ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ

Fakulta jaderná a fyzikálně inženýrská

Ing. Tomáš Oberhuber, Ph.D.
katedra matematiky
Fakulta jaderná a fyzikálně inženýrská
ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
Trojanova 13
120 00 PRAHA 2

Školitelský posudek diplomové práce studenta Bc. Matouše Fencla
„Paralelní algoritmy lineární algebry pro GPU“

Předkládaná diplomová práce se zabývá paralelní implementací operace násobení řídké symetrické matice s vektorem (sparse-matrix vector multiplication, SpMV) a řešení soustav lineárních rovnic pomocí přímé Gaussovy metody. Oboje je implementováno pro GPU pomocí nástroje CUDA, Gaussova eliminační metoda je implementována dokonce i s podporou rozhraní MPI pro GPU klastry.

První kapitola popisuje nástroje použité k implementaci. Jde zejména o prostředí CUDA pro vývoj programů pro karty od firmy NVidia. Nechybí zde ani základní popis hardwarové architektury těchto karet. Dále je zde velmi stručně zmíněno rozhraní OpenMP pro programování vícejádrových procesorů a také rozhraní MPI pro programování paralelních architektur s distribuovanou pamětí. Krátce je zde také zmíněna knihovna TNL, což je numerická knihovna vyvíjená na katedře matematiky na FJFI. Tato knihovna nabízí vyšší jednotnou vrstvu pro vývoj paralelních algortimů jak pro GPU tak pro vícejádrová CPU. Autor využil některé struktury této knihnovny a vyvinuté algoritmy by se následně naopak měly stát součástí této knihovny, která je volně dostupná na internetu.

Druhá kapitola se věnuje operaci násobení řídkých matic s vektorem. Jde o operaci, která je velice častá v řadě výpočetních algoritmů, zejména např. v iterativních metodách pro řešení soustav lineárních rovnic, jako jsou třeba metody Krylovových podprostorů. Jelikož jsou řídké matice často neregulární a heterogení datové struktury, je jejich efektivní ukládání a zejména násobení s vektorem úlohou, která se stále těší velké pozornosti ze strany komunity zabývající se vysoce vykonými výpočty (high-performance computing, HPC). Výrazně méně pozornosti je věnováno ukládání symetrických matic na GPU, kdy část matice nad diagonálou není nutné ukládat explicitně v paměti CPU nebo GPU, ale je možné ji rekonstruovat během výpočtu operace SpMV. Právě to je náplní této kapitoly. Autor na základě poměrně jednoduchého formátu Ellpack zkouší dvě modifikace. Tou první je využití atomických instrukcí při zápisu do výstupního vektoru operace SpMV. Tou druhou je pak dekompozice matice pomocí speciálního obarvení na třídy maticových řádku v rámcí nichž nebude docházet při zápisu do výstupního vektoru k žádným konfliktům.

V následující třetí kapitole autor testuje obě zmíněné modifikace na sadě řídkých symetrických matic. Poměrně překvapivě se ukázalo, že mnohem jednodušší přístup pomocí atomických instrukcí je efektivnější. Na základě tohoto zjištění již byla provedena i implementace podpory řídkých symetrických matic v knihovně TNL.

Čtvrtá kapitola se zabývá paralelní implementací Gaussovy metody na GPU. Pro odstranění některých čistě sekvenčních závislostí byla zvolena Gaussova-Jordanova metoda, která převádí vstupní matici na diagonální místo horní trojúhelníkové. To umožňuje získat více paralelních výpočtů v první fází a navíc zcela odpadá zpětná substituce, která se velice obtižně paralelizuje. Určitou nevýhodou tohoto přístupu je nemožnost výpočtu LU rozkladu tímto algoritmem. Je implementován algoritmus s a bez pivotingu, který zlepšuje numerickou stabilitu a eliminuje požadavek silné regularity vstupní matice. Autor dále provedl paralelizaci pomocí rozhraní MPI, čímž vznikl algoritmus schopný běžet na klastrech GPU karet.

V páté kapitole jsou prezentovány výsledky paralelní Gaussovy-Jordanovy eliminační metody. Zde záleží zejména na rozměrech matice soustavy. Pro dostatečně velké matice se podařilo dosáhnout až více než desetinásobného urychlení, což je velice dobrý výsledek. Paralelizace pomocí MPI byla bohužel testována pouze na dvou kartách, což ale není chyba autora. Naměřené urychlení 1.66 považuji za dobrý výsledek. U tabulky číslo 5.9 by měla být ještě dopočítána efektivita, která v nejlepším případě odpovídá hodnotě 0.83. To je dobrý výsledek. Bude ale ještě potřeba provést testy na více GPU, abychom vědeli lépe, jak dobrý tento algoritmus je.

Autor splnil kompletně celé zadání v míře, která předčila má očekávání. Není mnoho, co by bylo možné autorovy vytýkat. Na konzultace docházel pravidelně, alespoň pokud to způsob výuky umožňoval. Naopak oceňuju, že práci zvládadl dokončit i v režimu výuky na dálku, kdy jsme spolu komunikovali jen pomocí mailu a autor tak prokázal i schopnost samostatné práce. S textem samotným autor trochu zápasil, první verze byly hůře srozumitelné a obsahovaly i jazykové chyby, ale myslím, že vše se podařilo opravit a výsledný text je srozumitelný včetně prezentace výpočetních výsledků. Pokud by se v případě operace SpMV se symetrickými maticemi ukázalo, že algoritmus s maticovým obarvením je rychlejší, bylo by zřejmě možné tento výsledek publikovat v mezinárodním impaktovaném časopise. Přístup pomocí atomických operací je na samostatnou publikaci příliš jednoduchý, ale zřejmě tento výsledek využijeme v jiném širším článku.

Na závěr tedy mohu shrnout tvrzením, že dosažené výsledky jsou velice přínosné a s přihlédnutím ke všemu výše uvedenému navrhuji práci hodnotit známkou **A**, tedy **výborně**.

V Praze, 17. července 2020. Ing. Tomáš Oberhuber, Ph.D.