyny równoległej*, której procesory i połączenia między nimi odpowiadają węzłom i połączeniom drzewa skierowanego.

*Drzewo ternarne - ternary tree.*

- 1. *Drzewo skierowane*, w którym każdy węzeł ma nie więcej niż 3 następniki.
- 2. Architektura *maszyny równoległej*, której procesory i połączenia między nimi odpowiadają węzłom i połączeniom drzewa ternarnego.

*Efekt Amdahla - Amdahl effect.* - Wartość przyśpieszenia  $s$  maszyny o  $p$  procesorach, wykonującej ciąg obliczeń o znormalizowanej długości 1, jest ograniczona od góry przez następujące wyrażenie:

$$s \leq 1/(f + (1 - f)/p),$$

gdzie:  $0 \leq f < L$  jest znormalizowaną długością obliczeń sekwencyjnych.

Oznacza to, że jeżeli rośnie rozmiar zadania, a  $f$  nie zmienia się, to wzrost przyśpieszenia można otrzymać przez zwiększenie liczby procesorów  $p$ .

*Efektywność wykorzystania procesorów* - efficiency of processors utilization. - Stosunek przyśpieszenia obliczeń do liczby procesorów użytych w celu uzyskania tego przyśpieszenia.

*Element operacyjny* - processing element. - Podstawowa jednostka wykonawcza w maszynach systolicznych i w maszynach o wysokim stopniu równoległości.

*Exchange* - exchange. - Operacja komunikacyjna w maszynie o architekturze typu *Idealnie Potasowany Stos*, polegająca na przekazaniu jednostki informacji od procesora o numerze  $i$  do procesora, którego numer otrzymuje się przez zanegowanie najmłodszego bitu w rozwinięciu binarnym liczby  $i$ .

*Fork/Join* - Fork/Join. - Konstrukcja językowa stosowana w językach równoległych wyższego poziomu, pozwalająca inicjować/łączyć procesy współbieżne utworzone przez dany proces.

*Funkcja zgodności* - consensus function. - Funkcja rzeczywista, opisująca stan maszyny Boltzmann. Wartość funkcji zgodności zależy od stanu procesorów (pobudzone/niepobudzone) i od wartości funkcji zdefiniowanej na parach wierzchołków będącej miarą potrzeby połączenia danej pary procesorów.

*Gigaflop* - Gigaflop. - Miliard operacji zmienno-przecinkowych na sekundę.

*Głębokość algorytmu równoległego* - parallel algorithm depth. - Złożoność czasowa (nakład czasu) algorytmu równoległego.

*Graf procesu* - process flow graph. - Graf skierowany, którego wierzchołkami są zdarzenia, a łuki pokazują kolejność występowania zdarzeń i ich powiązania logiczne.

*Gwiazda* - star.

- 1. Drzewo skierowane o wysokości 1.
- 2. Architektura maszyny równoległej, której procesory i połączenia między nimi odpowiadają węzłom i połączeniom gwiazdy.



*Hipersześcian - hypercube.*

- 1. Sześcian uogólniony na wymiar  $n > 3$ . Przy numeracji węzłów liczbami naturalnymi od 0 do  $(2^{**}n) - 1$ , bezpośrednio połączone są te węzły, których numery przedstawione rozwinięciem binarnym, różnią się dokładnie jednym bitem.
- 2. Architektura *maszyny równoległej*, której procesory i połączenia między nimi odpowiadają *hipersześcianowi*.

*Hipersześcian cykli - cube connected cycles.* - Sieć i architektura typu *hipersześcian*. Wierzchołkami są zespoły procesorów połączonych w cykl.

*Hipoteza obliczeń równoległych - parallel computation thesis.* - Czas obliczeń *maszyny równoległej* o nieograniczonej liczbie procesorów, jest wielomianowo równoważny zajętości pamięci maszyny sekwencyjnej.

*Idealnie potasowany stos (IPS) - perfect shuffle network.*

- 1. Sieć: tworzy ją  $p = 2^{**}k$ ,  $k \geq 1$ , węzłów ponumerowanych liczbami naturalnymi od 0 do  $p - 1$  i połączonych między sobą w ten sposób, że węzły o numerach:  $i$ ,  $i + p/2$ ,  $2i - 1$  tworzą łuki z węzłami o numerach odpowiednio:  $2i - 1$ ,  $2i$ ,  $2i$ , dla  $i = 1, \dots, p/2$ .
- 2. Architektura *maszyny równoległej*, w której procesory i połączenia między nimi odpowiadają węzłom i łukom sieci *IPS*.

*Jednostka funkcyjna - functional unit.* - Wspecjalizowany układ półprzewodnikowy (sumator, mnożnik, komparator), zwykle kaskadowy, stosowany w dużych komputerach, przede wszystkim wektorowych, dzięki czemu podstawowe operacje mogą być potokowane i tym samym wykonywane znacznie szybciej.

*Język komunikatów - message oriented language.* - Język programowania realizujący koncepcję obliczeń opartą na wymianie komunikatów między niezależnymi procesami.

*Język proceduralny - procedure oriented language.* - Język programowania stosujący zmienne globalne do komunikowania i synchronizacji procesów.

*Język równoległy - parallel language.* - Język programowania *maszyn równoległych* będący adaptacją języka sekwencyjnego (np.

Parallel Fortran) lub językiem zaprojektowanym wprost do tego celu (np. Occam).

*Kanał - channel.* - Połączenie, zwykle obsługiwane przez odrębny procesor, służące do komunikacji między procesorami.

*Kompilator wektorowy - vectorizing compiler.* - Program kompilujący programy napisane jako sekwencyjne w ten sposób, że maksymalnie możliwa liczba bloków jest realizowana wektorowo.

*Komunikujące się procesy sekwencyjne - Communicating Sequential Processes (CSP).* - Model obliczeń równoległych, w którym procesy zachowują zupełną autonomię a wspólne korzystanie z wyników, przy zachowaniu niezbędnych następstw czasowych, dokonuje się przez wymianę komunikatów.

*Komutacja pakietów - packet switching.* - Tryb komunikacji, w którym pakiety są przekazywane od nadawcy do odbiorcy nie według z góry wybranej drogi, lecz zgodnie z bieżącą dostępnością łączy (tzw. pocztowy tryb komunikacji).

*Komutacja połączeń - circuit switching.* - Tryb komunikacji, w którym pakiety są przesyłane od nadawcy do odbiorcy wzdłuż drogi ustalonej przed rozpoczęciem transmisji i nie zmieniającej się w trakcie jej trwania (tzw. telefoniczny tryb komunikacji).

*Komutator krzyżowy - crossbar switch.* - Typ komutatora sterowanego utworzony z przełączników umieszczonych w węzłach dwuwymiarowej, prostokątnej siatki połączeń umożliwiający w maszynie wieloprocessorowej komunikację każdego procesora z każdym.

*Komutator motylowy - butterfly switch.* - Typ komutatora używanego w sieciach motylowych.

*Krata - mesh.*

- 1. Sieć  $n$ -wymiarowa,  $n \geq 2$ , której węzły tworzą siatkę. Węzły są ponumerowane liczbami naturalnymi od 0 do  $2^{**n}$ , przy czym każdy węzeł wewnętrzny siatki jest połączony z  $2n$  sąsiadami, których numery różnią się jednym bitem liczby binarnej przedstawiającej numer danego węzła. Każdy węzeł zewnętrzny może mieć albo tylko  $n$  sąsiadów (*krata bez domknięcia*), albo też  $2n$  węzłów sąsiadujących (*krata z domknięciem*).
- 2. Architektura maszyny równoległej, której procesory i połączenia między nimi odwzorowują kratę.

*Makropotokowanie - macropipelining.* - Potokowanie obliczeń na poziomie algorytmu. (Patrz: *algorytm potokowy*).

*Maska - mask.* - Wektor binarny używany w maszynach *SIMD* do wyłączenia z działania niektórych procesorów dla pewnych instrukcji.

*Masowa równoległość - massive parallelizm.* - Pojęcie używane w odniesieniu do zadań (algorytmów) i maszyn równoległych, dla których *ziarnistość* - odpowiednio liczba procesorów jest rzędu  $10^{**4} + 10^{**6}$ .

*Maszyna Boltzmanna - Boltzmann machine.* - Maszyna zbudowana z bardzo dużej liczby dwustanowych elementów operacyjnych, których stan oraz sposób połączenia są funkcjami losowymi. Z każdym stanem maszyny jest związana (jednoznacznie) pewna wartość funkcji nazywanej *funkcją zgodności*.

*Maszyna MIMD - MIMD (Multiple Instruction stream Multiple Data stream) machine.* - Maszyna równoległa, w której strumień wektorów rozkazów oddziałuje (synchronicznie) na strumień wektorów danych. Możliwy jest również niesynchroniczny wariant maszyny *MIMD*. (Patrz hasła: *wielokomputer, wieloprocesor*).

*Maszyna MISD - MISD (Multiple Instruction stream Single Data stream) machine.* - Maszyna równoległa, w której strumień wektorów rozkazów oddziałuje na strumień złożony z, być może różnych, wektorów danych o identycznych składowych. Maszyna jest czasami utożsamiana z *maszyną potokową*.

*Maszyna neuronowa - neural computer.* - Rodzaj maszyny równoległej zbudowanej według koncepcji funkcjonowania mózgu ludzkiego z wykorzystaniem pojęcia i modelu neuronu.

*Maszyna o bardzo dużym stopniu równoległości - massively parallel machine.* - Maszyna równoległa o liczbie procesorów rzędu  $10^{**4} + 10^{**6}$ .

*Maszyna o ograniczonym zbiorze instrukcji (RISC) - reduced instruction set computer (RISC).* - Maszyna o ograniczonym zbiorze instrukcji, z których każda jest wykonywana w mniejszej liczbie cykli CPU.

*Maszyna równoległa - parallel computer.* - Maszyna cyfrowa przygotowana pod względem architektury i oprogramowania do wykonywania obliczeń równoległych.

*Maszyna systoliczna - systolic computer.* - Specjalistyczna maszyna SIMD w technologii BDSS, złożona z prostych elementów operacyjnych tworzących regularną sieć (Krata dwuwymiarowa, Hipersześcian, Idealnie potasowany stos, Drzewo itp.). Cechuje ją rytmiczność operacji obliczeniowych i komunikacyjnych, wykonywanych na wektorach danych uformowanych w potoki.

*Maszyna wektorowa - vector computer.* - *Maszyna równoległa*, której architektura umożliwiła synchroniczne wykonywanie wektora instrukcji na wektorze danych. W zawężonym pojęciu maszyna wektorowa jest utożsamiana z superkomputerem.

*Maszyny SIMD - SIMD (Single Instruction stream Multiple Data stream) machines.* - Maszyny, w których strumień wektorów rozkazów o identycznych składowych oddziałuje, synchronicznie na strumień wektorów danych:

- *SIMD-CC (SIMD-CC = SIMD-cube connected machine).* - Maszyna SIMD o architekturze hipersześcianu.
- *SIMD-CCC (SIMD-CCC = SIMD-cube connected cycles machine).* - Maszyna SIMD o architekturze hipersześcianu cykli.
- *SIMMD-CRCW (SIMD-CRCW = SIMD-concurrent read concurrent write machine).* - Maszyna SIMD dopuszczająca jednoczesne czytanie z i pisanie do tego samego miejsca pamięci.
- *SIMD-CREW (SIMD-CREW = SIMD-concurrent read exclusive write machine).* - Maszyna SIMD dopuszczająca jednoczesne czytanie z tego samego miejsca pamięci, ale wykluczająca jednoczesny zapis.
- *SIMD-ERCW (SIMD-ERCW = SIMD-exclusive read concurrent write machine).* - Maszyna SIMD, wykluczająca jednoczesne czytanie z tego samego miejsca pamięci, ale dopuszczająca jednoczesny zapis.
- *SIMD-EREW (SIMD-EREW = SIMD-exclusive read exclusive write machine).* - Maszyna SIMD wykluczająca jednoczesne czytanie/ pisanie z/do tego samego miejsca pamięci.
- *SIMD-MC (SIMD-MC = SIMD-mesh connected machine).* - Maszyna SIMD, której procesory tworzą kratę.

- *SIMD-P* (*SIMD-P = SIMD-Pyramid machine*). - Maszyna *SIMD* o architekturze piramidy.
- *SIMD-PS* (*SIMD-PS = SIMD-perfect shuffle machine*). - Maszyna *SIMD* o architekturze *IPS*.
- *SIMD-SM* (*SIMD-SM = SIMD - shared memory machine*). - Maszyna *SIMD* ze wspólną pamięcią.
- *SIMD-SM-R* (*SIMD-SM-R*). - Maszyna *SIMD-CREW*.
- *SIMD-SM-RW* (*SIMD-SM-RW*). - Maszyna *SIMD-CRCW*.

*MIMD - MIMD = Multiple Instruction stream, Multiple Data stream.*  
- Organizacja obliczeń równoległych polegająca na wykonywaniu zestawu instrukcji na zestawie danych.

*Model BDSSS - VLSI model.* - Odmiana modelu *MP-RAM* dostosowana do modelowania obliczeń na maszynach równoległych będących modułami *BDSS*. Różne warianty modelu *BDSS* różnią się między sobą założeniami o liczbie warstw modułu, sposobem uwzględnienia opóźnień komunikacyjnych na połączeniach i założeniami o powierzchni zajmowanej przez elementy półprzewodnikowe modułu. Złożoność obliczeniowa algorytmów w tych modelach jest wyrażana iloczynem powierzchni modułu i kwadratu czasu.

*Model MP-RAM - MP-RAM (message passing random access) model.*  
Odmiana modelu *P-RAM*. W tym modelu nie ma pamięci wspólnej, każdy procesor ma własną pamięć. Komunikacja między procesorami odbywa się przez wymianę komunikatów.

*Model P-RAM = maszyna równoległa o swobodnym dostępie - P-RAM (parallel random access) model.* Uogólnienie sekwencyjnego modelu *RAM* na model obliczeń równoległych. Nieograniczona liczba procesorów,  $p > 1$ , ma bezkonfliktowy dostęp do nieograniczonej pamięci. Czas wykonywania operacji podstawowych i czas dostępu do pamięci nie są zależne ani od  $p$ , ani od rodzaju operacji. Uruchomienie  $p$  procesorów wymaga  $\log(p)$  jednostek czasu.

*Obliczenia (przetwarzanie) równoległe - parallel computing (processing).* - Organizacja obliczeń przy użyciu wielu procesorów mająca na celu szybszego rozwiązywania zadań o dużych rozmiarach.

*Obliczenia (programowanie, przetwarzanie) współbieżne - concurrent computing (programming, processing).* - Patrz: *współbieżność*.

*Occam - Occam.* - Język programowania transputerów.

*Optymalizacja równoległa - parallel optimization.* - Zastosowanie maszyn i obliczeń równoległych do rozwiązywania zadań optymalizacji.

*Pakiet - packet.* - Zbiór danych o ustalonym formacie tworzących jednostkę przekazywaną według ustalonej drogi od źródła do odbiorcy.

*Pamięć asocjacyjna - associative memory.* - Pamięć, w której dane są adresowane według zawartości, a nie według zajmowanego miejsca. Wyszukiwanie potrzebnej informacji odbywa się przez sprawdzanie zgodności określonych warunków logicznych.

*Parakomputer - paracomputer.* - Patrz: *Model P-RAM*.

*Pasma transmisji - communication width.*

- 1. Szerokość pasma (transmisji).
- 2. Rozmiar pamięci wspólnej w maszynach SIMD.

*Piramida - pyramid.*

- 1. Sieć w kształcie pełnego drzewa kwadratowego o wysokości  $\log_4 p$ , gdzie  $p$  jest liczbą procesorów. Węzły znajdujące się na tej samej głębokości połączone są w kratę. Każdy węzeł wewnętrzny piramidy jest połączony z dziewięcioma sąsiadami.
- 2. Architektura  $p$ -procesorowej,  $p > 5$ , maszyny równoległej, w której procesory i połączenia między nimi odpowiadają węzłom i ich połączeniom w piramidzie.

*Podproces - task.* - Część składowa procesu równoległego. Pojęcie to jest definiowane precyzyjnie w językach programowania równoległego (Parallel Fortran, Modula, Ada).

*Podwójne drzewo binarne - double binary tree.* - Sieć i architektura maszyny równoległej utworzone z dwóch identycznych drzew binarnych o wspólnych liściach (drzewo to ma dwa korzenie).

*Potokowanie (konwejerizacja) obliczeń - pipelining.* - Sposób zwiększania współbieżności obliczeń polegający na dzieleniu procesów na uporządkowany ciąg segmentów, które są wykonywane przez różne stopnie procesora potokowego jednocześnie.



*Potokowanie instrukcji - instruction pipelining.* - Organizacja obliczeń umożliwiająca jednoczesne wykonywanie wielu segmentów pojedynczej instrukcji na procesorze potokowym.

*Potokowanie programu - software pipelining.* - Potokowanie obliczeń na poziomie programu.

*Proces - process.* - Autonomiczny ciąg działań maszyny cyfrowej.

*Proces obliczeniowy - computing process.* - Ciąg zdarzeń w maszynie cyfrowej mających na celu wyznaczenie pewnej wartości.

*Procesor - processor.* - W obliczeniach równoległych pojęcie to obejmuje jednostki obliczeniowe o bardzo różnej mocy, od prostego elementu operacyjnego w maszynach systolicznych, po rozbudowane jednostki centralne (CPU) w wielokomputerach.

*Procesor asocjacyjny - associative processor.* - Procesor z pamięcią asocjacyjną.

*Procesor dołączony - attached processor.* - Procesor (zwykle tablicowy) dołączony do procesora sekwencyjnego dla zwiększenia szybkości obliczeń.

*Procesor potokowy (kaskadowy) - pipelined processor.* - Procesor utworzony z szeregowo połączonych segmentów przeznaczony do potokowania obliczeń. Liczba segmentów decyduje o przyspieszeniu obliczeń w porównaniu z procesorem jednostopniowym, i o opóźnieniu wyników na wyjściu w stosunku chwili pojawienia się danych na wejściu.

*Procesor tablicowy - array processor.* - Procesor złożony z  $k \times 2$  elementów operacyjnych tworzących tablicę dwuwymiarową i połączonych między sobą według zasady sąsiedztwa. Służy jako przystawka synchroniczna do maszyny sekwencyjnej, przyspieszająca obliczenia w przypadku macierzowej reprezentacji danych.

*Program testów porównawczych - benchmark program* - Standardowy program, w języku wyższego poziomu, używany do porównań prędkości i efektywności maszyn, języków, kompilatorów.

*Przesyłanie danych - data transmission.* - Metody i techniki wymiany informacji między procesorami oraz ich komunikacji z otoczeniem zewnętrznym.

- Przetwarzanie pionowe - vertical processing.* - Przetwarzanie tablicy dwuwymiarowej w kolejności kolumn.
- Przetwarzanie wierszowe - horizontal processing.* - Przetwarzanie tablicy dwuwymiarowej w kolejności wierszy.
- Przyśpieszenie - speedup.* - Stosunek czasu rozwiązania danego zadania przez najlepszy algorytm sekwencyjny do czasu rozwiązania tego samego zadania przez rozpatrywany *algorytm równoległy*. (Jest to jedna z możliwych definicji, w literaturze są spotykane także inne definicje).
- Przyśpieszenie superliniowe - superlinear speedup.* - *Przyśpieszenie* o wartości większej niż liczba procesorów.
- Relacja nadrzędność/podrzędność - master/slave relationship.* - Forma sterowania procesem obliczeń polegająca na tym, że jeden procesor steruje bezpośrednio jednym lub kilkoma procesorami podporządkowanymi.
- Równoległość - parallelizm.* - Cecha zadań numerycznych i zadań przetwarzania symboli pozwalająca wydzielić procesy, które mogą być realizowane *współbieżnie*.
- Równoległość naturalna - explicit parallelizm.* - Cecha zadań obliczeniowych umożliwiająca ich dekompozycję na części, które mogą być realizowane jako *procesy współbieżne*.
- Segment - segment.* - Część składowa procesu powstała z jego podziału w trakcie *potokowania obliczeń*. *Semafor - semaphore.* - Zmienna globalna typu INTEGER stosowana do synchronizacji procesów.
- Shuffle - shuffle.* - Operacja przekazania jednostki informacji, w maszynie równoległej o architekturze typu *Idealnie Potasowany Stos*, od procesora o numerze  $i$  do procesora, którego numer otrzymuje się z liczby  $i$  przez cykliczne przesunięcie, w jej rozwinięciu binarnym, każdego bitu o jedno miejsce w lewo.
- Sieć motylowa - butterfly network.* - Sposób połączenia procesorów polegający na umieszczeniu  $(k + 1) 2^{**k}$  procesorów w  $2^{**k}$  kolumnach i  $k + 1$  wierszach. Niech para indeksów  $(i, j)$  określa procesor  $P$  na przecięciu  $i$ -go wiersza i  $j$ -ej kolumny ( $0 \leq i \leq k$ ,  $0 \leq j \leq 2^{**k}$ ), wtedy procesor  $P(i, j)$ ,  $i > 0$ , jest połączony z dwoma procesorami w wierszu  $(i - 1)$ :  $P(i - 1, j)$  i  $P(i - 1, m)$ , gdzie



$m$  jest liczbą całkowitą otrzymaną z zanegowania  $i$ -go najwyższego bitu w binarnym rozwinięciu liczby  $j$ .

*Sieć wielostopniowa - multistage network.* - Sieć komunikacji między procesorami utworzona z połączonych kaskadowo przełączników.

*SIMD - SIMD (single instruction multiple data).* - Organizacja obliczeń równoległych, polegająca na jednoczesnym działaniu wektora identycznych instrukcji na wektor danych.

*Sterowanie strumieniem danych - data driven* (także: *data flow, control* - Organizacja obliczeń maszyny równoległej polegająca na wykonywaniu instrukcji w porządku dostępności danych.

*Sterowanie strumieniem potrzeb - demand driven control.* - Organizacja obliczeń maszyny równoległej oparta na wykonywaniu instrukcji w porządku zapotrzebowania na określone wyniki.

*Sterowanie strumieniem rozkazów - control driven.* - Organizacja obliczeń maszyny cyfrowej polegająca na wykonywaniu instrukcji zgodnie z zawartością rejestru rozkazów.

*Stopień - stage.* - Część składowa procesora potokowego, potokowej jednostki funkcyjnej. Połączone szeregowo stopnie tworzą kaskadę.

*Superkomputer - supercomputer.* - Maszyna cyfrowa utworzona z różnych, wąsko wyspecjalizowanych, jednostek funkcjonalnych o wektorowej i potokowej organizacji obliczeń, przede wszystkim numerycznych. Inna definicja: *superkomputer* jest to maszyna cyfrowa zdolna do rozwiązywania zadań obliczeniowych z prędkością przewyższającą wszystkie dostępne, w danym czasie, maszyny.

*Synchroniczna wymiana komunikatów - synchronous message passing.* - Komunikacja polegająca na wymianie komunikatów realizowana bez buforowania, ale z blokowaniem nadajnika i odbiornika.

*Szerokość algorytmu równoległego - parallel algorithm width.* - Minimalna liczba procesorów koniecznych do wykonania algorytmu.

*Szerokość pasma - bandwidth.* - W odniesieniu do kanału transmisji danych jest to maksymalna liczba jednostek informacji (bitów, bajtów) możliwych do przesłania w jednej sekundzie.

*Tablica procesorów - processor array. - Maszyna wektorowa SIMD.*

Pojęcie utożsamiane często z procesorem tablicowym lub maszyną systoliczną.

*Tablica systoliczna (procesor systoliczny) - systolic array (systolic processor).* - Zespół elementów operacyjnych o różnorodnej topologii połączeń, pracujących synchronicznie i rytmicznie łącząc takt wykonywania operacji z taktem przekazywania danych.

*Transmisja - transmission. - Przesyłanie danych.*

*Transmisja asynchroniczna - asynchronous message passing.* - Tryb przesyłania danych, dla którego moment odbioru danych nie jest z góry określony, niemniej pakiety danych są przekazywane w regularnych przedziałach czasu.

*Transmisja buforowana - buffered message passing.* - Organizacja przesyłania pakietów, która umożliwia ich przechowywanie w buforach znajdujących się na drodze od nadawcy do odbiorcy.

*Transmisja synchroniczna - synchronous transmission.* - Rodzaj transmisji, w którym sygnały są przesyłane w regularnych odstępach czasu kontrolowanych sygnałem synchronizującym.

*Transmisja w sieci IPS - perfect shuffle routing.* - Komunikacja przy wykorzystaniu operacji *Shuffle*, *Unshuffle* i *Exchange*.

*Transputer - transputer.* - Procesor 32-bitowy RISC, przygotowany układowo i programowo do tworzenia sieci wieloprocesorowych i obliczeń równoległych. Sieci transputerowe pracują w trybie komunikujących się procesów sekwencyjnych przez synchroniczne i niebuforowane kanały transmisji. Językiem programowania równoległego integralnie związanym z transputerem jest *Occam*.

*Ultrakomputer - ultracomputer.* - Patrz *Model MP-RAM*.

*Unshuffle - unshuffle.* - Operacja przekazania jednostki informacji, w maszynie równoległej o architekturze typu *Idealnie Potasowany Stos*, od procesora o numerze *i* do procesora, którego numer otrzymuje się z liczby *i* przez cykliczne przesunięcie, w jej rozwinięciu binarnym, każdego bitu o jedno miejsce w prawo. *Wątek - thread.* - Część składowa procesu równoległego. Pojęcie to jest definiowane ściśle w konkretnych językach programowania. Dla przykładu, w *Parallel Fortran* (firmy *3L*) wątki są generowane

przez procesy. Wątki danego procesu mają z nim wspólny program oraz dane i komunikują się w jego obrębie.

*Wektor maskujący - masking vector.* - Patrz maska.

*Whetstone - Whetstone.* - Program testujący szybkość obliczeń w odniesieniu do działań zmiennie-przecinkowych.

*Wielokomputer - multicomputer.* - System złożony z wielu CPU przeznaczony do obliczeń równoległych, nie dysponujący wspólną pamięcią.

*Wieloprocessor - multiprocessor.* - wielokomputer + wspólna pamięć.

*Współbieżność (współbieżne: przetwarzanie, obliczenia, operacje, programowanie) - Concurrency (Concurrent: processing, computing, execution, programming).* - Organizacja obliczeń umożliwiająca jednoczesną realizację wielu procesów, w celu uzyskania większej przepustowości komputera.

*Wydajność - throughput.* - Możliwości obliczeniowe maszyny, wyrażone przeciętną liczbą operacji na jednostkę czasu, odniesioną w stosunku do wszystkich zadań, a nie do jednego konkretnego zadania.

*Zadania/algorytmy o bardzo dużym stopniu równoległości - massively parallel problems/algorithms.* Zadania/algorytmy umożliwiające odwzorowanie na procesy równoległe, których liczba może być rzędu  $10^{**4} \div 10^{**6}$ .

*Ziarnistość - granularity (grain size).*

- 1. Dla zadań: liczba współbieżnych procesów, na które daje się podzielić proces przedstawiający zadanie.
- 2. Dla algorytmu równoległego : średnia liczba operacji, które można wykonać bez potrzeby komunikacji między procesami.

## Dodatek 2

### RAPORTY I PUBLIKACJE ZAKŁADU PROGRAMOWANIA MATEMATYCZNEGO IBS PAN DOTYCZĄCE RÓWNOLEGŁEJ OPTYMALIZACJI DYSKRETNEJ (1986-92)

#### I. Publikacje

1. Bertocchi M., Brandolini L., Słomiński L., Sobczyńska J. (1992): "A Monte-Carlo Approach for 0-1 Programming Problems". *Computing* 48, 259-274.
2. Dudziński K. (1986): "Wybrane równoległe algorytmy kombinatoryczne". *Roczniki PTM Seria III, Matematyka Stosowana XXVIII*, 91-123.
3. Kaliszewski I., Wojtowicz M. (1991) "Modest. A language for parallel programming". *Prace IPI PAN*, 691, Warszawa.
4. Słomiński L., Kaliszewski I. (1988): "Problemy obliczeń równoległych". *Prace IBS PAN*, 168, Warszawa.
5. Słomiński L. (1988): "Algorytmy równoległe znajdowania minimumów rozgałęzień". *Zeszyty Naukowe Politechniki Śląskiej, ser. Automatyka* 94, 287-301.
6. Słomiński L. (1988): "Obliczenia równoległe i optymalizacja". W: *Materiały Konferencji Naukowej: Problemy współczesnej radiolokacji*, WAT, Warszawa, 124-130.
7. Słomiński L. (1989): "Parallel Algorithms for Optimization". W: *Proceedings of the 3rd Polish-Finish Symp. on Methodology and Applications of Decision Support Systems*, red. R. Kulikowski, IBS PAN, Warszawa, 219-233.
8. Słomiński L. (1990): "Implementacje algorytmów kombinatorycznych na symulowanych sieciach wielomikroprocesorowych".

Zeszyty Naukowe Politechniki Śląskiej, ser. Automatyka, 100, 287-300.

9. Słomiński L. (1990): "Parallel Computing for Flexible Manufacturing Systems". W: Decision Making Models for Management and Manufacturing, red. R. Kulikowski, Omnitech Press, Warszawa, 215-229.
10. Słomiński L. (1990): "Komputery równoległe dla rozwiązywania zadań optymalizacji dyskretnej i sztucznej inteligencji". Materiały konferencji "Sztuczna inteligencja w zarządzaniu", Warszawa, listopad 1989, PTBios i IBS PAN, red. A. Straszak, Z. Nahorski, C. Iwański, Warszawa, 201-210.

## II. Raporty (latami, wg alfabetu)

### 1986

1. Dudziński K.: "Obliczenia równoległe: Wybrane algorytmy kombinatoryczne". ZPM 20/86, IBS PAN, Warszawa.
2. Grygiel G.: "Obliczenia równoległe: Algorytm równoległy dla zadań przydziału". ZPM 24/86, IBS PAN, Warszawa.
3. Kaliszewski I.: "Obliczenia równoległe: Modele obliczeń w języku Ameba-Pascal". ZPM 23/86, IBS PAN, Warszawa.
4. Kaliszewski I., Słomiński L.: "Obliczenia równoległe: przegląd zagadnień". ZPM 17/86, IBS PAN, Warszawa.
5. Majchrzak J.: "Obliczenia równoległe: uwarunkowania i prognozy". ZPM 21/86, IBS PAN, Warszawa.
6. Słomiński L.: "Obliczenia równoległe: algorytm równoległy wyznaczania dendrytu minimaxowego w grafie skierowanym". ZPM 19/86, IBS PAN, Warszawa.

7. Waluk B.: "Obliczenia równoległe: Zastosowanie niektórych metod dekompozycji macierzy współczynników do rozwiązywania układów równań". ZPM 22/86, IBS PAN, Warszawa.

**1987**

1. Grygiel G.: "Zadanie przydziału: algorytm równoległy i eksperymentalna ocena liczby iteracji". ZPM 34/87, IBS PAN, Warszawa.
2. Kaliszewski I.: "Parallel counterparts of algorithms for determining minimal elements of discrete sets". ZPM 30/87, IBS PAN, Warszawa.
3. Kaliszewski I., Wojtowicz M.: "Język Modest". ZPM 31/87, IBS PAN, Warszawa.
4. Kulczycka D.: "Równoległe algorytmy b-skojarzeń w drzewach". ZPM 23/87, IBS PAN, Warszawa.
5. Majchrzak J., Skawiński A.: "Sieci komputerowe - przegląd literatury i ocena możliwości przetwarzania równoległego". ZPM 17/87, IBS PAN, Warszawa.
6. Majchrzak J., Skawiński A.: "Pakiet PPLANEUP do organizacji przetwarzania równoległego w sieci D-Link komputerów IBM PC/XT/AT". ZPM 18/87, IBS PAN, Warszawa.
7. Majchrzak J., Skawiński A.: "Propozycja bezgradientowej metody równoległej optymalizacji nieliniowej bez ograniczeń". ZPM 19/87, IBS PAN, Warszawa.
8. Słomiński L.: "Narzędzia optymalizacji równoległej na maszynach sekwencyjnych". ZPM 13/87, IBS PAN, Warszawa.
9. Słomiński L.: "Problemy równoległych algorytmów dla przepływu maksymalnego w sieci". ZPM 14/87, IBS PAN, Warszawa.

**1988**

1. Gondek H., Słomiński L.: "Implementacja na symulowanej maszynie równoległej typu dwudrzewo algorytmu Dijkstry wyznaczania drzewa dróg najkrótszych w digrafie". ZPM 37/88, IBS PAN, Warszawa.
2. Grygiel G., Kulczycka D.: "Ocena systemu symulacyjnego JUPITER-86". ZPM 24/88, IBS PAN, Warszawa.
3. Grygiel G.: "Symulacja algorytmu równoległego dla zadania przydziału przy pomocy pakietu JUPITER-86". ZPM 44/88, IBS PAN, Warszawa.
4. Kaliszewski I., Kulczycka D., Słomiński L.: "Równoległy algorytm Prima-Dijkstry wyznaczania dendrytu minimalnego: implementacja na symulowanej strukturze typu dwudrzewo". ZPM 5/88, IBS PAN, Warszawa.
5. Kaliszewski I., Słomiński L.: "Problemy obliczeń równoległych". ZPM 38/88, IBS PAN, Warszawa.
6. Kulczycka D.: "Algorytmy równoległe b-skojarzeń w drzewach: implementacja w systemie symulacyjnym Jupiter". ZPM 6/88, IBS PAN, Warszawa.
7. Majchrzak J.: "Pakiet PPLANEUP wersja 2 dla wspomaganie przetwarzania równoległego w sieciach D-Link komputerów IBM PC/XT/AT". ZPM 41/88, IBS PAN, Warszawa.
8. Słomiński L.: "Algorytmy równoległe znajdowania minimaksowych rozgałęzień". ZPM 1/88, IBS PAN, Warszawa.
9. Słomiński L.: "Obliczenia równoległe i optymalizacja". ZPM 4/88, IBS PAN, Warszawa.

**1989**

1. Gondek H., Słomiński L.: "Symulowane odmiany struktury dwudrzewo dla algorytmów najkrótszych dróg w digrafie". ZPM 10/89, IBS PAN, Warszawa.

2. Gondek H.: "Równoległy algorytm mnożenia macierzy. Implementacja i eksperyment obliczeniowy na symulowanej strukturze typu mesh i hypercube o  $n^2$  procesorach". ZPM 11/89, IBS PAN, Warszawa.
3. Grygiel G.: "Równoległość na poziomie podprocedur na przykładzie zadania przydziału". ZPM 13/89, IBS PAN, Warszawa.
4. Grygiel G.: "Program AZP rozwiązujący algebraiczne zagadnienie przydziału - opis funkcjonalny". ZPM 18/89, IBS PAN, Warszawa.
5. Kaliszewski I.: "VM1 - A parallel algorithms to determine minimal elements of discrete sets". ZPM 16/89, IBS PAN, Warszawa.
6. Kulczycka D.: "Algorytmy równoległe w sieci drzew ortogonalnych - realizacja w systemie symulacyjnym JUPITER-86". ZPM 12/89, IBS PAN, Warszawa.
7. Majchrzak J.: "O programowaniu dla transputerów INMOS T800". ZPM 14/89, IBS PAN, Warszawa.
8. Słomiński L.: "Parallel Algorithms for Optimization". ZPM 1/89, IBS PAN, Warszawa.
9. Słomiński L.: "Komputery równoległe dla rozwiązywania zadań optymalizacji i sztucznej inteligencji". ZPM 3/89, IBS PAN, Warszawa.
10. Słomiński L., Sobczyńska J.: "Programy i dane w języku Fortran 77 dla wybranych zadań dyskretnych, z uwzględnieniem potrzeb wektoryzacji". ZPM 4/89, IBS PAN, Warszawa.
11. Sobczyńska J.: "Charakteryzacja wieloprocessorowej struktury Perfect Shuffle. Równoległy algorytm mnożenia macierzy i jego implementacja w tej strukturze". ZPM 9/89, IBS PAN, Warszawa.

## 1990

1. Grygiel G.: "Implementacja asynchronicznego algorytmu dla zadania przydziału". ZPM 26/90, IBS PAN, Warszawa.



2. Kaliszewski I.: "On determining minimal elements of discrete sets on a network of transputers". ZPM 4/90, IBS PAN, Warszawa.
3. Kulczycka D.: "Prosty algorytm równoległy znajdowania skojarzenia maksymalnego". ZPM 18/90, IBS PAN, Warszawa.
4. Majchrzak J.: "Transputer i sieci wielotransputerowe". ZPM 30/90, IBS PAN, Warszawa.
5. Słomiński L.: "Komunikacja w systemach wieloprocesorowych i jej wpływ na efektywność obliczeń". ZPM 22/90, IBS PAN, Warszawa.

### 1991

1. Bertocchi M., Butti A., Sergi P., Słomiński L., Sobczyńska J.: "Stochastic and Deterministic Optimization on 0-1 Multiknapsack Problem". *Quaderni del Dipartimento di Matematica, Statistica, Informatica e Applicazioni*, No 14/91, Bergamo.
2. Kulczycka D.: "Algorytm  $O(|E|)$  skojarzenia w grafie dwudzielnym". ZPM 13/91, IBS PAN, Warszawa.

### 1992

1. Bertocchi M., Butti A., Słomiński L.: "Fixed Precision Random Search Procedure for the Binary Multiknapsack Problem". (Revised version). *Quaderni del Dipartimento di Matematica, Statistica, Informatica e Applicazioni*, No 18/92, Bergamo.
2. Grygiel G., Kabanow P., Słomiński L., Sobczyńska J.: "Wykorzystanie sieci transputerowych do rozwiązania wielowymiarowego zadania załadunku metodą Monte-Carlo". /w języku rosyjskim/ ZPM 12/92, IBS PAN, Warszawa.

