

Semestrálna práca MI-PAR 2013/2014:

Paralelný algoritmus pre rozklad obdĺžnikov

Jakub Melezínek

Martin Klepáč

November 22, 2013

1 Definícia problému

Našou úlohou bolo vytvoriť program, ktorý implementuje usporiadanie obdĺžnikov v 2D mriežke so zachovaním minimálneho celkového obvodu týchto obdĺžnikov.

Obsah obdĺžnika je daný hodnotou uloženou v 2D mriežke, pričom tento element musí byť súčasťou obdĺžnika s daným obsahom.

Jednotlivé obdĺžniky sú vzájomne disjunktné až na spoločné vrcholy a hrany, pričom zjednotenie všetkých obdĺžnikov pokrýva pôvodnú mriežku.

Riešení, ako rozdeliť mriežku na jednotlivé obdĺžniky, môže byť viac – v takom prípade hľadáme riešenie s minimálnym obvodom – zároveň ale riešenie nemusí existovať.

2 Formát vstupu, výstupu

Formálne, vstup vyjadríme pomocou

- a, b = prirodzené čísla predstavujúce rozmery mriežky
- $H[1..a][1..b]$ = mriežka
- n = prirodzené číslo predstavujúce počet obdĺžnikov vo vnútri mriežky

Výstupom algoritmu je okrem celkového obvodu dielčích obdĺžnikov vyfarbená mriežka, t.j. mriežka, v ktorej každému elementu je priradené písmeno abecedy, ktoré jednoznačne identifikuje obdĺžnik, ktorého je bod súčasťou.

3 Implementácia sekvenčného riešenia

Primárny cieľ sekvenčného riešenia, na ktorom ďalej stavíme v paralelnej implementácii, spočíva v nájdení a následnom prehľadaní celého stavového priestoru množiny potenciálnych riešení.

Veľkosť mriežky	T(n) [s]
15x15	29
20x20	473
21x21	571
23x23	983

Table 1: Trvanie sekvenčného výpočtu

Stavový priestor v našom prípade rozumieme množinu mriežok, do ktorých postupne vkladáme obdĺžniky s dvojicou parametrov

1. *shape* - veľkosť obdĺžnika (dĺžka x šírka)
2. *position* - pozícia v mriežke

Očividne, *shape* obdĺžnika je určený jeho plochou, zatiaľ čo *position* je vlastnosťou mriežky a ostatných obdĺžnikov v nej existujúcich - napr. obdĺžnik nedokážem vložiť do mriežky, pokiaľ jeden z jeho rozmerov presahuje veľkosť mriežky alebo je okolie obdĺžnika posiate inými, už zafixovanými obdĺžnikmi.

Nami implementované triviálne sekvenčné riešenie potom spočíva v prehľadaní celého stavového priestoru bez orezávania neperspektívnych mriežok a výberom ideálneho riešenia z množiny riešení.

Trvanie sekvenčného výpočtu na výpočtovom klastri star.fit.cvut.cz je znázornené v tabuľke 1.