Zpráva o řešení domácího úkolu č. 1

**Exaktní metody řešení rozhodovacího problému batohu**

<https://moodle-vyuka.cvut.cz/mod/assign/view.php?id=153050>

0/1 problém batohu je NP-těžký optimalizační problém. Tato zpráva informuje o výsledcích experimentů, jejichž účelem bylo potvrdit hypotézu, že rozhodovací verze tohoto problému nabývá exponenciální časové složitosti. Pro experiment posloužily dvě sady testovacích dat – NR a ZR – s různými velikostmi instancí, na kterých byli otestovány celkem dva algoritmy. Pro několik prvních sad instancí dané velikosti (v rámci sady testovacích dat) byly zaznamenávány doby běhu (počet navštívených konfigurací + čas CPU) a z nich vycházející průměrné a maximální časy. Tyto dva parametry byly zaznamenány do grafu pro každý dílčí algoritmus. Pro zvolenou sadu instancí dané velikosti byly vytvořeny histogramy, které také zobrazují rozdíly mezi těmito algoritmy.

**ZVOLENÉ ALGORITMY**

**1**

Označení v experimentech: **BRUTE FORCE**

Popis: Algoritmus prochází všechny konfigurace až do samotných listů rekurze, ve kterých vyhodnotí podmínky (min. cena a max. váha). Jakmile konfigurace splňuje tyto podmínky, algoritmus skončí s návratovou hodnotou *True*.

**2**

Označení v experimentech: **BETTER** **BRUTE FORCE**

Popis: Totožný s BRUTE FORCE. V rekurzivní funkci je navíc obsažena podmínka pro max. váhu. Pakliže přidávaná věc (společně s již přidanými) překročí nosnost batohu, algoritmus pokračuje tak, jako by danou věc do batohu vůbec nepřidal (a testuje další lehčí věci).

**3**

Označení v experimentech: **BRANCH & BOUND**

Popis: Kromě podmínky pro max. váhu jsou v rekurzivní funkci i další podmínky pro min. cenu. Pakliže přidávaná věc splní podmínku pro nosnost a překročí vyžadovanou min. cenu, je nalezeno řešení a algoritmus skončí s návratovou hodnotou *True*. Jestliže je splněna pouze nosnost, algoritmus vypočítá ceny dalších věcí přidávaných do batohu v budoucnu. Pokud dosavadní cena včetně ceny přidávané věci v součtu s cenami budoucích věcí nepřekročí min. cenu, výpočet skončí s návratovou hodnotou *False*.

**PLATFORMA**

Hardware: Notebook Lenovo, Intel Core i5-8250U 1.6 GHz, 4 GB RAM

Operační systém: Windows 10 Home, 64bit

Programovací jazyk: Python

**PROVEDENÉ EXPERIMENTY**

**1**

Algoritmus: BRUTE FORCE

Testovací sada: NR

Velikosti instancí: 4, 10, 15, 20



Data:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Velikost instance** | **Maximální počet konfigurací** | **Průměrný počet konfigurací** | **Maximální čas CPU (ms)** | **Průměrný čas CPU (ms)** |
| 4 | 16 | 11.308 | 15.625 | 0.03125 |
| 10 | 1024 | 627.01 | 15.625 | 0.3125 |
| 15 | 32768 | 18235.032 | 31.25 | 9.90625 |
| 20 | 1048576 | 569966.918 | 1093.75 | 352.5 |

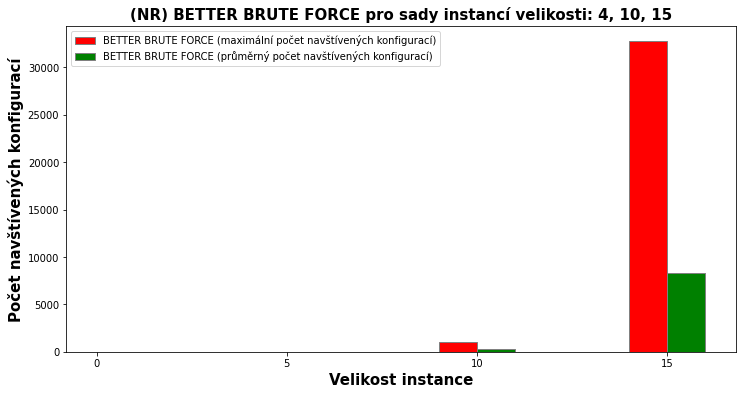
Komentář autora experimentu: Z výše uvedeného grafu bylo vynecháno měření pro instance velikosti 20, jelikož by v důsledku exponenciální tendence způsobilo daleko větší kontrast vůči předchozím výsledkům. Ostatně i instance velikosti 10 a 15 tento schodek způsobují vůči instanci velikosti 4, která je v grafu zcela nepostřehnutelná. Výsledky z tabulky dokumentují postupné ztěžování a zpomalování výpočtu v závislosti na velikosti instance.

**2**

Algoritmus: BETTER BRUTE FORCE

Testovací sada: NR

Velikosti instancí: 4, 10, 15, 20



Data:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Velikost instance** | **Maximální počet konfigurací** | **Průměrný počet konfigurací** | **Maximální čas CPU (ms)** | **Průměrný čas CPU (ms)** |
| 4 | 15 | 4.492 | 15.625 | 0.03125 |
| 10 | 1023 | 264.15 | 15.625 | 0.15625 |
| 15 | 32766 | 8299.524 | 46.875 | 5.21875 |
| 20 | 1048575 | 285881.952 | 1093.75 | 187.0625 |

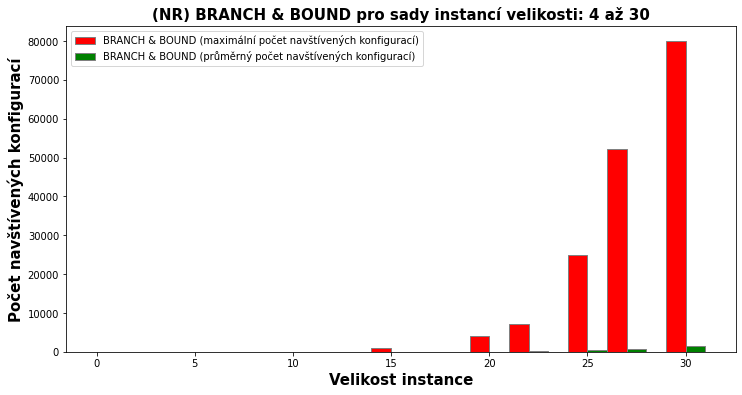
Komentář autora experimentu: Podobně jako u předchozího experimentu. Nutno podotknout, že podmínka pro max. váhu (jediný rozdíl vůči BRUTE FORCE) urychluje výpočet pouze nepatrně při porovnání výše uvedených dat.

**3**

Algoritmus: BRANCH & BOUND

Testovací sada: NR

Velikosti instancí: 4, 10, 15, 20, 22, 25, 27, 30, 32, 35



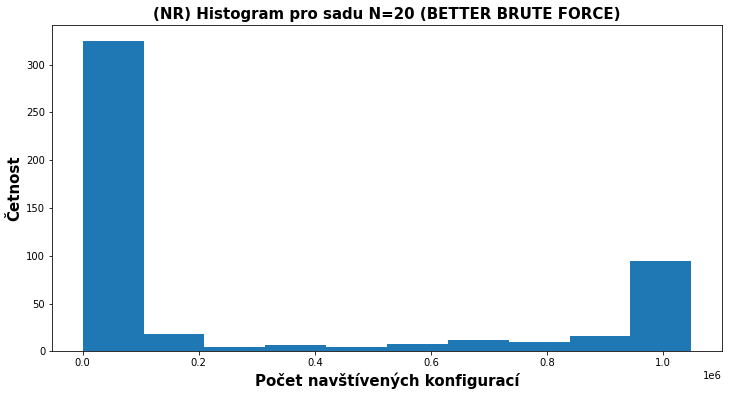
Data:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Velikost instance** | **Maximální počet konfigurací** | **Průměrný počet konfigurací** | **Maximální čas CPU (ms)** | **Průměrný čas CPU (ms)** |
| 4 | 4 | 0.526 | 15.625 | 0.03125 |
| 10 | 53 | 1.766 | ≈0.0 | ≈0.0 |
| 15 | 861 | 8.532 | 15.625 | 0.03125 |
| 20 | 4114 | 42.434 | 31.25 | 0.34375 |
| 22 | 7182 | 92.982 | 62.5 | 0.875 |
| 25 | 24814 | 447.796 | 281.25 | 4.34375 |
| 27 | 52114 | 628.61 | 390.625 | 7.46875 |
| 30 | 79966 | 1375.77 | 2890.625 | 28.40625 |
| 32 | 795171 | 8650.222 | 10734.375 | 103.75 |
| 35 | 1022719 | 11179.588 | 19171.875 | 238.1875 |

Komentář autora experimentu: Pokud se podíváme na průměrný počet navštívených konfigurací (tj. zelené sloupce) u BRUTE FORCE, BETTER BRUTE FORCE a zde u BRANCH & BOUND, lze si povšimnout rostoucího rozdílu vůči maximálnímu počtu navštívených konfigurací (tj. červené sloupce). Toto je nejspíše způsobeno postupnou optimalizací výpočtu. U uvedených časů CPU si lze dále povšimnout mírných nesrovnalostí u nejmenších velikostí instance. Tyto časy jsou pravděpodobně menší než dané nejmenší rozlišitelné hodnoty.

**HISTOGRAMY**







Komentář autora experimentu: Všechny histogramy naměřených hodnot mají konvexní tvar. Postupně lze pozorovat přesun naměřených hodnot do levé části histogramu. Toto je nejspíše způsobeno tím, že o většině instancí se v důsledku optimalizací rozhodne již v prvních okamžicích výpočtu.

Předchozí experimenty byly provedeny obdobně i pro testovací sadu ZR, jak lze číst dále…

**4**

Algoritmus: BRUTE FORCE

Testovací sada: ZR

Velikosti instancí: 4, 10, 15, 20



Data:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Velikost instance** | **Maximální počet konfigurací** | **Průměrný počet konfigurací** | **Maximální čas CPU (ms)** | **Průměrný čas CPU (ms)** |
| 4 | 16 | 11.056 | ≈0.0 | ≈0.0 |
| 10 | 1024 | 559.398 | 15.625 | 0.3125 |
| 15 | 32768 | 16729.86 | 31.25 | 9.15625 |
| 20 | 1048576 | 527022.88 | 937.5 | 312.375 |

**5**

Algoritmus: BETTER BRUTE FORCE

Testovací sada: ZR

Velikosti instancí: 4, 10, 15, 20



Data:

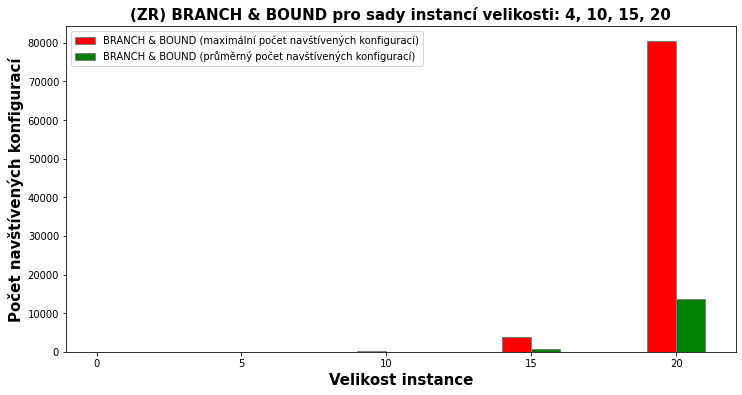
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Velikost instance** | **Maximální počet konfigurací** | **Průměrný počet konfigurací** | **Maximální čas CPU (ms)** | **Průměrný čas CPU (ms)** |
| 4 | 12 | 6.436 | ≈0.0 | ≈0.0 |
| 10 | 825 | 399.638 | 15.625 | 0.25 |
| 15 | 27879 | 13069.94 | 31.25 | 8.4375 |
| 20 | 929327 | 434649.152 | 875.0 | 288.46875 |

**6**

Algoritmus: BRANCH & BOUND

Testovací sada: ZR

Velikosti instancí: 4, 10, 15, 20, 22



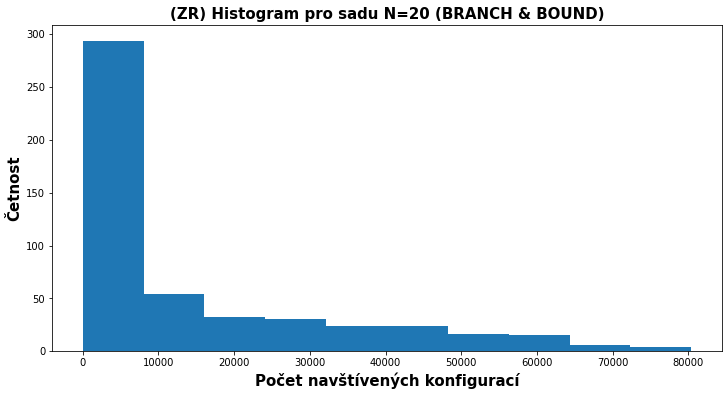
Data:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Velikost instance** | **Maximální počet konfigurací** | **Průměrný počet konfigurací** | **Maximální čas CPU (ms)** | **Průměrný čas CPU (ms)** |
| 4 | 7 | 1.574 | 15.625 | 0.03125 |
| 10 | 209 | 41.648 | 15.625 | 0.125 |
| 15 | 3724 | 713.11 | 31.25 | 3.25 |
| 20 | 80394 | 13580.194 | 312.5 | 67.21875 |
| 22 | 299221 | 39946.078 | 1046.875 | 192.34375 |

**HISTOGRAMY**







**ZÁVĚR**

Při srovnání testovací sady ZR vůči NR lze vydedukovat, že sada ZR je výpočetně výrazně náročnější než sada NR. U sady NR jsme pomocí BRANCH & BOUND počítali až do velikosti instance 35 v přijatelném čase (pro danou platformu), kdežto u sady ZR max. do velikosti 22. Z tohoto srovnání zároveň vyplývá, že střední hodnota výpočetní složitosti závisí na různých sadách instancí (průměrný počet navštívených konfigurací byl u sady ZR větší než u NR). Instance jsou pravděpodobně schválně zvolené tak, aby rekurze při jejich řešení (i s uvážením různých optimalizací) skončila vždy později. Algoritmus BRUTE FORCE je nejhorší možné řešení pro problém batohu. BETTER BRUTE FORCE urychluje výpočet pouze nepatrně. Avšak BRANCH & BOUND prořezává stavový prostor podstatně více, a tak je výpočet výrazně rychlejší. Všechny výše uvedené grafy a výsledky vykazují exponenciální růst (i v případě BRANCH & BOUND), tudíž lze prohlásit hypotézu experimentů za potvrzenou. Nejhorší případy nebyly měřeny (z důvodů velké časové náročnosti na danou platformu), avšak podle potvrzené hypotézy nejspíše vyhovují očekávané exponenciální závislosti.