
Paralelné programovanie

Bc. št. prog. Informatika - 2010/2011

Ing. Michal Čerňanský, PhD.

Fakulta informatiky a
informačných technológií,
STU Bratislava

Prehľad tém

- Paralelné a distribuované spracovanie
 - Témy predmetu
 - Programátorské modely
 - Organizácia predmetu
 - Podmienky abslovovania
-

Paralelné distribuované spracovanie

- NVidia Tesla D870
Supercomputer



Paralelné distribuované spracovanie

- Dôvody
 - Vysoký výkon (vedecké výpočty, grafika)
 - Spol'ahlivosť
 - Prirodzená distribuovanosť
 - Najvyšší cieľ – automatická paralelizácia (Holy Grail)
 - Zdroje paralelizmu – inštrukcie, dáta, úlohy
 - Výsledky
 - Moderné kompilátory
 - Moderné procesory
-

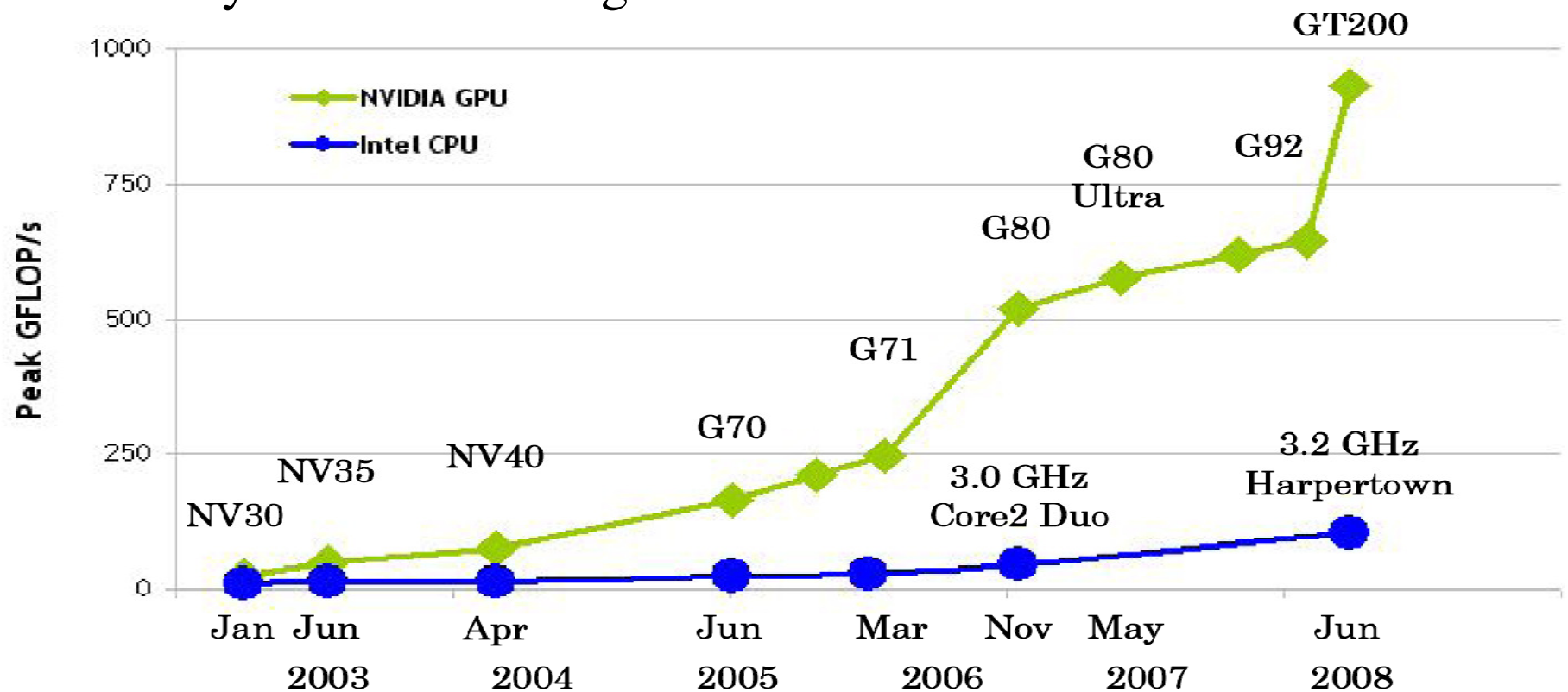
Paralelné distribuované spracovanie

■ Problémy

- Tepelná disipácia
- Rýchlosť šírenia signálu

■ Riešenie

- Viacej výpočtových jadier na procesor



Paralelné distribuované spracovanie

- Ale
 - Komplikovanejšia tvorba programov
 - Výskum
 - Nové jazyky alebo rozšírenia jazykov
 - Podporné prostriedky
 - Čiastočná automatická paralelizácia
-

Paralelné distribuované spracovanie

- Výskumné trendy
 - Vysokovýkonné počítanie (High Performance computing)
 - Vedecké počítanie (Scientific Computing)
 - Adaptácia algoritmov pre využitie masívneho paralelizmu
 - Špecializované programovacie jazyky a podporné nástroje
-

Paralelné distribuované spracovanie

- Súčasné HW prostriedky
 - Viacjadrové procesory
 - Mnohojadrové (grafické) procesory
 - Počítačové klastre
 - Počítačové gridy
 - Súčasné SW prostriedky
 - Message Passing Interface (MPI)
 - Open Multiprocessing (OpenMP)
 - Cuda a Open Computing Language (OpenCL)
-

Paralelné distribuované spracovanie

- Scientific Computing:

www.scientificcomputing.com

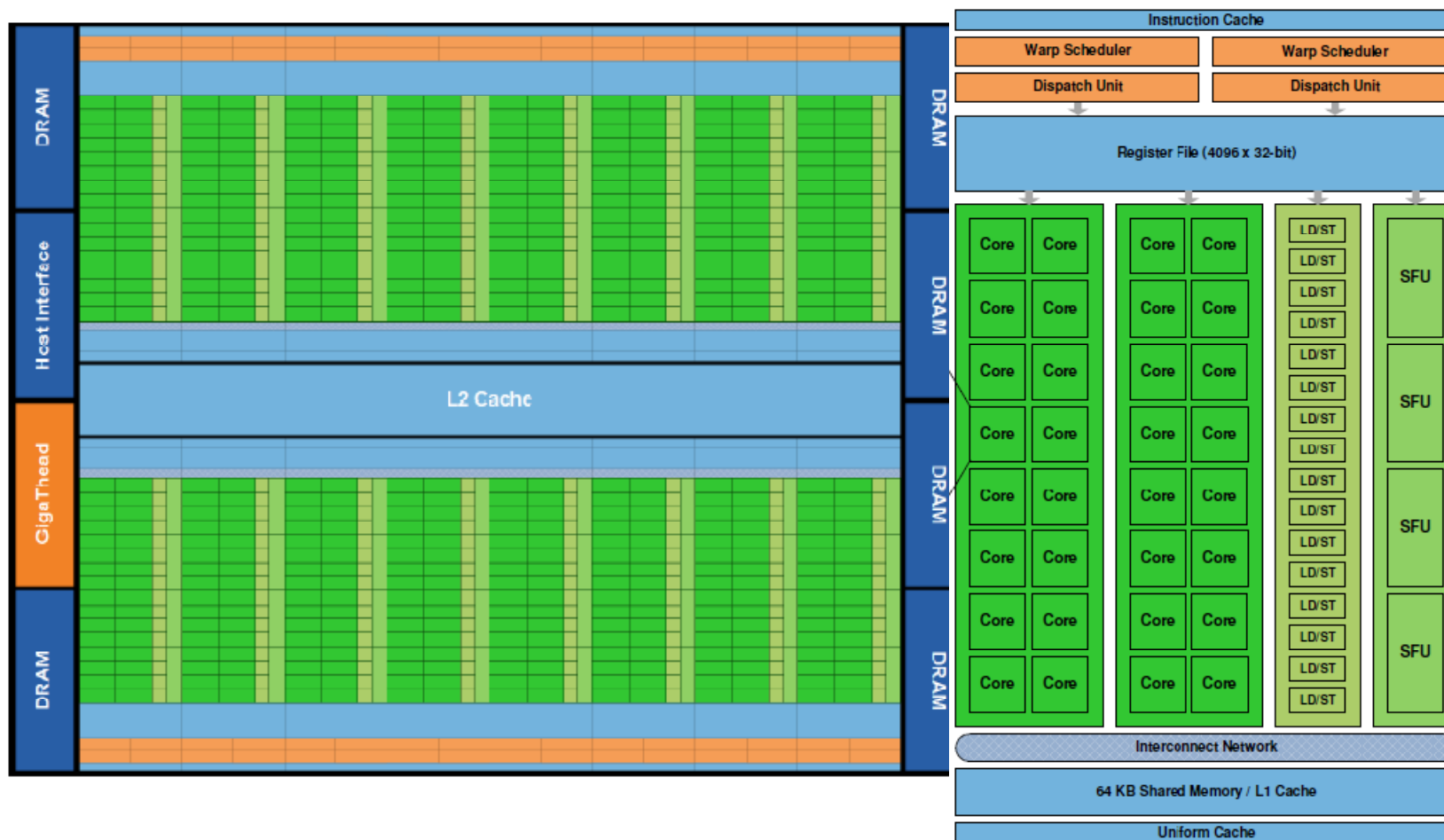
- GPGPU:

www.gpgpu.org

- Cuda Zone:

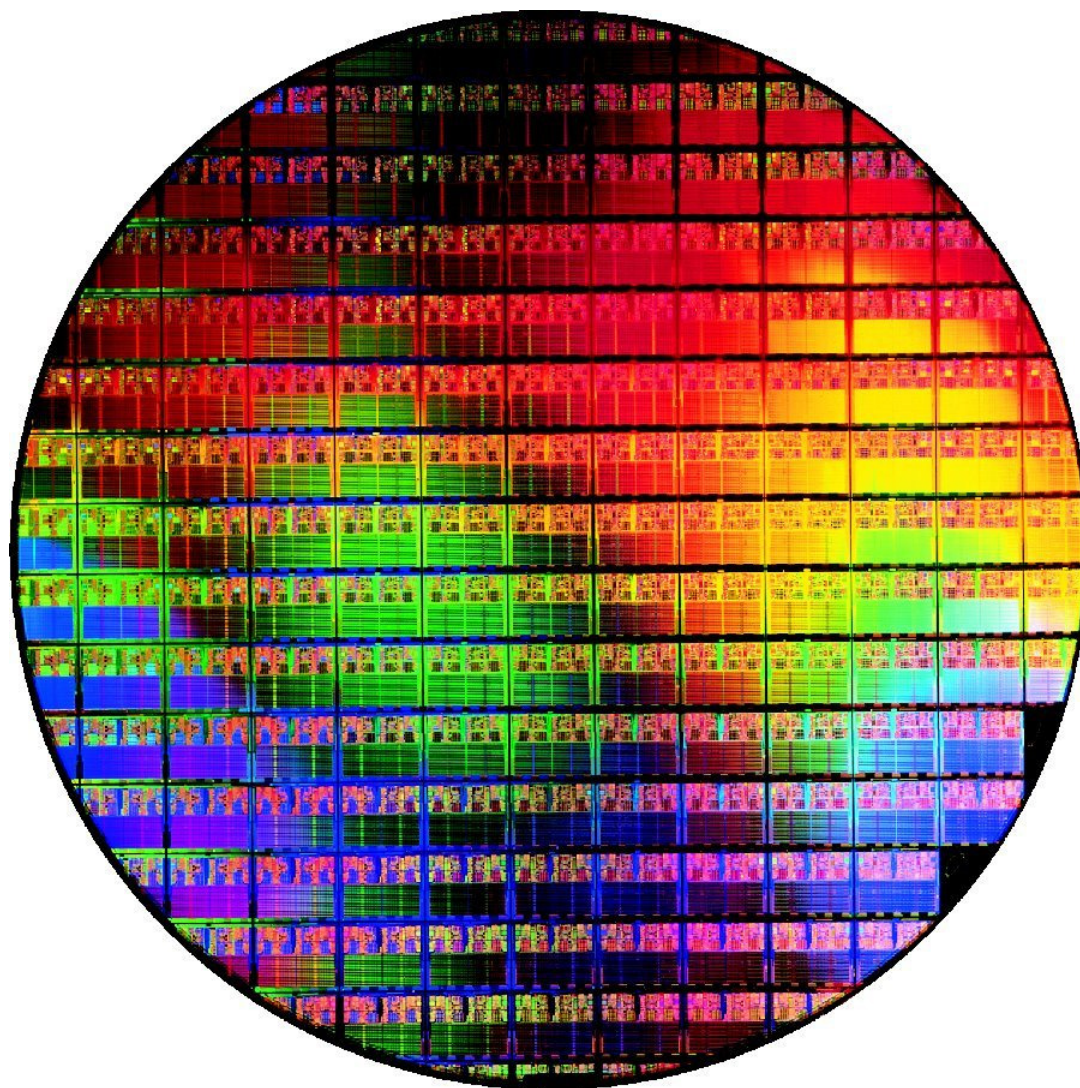
www.nvidia.com/object/cuda_home.html

Paralelné distribuované spracovanie



- NVidia Fermi Architecture (October 2009)

Paralelné distribuované spracovanie



Informačný list predmetu

Predpokladané znalosti, zručnosti a schopnosti:

- Znalosti súvisiace s návrhom a realizáciou sekvenčných algoritmov v programovacom jazyku C/C++.

Spôsob hodnotenia a ukončenia štúdia predmetu:

- Zápočet (60%) a skúška (40% z celkového hodnotenia)

Priebežné hodnotenie:

- Projekty menšieho rozsahu 4x,
- semestrálny projekt 1x,

Záverečné hodnotenie:

- Záverečný test
-

Informačný list predmetu

Cieľ predmetu:

- Poskytnúť znalosti o metódach a prostriedkoch paralelného spracovania s dôrazom na ich využitie pri tvorbe aplikácií.
 - Získať praktické skúsenosti s návrhom efektívnych paralelných algoritmov pomocou programovacích modelov pre symetrické multiprocesory a počítačové klastre,
 - Osobitná pozornosť bude venovaná programovacím modelom pre mnohojarové grafické procesory.
-

Informačný list predmetu

Stručná osnova predmetu:

- Flynnova taxonómia, Amhdalov zákon, Gustafsonov zákon
 - Systémy so zdieľanou a distribuovanou pamäťou, multiprocesory a multipočítače
 - Podmienky paralelizmu, dátová a zdrojová nezávislosť
 - Zdroje paralelizmu, paralelizmus na úrovni inštrukcií, dátový paralelizmus, paralelizmus úloh
 - Návrh paralelných programov, komunikácia, synchronizácia (atomické operácie, bariery, semaféry, mutexy), závislosť medzi dátami, dekompozícia, granularita, rozkladanie zát'aže
-

Informačný list predmetu

Stručná osnova predmetu (pokračovanie):

- Paralelné programovacie modely, model vlákien, model zasielania správ
 - Explicité použitie vlákien – Pthreads (resp. Java threads, Win32 threads, ...)
 - Implicitné použitie vlákien – OpenMP
 - Programovanie systémov s distribuovanou pamäťou – MPI
 - Programovanie mnohojadrových grafických procesorov – CUDA, OpenCL
 - Analytické modelovanie paralelných programov, analýza výkonnosti, ladenie
-

Informačný list predmetu

Stručná osnova predmetu (pokračovanie):

- Vzory pre paralelné programovanie
 - Iné paralelné programátorské modely napr.:
 - **Cilk++**,
 - **Map-reduce**,
 - **Unified Parallel C**,
 - **Thread Building Blocks**,
 - **Erlang**,
 - **BOINC**,
 - **CellBE**,
 - ...
-

Programovacie modely

Pthreads – POSIX Threads

- **Explicitné použitie vlákien**

```
static void *thread_func(void *vptr_args)
{
    ...
    return NULL;
}
```

```
pthread_t thread;
```

```
pthread_create(&thread, NULL, thread_func, NULL);
```

```
pthread_join(thread, NULL);
```

Programovacie modely

OpenMP – Open Multiprocessing

- **Implicitné použitie vlákien**

```
int main(int argc, char* argv[])
{
    #pragma omp parallel
    printf("Hello, world.\n");
    return 0;
}
```

```
#pragma omp parallel for
for (i = 0; i < N; i++)
{
    ...
}
```

Programovacie modely

MPI – Message Passing Interface

```
MPI_Init (&argc, &argv) ;
```

```
MPI_Comm_size (MPI_COMM_WORLD, &numprocs) ;
```

```
MPI_Comm_rank (MPI_COMM_WORLD, &myid) ;
```

```
MPI_Send (buff, BUFSIZE, MPI_CHAR, i, TAG,  
MPI_COMM_WORLD) ;
```

```
MPI_Recv (buff, BUFSIZE, MPI_CHAR, i, TAG,  
MPI_COMM_WORLD, &stat) ;
```

Programovacie modely

Cuda – Compute Unified Device Architecture,

```
dim3 blockDim(16, 16, 1);
dim3 gridDim(width / blockDim.x, height / blockDim.y, 1);

kernel<<< gridDim, blockDim, 0 >>>(d_odata, width, height);

__global__ void kernel(float* odata, int height, int width)
{
    unsigned int x = blockDim.x*blockIdx.x + threadIdx.x;
    unsigned int y = blockDim.y*blockIdx.y + threadIdx.y;
    float c = tex2D(tex, x, y);
    odata[y*width+x] = c;
}
```

Organizácia predmetu

- Prednášky – Michal Čerňanský
 - Pondelok 13:00 až 16:00
 - Účast' povinná (v rámci možností)
 - Cvičenia – Michal Čerňanský, Peter Drahoš
 - Pondelok 19:00 až 21:00 v TPC
 - Utorok 17:00 až 21:00 v TPC
 - Účast' nepovinná (podľa potreby)
 - Kozultačný charakter
 - Nutnosť odprezentovať a obhájiť úlohy a projekt
-

Organizácia predmetu

- Vypracovanie a odprezentovanie 4 úloh každá za 10 bodov
 - Pthreads - 4 týžden na cvičeniach
 - bude zadaná a vypracovaná na cvičení, max. čas na vypracovanie 90 min
 - OpenMP - 7 týždeň
 - MPI - 7 týždeň
 - CUDA – 10 týždeň
 - ulohy budú zadané v predstihu, bude možné ich vypracovať mimo cvičení a na cvičeniach ich prísť odprezentovať
 - - 2 body za každý týždeň oneskoreného odprezentovania
 - je možné prezentovať iba počas cvičení
-

Organizácia predmetu

- Vypracovanie a odprezentovanie projektu za 20 bodov
 - Tému dohodnúť a odovzdať zadanie najneskôr v 7 týždni
 - Vypracovať a odprezentovať v 12 týždni
 - - 4 body za každý týždeň oneskoreného odprezentovania
 - je možné prezentovať iba počas cvičení
 - Prezentuje sa produkt a odovzdáva sa krátka správa spolu s produktom
 - Možné témy:
 - Použitie metód a prostriedkov paralelného spracovania vo vašej bakalárskej práci alebo semestrálnom projekte – oznámiť aj vedúcemu práce
 - Spracovanie a odprezentovanie zvolenej problematiky
 - Témy z oblasti strojového učenia
 - Témy z oblasti počítačovej grafiky
 - ...
-

Organizácia predmetu

- Podmienky absolvovania predmetu
 - Vypracovanie, odprezentovanie a odovzdanie všetkých úloh a projektu
 - Tému projektu dohodnúť s cvičiacim a v stanovenom čase odovzdať zadanie projektu
 - Úlohy aj projekt musia byť funkčné, v zhode so zadaním - posúdi cvičiaci
 - Pri prezentácií je potrebné preukázať patričné vedomosti - posúdi cvičiaci
 - Najneskoršie odovzdanie na cvičeniach do konca semestra (- body)
 - Získanie aspoň 40 bodov za prácu počas semestra
 - Získanie aspoň 4 bodov za každú úlohu
 - Získanie aspoň 8 bodov za projekt
 - Získanie aspoň 56 bodov z celkového hodnotenie
-

Organizácia predmetu

- Prednášajúci aj cvičiaci si vyhradzujú právo udeliť bonusové body v prípade práce, ktorej kvalita alebo rozsah výrazne prevyšujú požadované parametre
 - Plagiarizmus a iné formy akademickej nečestnosti budú posúdené v zmysle štúdiijného poriadku fakulty
 - Hodnotenie predmetu FX
 - Informácia pre disciplinárnu komisiu
-