Posudek oponenta bakalářské práce

Student: Čelka Marek
Téma: Akcelerace aplikací na superpočítači pomocí jazyka Python (id 19653)
Oponent: Jaroš Jiří, Ing., Ph.D., UPSY FIT VUT

1. Náročnost zadání — průměrně obtížné zadání

Cílem práce je otestovat možnosti jazyka Python v paralelních aplikacích jak po stránce výkonnosti tak pro stránce produktivity programátora. Práce obsahuje jednoduché mikrotesty a jednu složitější aplikaci. Celkově lze náročnost zadání hodnotit jako středně obtížné.

2. Splnění požadavků zadání — zadání splněno s drobnými výhradami

Zadání bylo splněno až na dvě drobné výhrady:

- Autor v závěru tvrdí: "*Ďalším z cieľov bolo porovnať paralelné programy v programovacích jazykoch Python a C z hľadiska efektivity a náročnosti implementácie.*" Nic takového v práci není. Autor porovnal C a Python pouze na komunikačních mikrotestech. Pro násobení matic a rekonstrukci obrazu srovnání chybí. Autor navíc porovnal pouze výkon, vůbec se ale nevyjádřil k produktivitě práce a náročnosti na implementaci.
- Experimentální vyhodnocení na 1 uzlu považuji za naprosto nedostačující. Při rekonstrukci obrazu byly sice použity 4 uzly, ale úloha byla extrémně malá. Vhledem k tomu, že měl autor k dispozici celý cluster, mohl pracovat až s 2048 výpočetními jádry. Měl tedy možnost udělat velice zajímavé porovnání.

3. Rozsah technické zprávy — je v obvyklém rozmezí

Technická zpráva je v obvyklém rozsahu. Některé obrázky jsou však zbytečně velké a práci to neúměrně natahuje, viz strany 11, 12, 20, 21, 25, 26.

4. Prezentační úroveň předložené práce — 70 b. (C)

Prezentační úroveň technické zprávy je průměrná. Výhrady mám k:

- **abstraktu** - Abstrakt nemá být obsah, ale reklama - chybí mi zde základní výsledek, kterým by se autor rád pochlubil.
- **kapitole 2.3** - Kapitola není vůbec členěna do podkapitol, což vede na nepřehledný text.
- **kapitole 4** - Kapitola měl být lépe členěna. Autor měl vysvětlit co přesně che měřit, na jak velkých úlohách, na jakém počtu jader a co by testy chtěl ukázat.
- **závěru** - Závěr je hodně povrchní. Zde autor mohl mnohem lépe rozepsat dosažené výsledky.
- Chybí mi odkazy a ukázky na práce jiných autorů v oblasti distribuovaného počítání, kde byl Python reálně nasazen.

5. Formální úprava technické zprávy — 70 b. (C)

Po jazykové a typografické stránce je práce v pořádku. Chybí jen číslování některých rovnic, občas se vyskytuje desetinná tečka místo čárky a grafy v sekci mikrotestů nemají konzistentní popisky os.

6. Práce s literaturou — 60 b. (D)

Práci s literaturou nehodnotím příliš pozitivně. Práce necituje žádnou publikaci, která by se zabývala využitím Pythonu pro vědecké výpočty a počítání na superpočítačích, přičemž takových publikací jsou již stovky. Seznam literárních pramenů obsahuje celkem 10 elektronických manuálů, u kterých je citace uvedena velice jednoduše ([1]-[10]).

7. Realizační výstup — 75 b. (C)

Implementační část je celkem v popořádku, má ale svoje mouchy. Např. *collectComm* skončí na segmentation fault pokud se nezadají parametry (help se sám nevypíše). Pokud spustím paralelní rekonstrukci obrazu a dám -help, dostanu nápovědu tolikrát, kolik mám jader. Použití pozičních argumentů nepovažuji za správné řešení. Chybí možnost upsamplovat vstupní obrázek větší rozlišení.
Dokumentace je provedena v Doxygen formátu, je však docela povrchní. Nejvíce mi vadí, že celý výpočet rekonstrukce obrazu je v jedné funkci main, kde nejsou téměř žádné komentáře, a člověk si musí sám dohledat, co je výpočet a co zápis do souboru.

Z experimentálních výsledků jsem velice zklamán. Přestože měl autor k dispozici neomezené zdroje na superpočítači, použil maximálně 4 uzly a to ještě na příliš malé úloze (cca 2MB dat). Většina měření je pak pouze na jednom uzlu, což pro MPI nedává smysl.

Měření rychlosti zápisu do souboru, Obr. 8.3, je buď úplně špatně, nebo autor vynesl nesprávná data. Kupeckými počty odvodíme, že největší obrázek 1920x5120 má max 100 MB v textovém formátu. Dle autorova měření trvá zápis 50s v sekvenční verzi a 5s v paralelní verzi, tedy zapisujeme rychlostí 2-20MB/s. (Anselm zvládá cca 2GB/s).

8. Využitelnost výsledků

Tato práce je promarněnou šancí ukázat jak se Python chová na velkém superpočítači. Kdyby autor použil mnohem větší obrázky (cca 1GB) a zhruba 1024 výpočetních jader, mohla práce sloužit jako reference mnoha dalším vývojářům.

9. Otázky k obhajobě

- Jak jste měřil rychlost zápisu do souboru?
- Jak si vysvětlujte, že jste dosáhl výkonu odpovídajícímu 0.01 - 0.1% teoretické propustnosti diskového pole?
- Proč jste nepoužil mnohem větší počty procesorových jader, i fronta qexp jich má 128?
- Jak si vede vaše implementace násobení matic v porovnaní s funkcí dot() z Pythonu, aneb jak jste dobrý?

10. Souhrnné hodnocení **70 b. dobře (C)**

Mezi pozitivní aspekty této práce patří dobrá implementace a srozumitelný text. Mezi negativa pak velmi omezené experimentální vyhodnocení, horší práce s literaturou a další drobné nedostaly popsané výše. Celkově hodnotím práci stupněm **dobře (C - 70 bodů)**.

Prohlášení: Uděluji VUT v Brně souhlas ke zveřejnění tohoto posudku v listinné i elektronické formě.

V Brně dne: 1. června 2017

.....
podpis