

Porovnání paralelní implementace B algoritmu v Javě a C/C++

Petr Pernička¹

1 Úvod

Předpověď dopravy je pokus o odhadnutí počtu vozidel či cestujících, kteří chtějí použít dopravní síť k přepravě mezi různými místy zájmu. Tato oblast dopravního plánování je čím dál tím důležitější pro budoucí vylepšování dopravních sítí, zvláště se stále rostoucím počtem účastníků provozu. Jedním z modelů určující metodiku pro předpověď dopravy je takzvaný čtyřkrokový model, jehož posledním krokem je problém přiřazení dopravy.

Zde je představen problém přiřazení dopravy a Algoritmus B jakožto jeden z algoritmů řešící statickou variantu tohoto problému. Následně jsou zde uvedeny výsledky porovnání implementace paralelního Algoritmu B v Javě a C++.

2 Problém přiřazení dopravy

Ve studii Mitchell a Rapkin (1954) pokládající základy dopravní analýzy byl představen čtyřkrokový model, který je používán pro předpověď rozložení dopravních požadavků cestujících. Vstupem modelu je graf reprezentující dopravní síť, kde hrana reprezentuje silnici a vrchol pak křižovatku nebo zónu, což jsou agregovaná prostorová data, např. obec či sousedství. Zóny v modelu slouží jako zdroje a stoky dopravního toku.

V prvních třech krocích modelu se identifikuje velikost toků, které začínají a končí v každé ze zón. Tento výsledek je zaznamenán ve formě matice počátků a cílů, kde prvek na pozici ij reprezentuje tok ze zóny i do zóny j . Graf s maticí jsou vstupem posledního kroku — přiřazení dopravy. Ve statickém přiřazení dopravy jsou prvky matice konstanty a výsledek reprezentuje průměr přes modelovaný časový úsek.

Cílem přiřazení dopravy je rozdělit toky z matice mezi cesty v dopravní síti tak, aby platila uživatelská rovnováha formulovaná ve Wardrop (1952). Ta nastává právě tehdy, když žádný cestující nemůže snížit dobu své jízdy volbou alternativní trasy. Jelo-li by z i do j nějaké vozidlo samotné, zvolilo by nejkratší cestu, protože je nejrychlejší. Jakmile se však dopravní tok začne zvyšovat, nemusí být nejkratší cesta ta nejrychlejší. Tento dopravní odpor je charakterizován tzv. cenovými funkcemi. Nejpoužívanější z nich je BPR funkce představena v U.S. Bureau of Public Roads (1964).

3 Algoritmus B

Algoritmus B z Dial (2006) je numerický grafový algoritmus pro statické přiřazení dopravy a dosažení uživatelské rovnováhy. Stejně jako ostatní algoritmy pro statické přiřazení dopravy je B algoritmus iterativní — s každou iterací se více a více blíží uživatelské rovnováze. Efektivita B vychází z rozdělení sítě na keře, což jsou acyklické podgrafy s jediným kořenem.

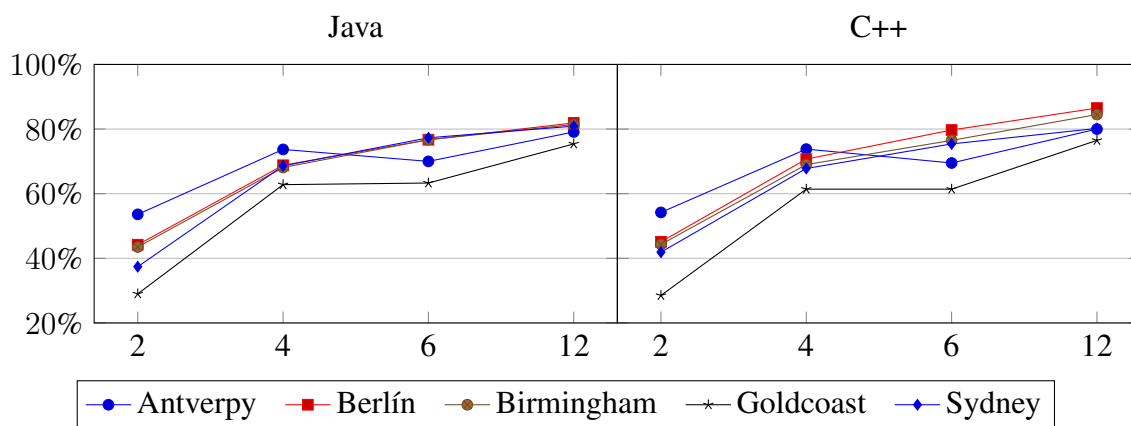
¹ student bakalářského studijního programu Informatika a výpočetní technika, specializace Informatika, e-mail: pperncka@students.zcu.cz

Algoritmus začíná tím, že si pro každou zónu v síti vytvoří keř v ní zakořeněný. V keři jsou pak jen hrany začínající ve vrcholu, který je blíže kořeni než vrchol, ve kterém hrana končí. Všechny tok z i do j je na začátku přiřazen nejkratší cestě mezi těmito zónami.

Následuje iterační fáze, kdy se v každé iteraci zpracují všechny keře a to povede ke konvergenci k uživatelské rovnováze. Každému keři se nejdříve zlepší topologie — jsou přidány některé hrany za zachování acyklicity. Následně se naleznou stromy minimálních a maximálních cest použitím standardního algoritmu pro nalezení těchto cest v acyklickém grafu. Poté se pomocí Newtonova algoritmu vyváží toky mezi minimální a maximální cestou ke každému vrcholu tak, aby mezi cestami nastala uživatelská rovnováha. Zpracování keře končí odstraněním hran, na kterých neteče žádný tok.

Samotný algoritmus končí poté, co dosáhl požadované přesnosti.

4 Výsledky porovnání



Obrázek 1: Urychlení paralelní implementace oproti sekvenční na počítači s AMD Ryzen 5 1600X. Horizontální osa značí počet vláken.

Algoritmus byl porovnáván na pěti reálných sítích. Bylo zjištěno, že implementace v C++ je o $\sim 8\%$ rychlejší a potřebuje o $\sim 4\%$ méně paměti. Nastaly však případy, kdy byla Java neznatelně rychlejší. Oba jazyky paralelizací urychlují algoritmus skoro stejně, viz Obrázek 1. S 12 vláknými může být algoritmus rychlejší až o 80%. Někdy však zvýšení počtu vláken dokáže zvýšit počet iterací potřebných ke konvergenci natolik, že výsledně algoritmus potřebuje více času.

Literatura

- Dial, R. B. (2006) A path-based user-equilibrium traffic assignment algorithm that obviates path storage and enumeration. *Transportation Research Part B*. Amsterdam, Elsevier, Volume 40, pp 917-936.
- Mitchell, R. B., Rapkin, C. (1954) *Urban Traffic: A Function of Land Use*. New York, Columbia University Press.
- United States. Bureau of Public Roads (1964) *Traffic Assignment Manual: For Application with a Large High Speed Computer*. U.S. Department of Commerce, Bureau of Public Roads, Office of Planning, Urban Planning Division.
- Wardrop, J. G. (1952) Some Theoretical Aspects of Road Traffic Research. *Proceedings of the Institution of Civil Engineers*. ICE Publishing, Volume 1, pp. 325-362.