POSUDEK OPONENTA ZÁVĚREČNÉ PRÁCE

I. IDENTIFIKAČNÍ ÚDAJE

Název práce:	Paralelní algoritmy pro operace s hustými maticemi
Jméno autora:	Alexandr Krastenov
Typ práce:	bakalářská práce
Fakulta:	Fakulta jaderná a fyzikálně inženýrská (FJFI)
Katedra:	Katedra matematiky
Oponent práce:	Ing. Jakub Klinkovský, Ph.D.
Pracoviště oponenta práce:	KSI FJFI ČVUT v Praze

II. HODNOCENÍ JEDNOTLIVÝCH KRITÉRIÍ

Zadání **průměrně náročné**

Hodnocení náročnosti zadání závěrečné práce.

Zadání práce je poměrně jednoduché z pohledu použitých algoritmů, ale náročné z pohledu optimalizací potřebných pro dosažení efektivní a prakticky použitelné implementace.

Splnění zadání **splněno**

Posuďte, zda předložená závěrečná práce splňuje zadání. V komentáři případně uveďte body zadání, které nebyly zcela splněny, nebo zda je práce oproti zadání rozšířena. Nebylo-li zadání zcela splněno, pokuste se posoudit závažnost, dopady a případně i příčiny jednotlivých nedostatků.

Zadání považuji za splněné, všechny požadavky jsou v práci dostatečně adresovány.

Zvolený postup řešení **vhodný s výhradami**

Posuďte, zda student zvolil správný postup nebo metody řešení.

Postup práce spočíval nejprve ve studiu počáteční implementace algoritmů, poté ve zkoumání vlastností pomocí profileru, následně v aplikování jednotlivých optimalizací, a nakonec ve spuštění benchmarku a vyhodnocení výkonu pro testovací množinu matic. K postupu mám jednu hlavní výhradu: v implementaci jsou unit testy oddělené od benchmarku a testují tedy jen jeden, finální kernel. Formálně tedy není zajištěno, že ve zkoumaných kernelech nejsou chyby, i když se jejich kód při vývoji výrazně měnil.

Odborná úroveň **průměrná**

Posuďte úroveň odbornosti závěrečné práce, využití znalostí získaných studiem a z odborné literatury, využití podkladů a dat získaných z praxe.

Práce podrobně popisuje všechny teoretické poznatky a optimalizační techniky, které jsou použité v implementační části. Často je ale poměrně zmatek v tom, co je důsledek čeho a jaký to má vliv v širším kontextu. Na některých místech je vynechán popis zásadních věcí, bez kterých se z textu nedá pochopit, jestli optimalizace funguje, nebo ne. Uvedu několik konkrétních problémů z kapitoly 2:

- sekce 2.1 - ve druhém kódu nemá být blockDim.x ani blockDim.y, ale vhodná konstanta, např. 32 (v původním textu [13] to není kvalitně vysvětleno, má to být jednorozměrný blok vláken, ale dvourozměrný grid bloků)
- sekce 2.2 - použití sdílené paměti pro sčítání vektorů neušetří žádné paměťové operace, protože ke každému prvku se přistupuje právě jednou - naopak přibudou operace navíc (sdílená paměť, synchronizace) - má to smysl až pro násobení matic
- kód v sekci 2.5 nedává smysl, správně je kód v sekci 2.2 kde se za BLOCK_SIZE zvolí velikost warpu

- optimalizace "warptiling" je oproti použitému zdroji [13] popsána velmi zjednodušeně, chybí hierarchický pohled na rozdělení práce mezi vlákna (popsaná optimalizace "register blocking" je také totéž, co "threadtiling" v [13])

Popis implementace v kapitole 3 má tyto nedostatky:

- původní kernel 1.1 v sekci 3.2.1 není popsaný kompletně, např. výpočty některých indexů jsou schované za "..." a hlavně zde chybí samotný výpočet součinu - zápis hodnot do "tileC"
- popis kernelu 1.2 představuje jen výpočet indexů "row" a "col", ale ne jejich použití
- kernel 1.3 popisuje jen použití sdílené paměti, která už je ale v kernelu 1.1, a není jasné použití indexů "row" a "col" - v čem se liší od předchozích dvou kernelů?
- Při popisu kernelů je vynechaná část o spouštění (výběr rozdělení vláken do bloků v gridu apod.) což je nedílná součást optimalizace CUDA kernelů. V této části je navíc chyba, která je zakořeněná velmi hluboko (už v původní implementaci, ze které student vycházel): zbytečná dynamická alokace sdílené paměti - všechny kernely používají jen statickou paměť.

Formální a jazyková úroveň **průměrná**

Posuďte správnost používání formálních zápisů obsažených v práci. Posuďte typografickou a jazykovou stránku.

Práce je psána spisovnou angličtinou a navzdory svému rozsahu obsahuje jen minimum gramatických chyb. Z typografického pohledu mohlo být konzistentnější používání fontů (\texttt a \textit) v teoretické části, některé pojmy resp. objekty nejsou zvýrazněné vůbec. Definice normy vektoru v rovnicích (1.3) a (1.4) mohly být matematicky korektnější (chybí argument funkce na levé straně rovnice).

Výběr zdrojů, korektnost citací **výborné**

Vyjádřete se k aktivitě studenta při získávání a využívání studijních materiálů k řešení závěrečné práce. Charakterizujte výběr pramenů. Posuďte, zda student využil všechny relevantní zdroje. Ověřte, zda jsou všechny převzaté prvky řádně odlišeny od vlastních výsledků a úvah, zda nedošlo k porušení citační etiky a zda jsou bibliografické citace úplné a v souladu s citačními zvyklostmi a normami.

Některé zdroje uvedené v bibliografii nejsou citovány v textu, např. [1] a [2]. I tak je ale bibliografie dostatečně obsáhlá na bakalářskou práci a citace jsou hojně používány na vhodných místech.

Další komentáře a hodnocení

Vyjádřete se k úrovni dosažených hlavních výsledků závěrečné práce, např. k úrovni teoretických výsledků, nebo k úrovni a funkčnosti technického nebo programového vytvořeného řešení, publikačním výstupům, experimentální zručnosti apod.

K prezentovaným výsledkům benchmarku a použité metodice mám několik výhrad:

- Popis profilování v sekci 3.2 - pokles počtu instrukcí a nárůst využití paměti neznamená, že nový kernel je rychlejší - pro takový závěr je potřeba porovnat skutečný čas výpočtu, ale z výsledků benchmarku to vypadá, že první 4 kernely 1.1-1.4 jsou téměř stejně rychlé.
- Všechny grafy v sekci 4.2 mají netypicky invertovanou vertikální osu (hodnoty blízké nule jsou nahoře a velké kladné hodnoty jsou dole).

- V popisu výsledků není zmíněno, jestli jsou matice uložené v row-major nebo column-major orientaci, ani jaký je datový typ (šablonový parametr RealType). Vypadá to na double, což je ok pro helios, ale ne pro gp1 (karta Quadro má 32x nižší výkon v double oproti float, na V100 je rozdíl jen 2x).
- V některých kernelech (1,2,3,6) je násobení parametrem "matrixMultiplicator" uvnitř hlavního výpočetního cyklu, ale dalo by se "vytknout" a provést těsně před zápisem do globální paměti - tím by počet instrukcí mohl klesnout až 2x. To se projeví hlavně v double precision na kartách GeForce a Quadro (např. gp1), kde jsou kernely 4 a 5 výrazně rychlejší, než ostatní problematické kernely.

III. CELKOVÉ HODNOCENÍ, OTÁZKY K OBHAJOBĚ, NÁVRH KLASIFIKACE

Shrňte aspekty závěrečné práce, které nejvíce ovlivnily Vaše celkové hodnocení. Uveďte případné otázky, které by měl student zodpovědět při obhajobě závěrečné práce před komisí.

Práce je velmi obsáhlá a pečlivě zpracovaná, ale popis jednotlivých optimalizací bych si představoval provedený lépe. Finální výsledky to chtělo před odevzdáním ještě jednou podrobně projít se školitelem, vychytat některé chyby a výpočet provést ještě jednou, ale chápu, že v časově omezeném prostoru se to plánuje špatně...

Předloženou závěrečnou práci hodnotím klasifikačním stupněm **C - dobře.**

Datum: 16.8.2024 Podpis: