

TCP vrstva pro verifikační nástroj DIVINE

Jiří Weiser

12. února 2016

DIVINE

- explicitní model checker
- pracuje s rozsáhlými grafy
- podpora paralelizmu pro urychlení výpočtu
 - ve sdílené paměti
 - v distribuované paměti

Paralelizmus ve sdílené paměti

- pouhé urychlení výpočtu
- podpora v moderních OS
 - rozhraní OS (OS Windows)
 - POSIX vlákna (*NIX-like OS)
- vše je připraveno pro použití

Paralelizmus v distribuované paměti

- urychlení výpočtu
- umožnění doběhnutí výpočtu
 - velké grafy se nemusí vejít do paměti
- většina OS nemá nativní podporu
- potřeba použít specifické nástroje
 - knihovna MPI
 - *de-facto* standard
 - knihovna PVM
 - vývoj pozastaven
- nástroj DIVINE používá knihovnu MPI

- podpora pro distribuované výpočty
- obecný standard
 - různé architektury
 - různé topologie
 - různé výpočetní modely
- poskytuje
 - komunikační primitiva
 - podporu pro běh
- existuje několik implementací
 - Open MPI, MPICH, proprietární. . .

Využití MPI nástrojem DIVINE

- malá podmnožina funkcí
 - 3 funkce zajišťující komunikaci
- problematický paralelní přístup
 - většinou není dostupný
 - případně neefektivní
- komunikace vedena skrze jedno vlákno
 - zbytečná synchronizace
 - nutnost kopírovat data
- samo MPI využívá vyrovnávací paměti
 - další režie spojená s kopírováním

Předpoklady:

- výpočetní stroje propojeny běžným síťovým spojením
- vnitřní “nezabezpečená,, síť
- nízký počet strojů
- nízký počet komunikačních primitiv

Cíle:

- paralelní přístup ke komunikačnímu rozhraní
- snížení objemu kopírovaných dat

- nová komunikační vrstva
- jiný model distribuované verifikace
 - nově možnost mít jako službu
- TCP protokol
- protokol pro ustanovení sítě
- propojení každý s každým

Různé přístupy ke komunikaci v rámci stroje:

1. jeden příjemce (scénář shared)
 - výkonné vlákno
 - dotazování
 - vlastní kanál
 - dedikované vlákno
 - příjem a zodpovídání dotazů
 - přiděluje práci výkonným vláknům
2. více příjemců (scénář dedicated)
 - výkonné vlákno
 - komunikace
 - v rámci stroje
 - stejná vlákna na jiných strojích
 - distribuce práce bez “dozoru,,
 - dedikované vlákno
 - udržování spojení

MPI

- *latence*: výborné výsledky
- *škálování*: horší škálování přes vlákna

Jeden příjemce

- *latence*: špatné výsledky
- *škálování*: vhodný pro jedno vlákno

Více příjemců

- *latence*: uspokojivé výsledky
- *škálování*: škáluje přes vlákna

V práci je zmíněna knihovna Boost Asio jako jedno z možných řešení, ale její zavržení mi nepřipadá zcela přesvědčivě zdůvodněno. Je závislost na externí knihovně tak velký problém, nebo by toto řešení mělo i jiné nevýhody (resp. nedostatky výhod)?

- výhodou je funkčnost na OS Windows
 - trochu odlišné rozhraní oproti POSIX
 - DIVINE nejde na OS Windows (aktuálně)

Otázka oponenta

- Asio je nabízena jako
 - knihovna Boostu
 - samostatná knihovna (závislá na Boostu)
- závislost na Boostu jsme již v projektu měli
 - přináší to malé trable
 - udržování aktuální verze
- rozdílné rozhraní oproti POSIX
 - stejně je třeba znát POSIX rozhraní
 - využití malé části
 - POSIX se stejně používá
- návrh síťového rozhraní do C++17
 - autoři Asio
 - použít spíš standard

Využitím knihovny Asio bychom nezískali žádné výhody.

V závěru tvrdíte, že použití Vaší implementace je pro DIVINE vhodné, nejedná se o účelové tvrzení? V čem naopak vidíte nedostatky Vaší implementace?

- obdobné rozhraní jako MPI
- řešení je funkční
- distribuovaný výpočet jako služba
 - s drobnou úpravou lze provozovat jako MPI
- poskytuje paralelní přístup
 - zlepšení oproti MPI přístupu
 - snížení množství kopírování
- malého rozsahu
 - případné zásahy nebudou komplikované

Testy, které nebyly z časových důvodů zahrnuty v diplomové práci:

- posílání větších bloků pamětí (1KB)
 - další zlepšení mé implementace oproti MPI
 - obzvláště scénář dedicated
- zpracování velkých grafů scénářem shared
 - docházelo k přehlcení systémových front
 - nekompletní zaslání zpráv
 - MPI trpělo stejnými problémy (v menší míře)
- časové limity operací
 - limit na operaci 5 vteřin
 - je třeba řešit při odesílání

Přes uvedené problémy se domnívám, že