Zpráva o řešení domácího úkolu č. 3

**Experimentální hodnocení kvality algoritmů**

<https://moodle-vyuka.cvut.cz/mod/assign/view.php?id=153057>

0/1 problém batohu je NP-těžký optimalizační problém. Pro **exaktní** řešení tohoto problému lze použít algoritmy a metody, jako např. *Brute Force*, *Branch & Bound* nebo *dynamické programování* (s dekompozicí dle ceny/váhy). Rychlejší alternativu představují **heuristiky**, jako např. *Greedy*, avšak je nutné u nich počítat s určitou mírou chybovosti. Vlastnosti těchto algoritmů lze zkoumat nejen v závislosti na velikosti instance tohoto problému (jako doposud), ale také např. na poměru kapacity batohu k sumární váze, korelaci váhy s cenou nebo granularitě. Můžeme tak lépe porozumět těmto metodám a jejich vlastnostem.

Tato zpráva informuje o výsledcích experimentů, jejichž účelem bylo vyhodnotit závislost **kvality** a **rychlosti** zmíněných algoritmů (jmenovitě tedy *Brute Force*, *Branch & Bound*, *dynamické programování* dle váhy a *Greedy*) na **poměru kapacity batohu k sumární váze**, **korelaci váhy s cenou** a **granularitě** (včetně zkoumání **robustnosti** těchto algoritmů). Pro experiment byly použity různé sady instancí generované specializovaným generátorem. Pro všechny tyto instance byly zaznamenávány doby běhu (tj. **průměrný čas CPU**) a chyby (tj. **průměrná chyba**). Tyto údaje jsou zobrazeny v doprovodných grafech a tabulkách dále v tomto dokumentu.

**ZVOLENÉ ALGORITMY**

**1**

Název: **HRUBÁ SÍLA**

Označení v experimentech: **BF**

Popis: Algoritmus prochází všechny konfigurace až do samotných listů rekurze, ve kterých vyhodnotí podmínky (tj. max. váhu). Ze všech konfigurací, které vyhovují těmto podmínkám, algoritmus vybere tu nejhodnotnější.

**2**

Název: **METODA VĚTVÍ A HRANIC**

Označení v experimentech: **B&B**

Popis: Algoritmus prochází všechny konfigurace až do samotných listů rekurze a ukládá nejlepší nalezené řešení. Po cestě kontroluje podmínky na **max. váhu** a **min. cenu**. Pokud dosavadní sestava věcí překročí nosnost batohu, algoritmus pokračuje jinou větví rekurze. Obdobně pokud ceny následujících věcí (společně s již dříve přidanými) nepřekročí cenu nejlepšího řešení. Algoritmus tak vždy vrací sestavu věcí s nejlepší cenou.

**3**

Název: **DYNAMICKÉ PROGRAMOVÁNÍ**

Označení v experimentech: **DP**

Popis: Výhoda dynamického programování spočívá v ukládání mezivýsledků, které jsou nadále používány pro následující výpočty. Algoritmus takové výsledky (v našem případě) ukládá do 2D pole. V závislosti na tom, kterou dekompozici používáme (dle váhy nebo ceny), do tohoto pole ukládáme průběžné ceny, příp. váhy pro různé sestavy věcí. Algoritmus pak tyto vypočtené ceny/váhy používá při rekurzivním prohledávání, čímž dosahuje zrychlení výpočtu. Pro experimenty uvedené dále v tomto dokumentu byla použita dekompozice **dle váhy**.

**4**

Název: **GREEDY**

Označení v experimentech: **GR**

Popis: Algoritmus nejprve heuristicky seřadí všechny věci podle klesajícího poměru cena/váha. Z této posloupnosti jsou následně vybírány jednotlivé věci, dokud je batoh unese. Takový algoritmus však nepracuje přesně, vzato s vyšší rychlostí.

**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**

Celkem bylo provedeno 13 experimentů rozdělených do samostatných bloků podle testovaných parametrů generátoru instancí a robustnosti. Tyto parametry byly nejdříve testovány v pilotních experimentech pro jednotlivé algoritmy. Jakmile byly zjištěny jakékoliv závislosti, byly provedeny i rozsáhlé experimenty, pro které jsou v tomto dokumentu k dispozici grafy a tabulky s daty.

**[POMĚR KAPACITY BATOHU K SUMÁRNÍ VÁZE]**

Pilotní experimenty odhalily závislost **výpočetní náročnosti** algoritmů B&B, DP (dle váhy) a **kvality** GR na tomto parametru. Naopak u algoritmů BF a GR nebyla zjištěna žádná podobná časová závislost. Pro experimenty v této sekci bylo použito následující nastavení generátoru:

./kg2 -n 10 -N 50 -W 5000 -C 5000 -m 0.1–0.9

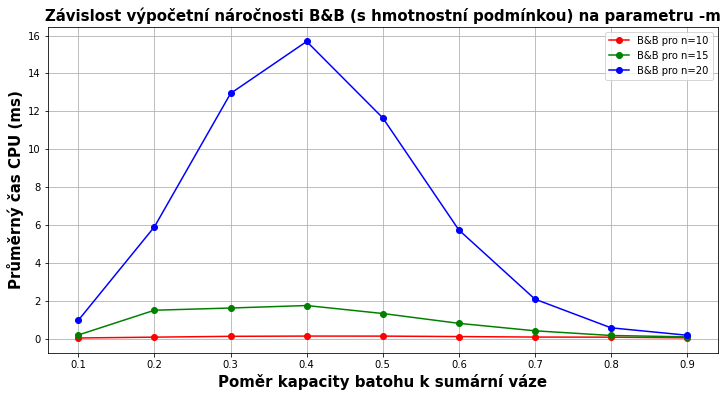
./kg2 -n 15 -N 50 -W 100 -C 10000 -m 0.1–0.9

./kg2 -n 17 -N 50 -W 6000 -C 4000 -m 0.1–0.9

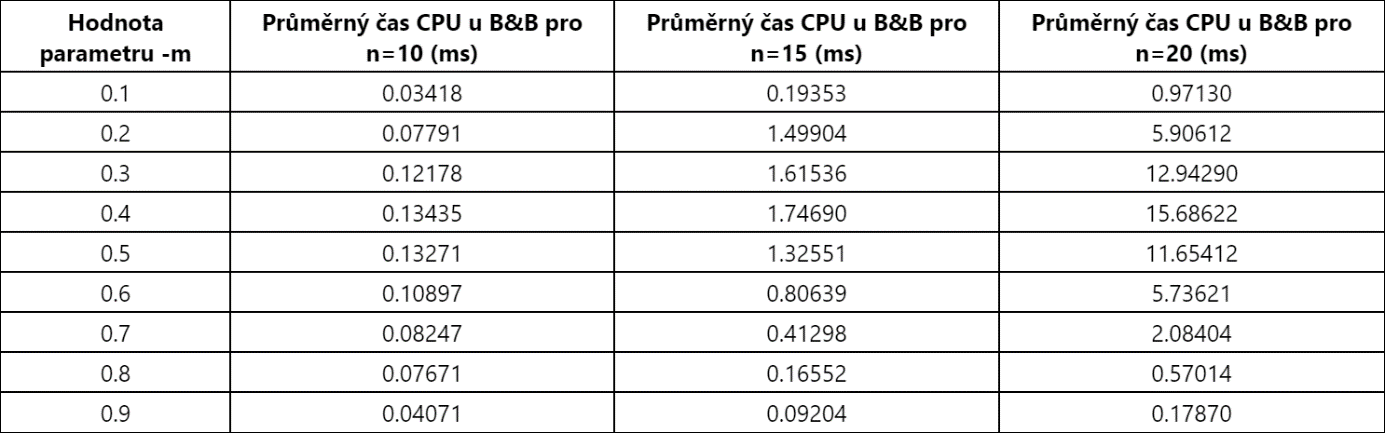
./kg2 -n 20 -N 50 -W 10000 -C 100 -m 0.1–0.9

**1**

Graf:



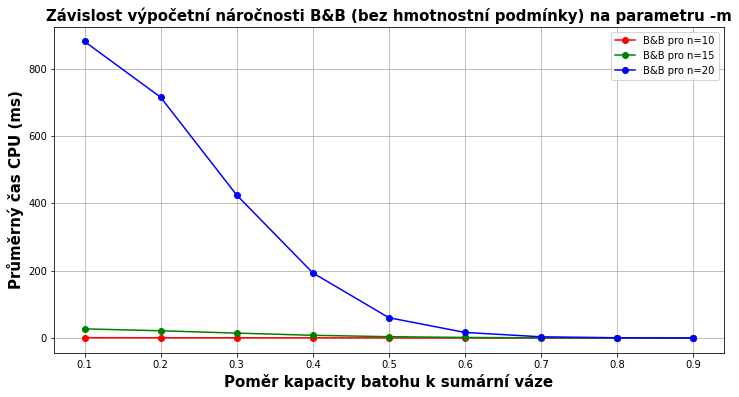
Data:



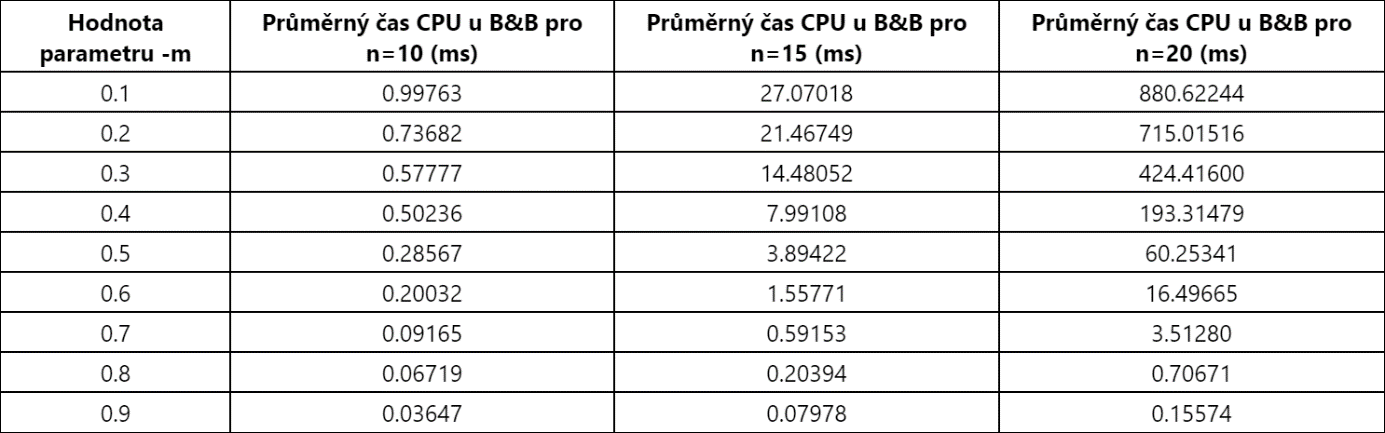
Komentář autora experimentu: V tomto a následujícím experimentu byly použity dva druhy B&B – včetně podmínky na **max. váhu** a bez ní. Tato podmínka pouze kontroluje, zdali nebyla překročena nosnost batohu a podle toho ořezává stavový prostor. Všechny křivky pro různé velikosti instancí a další parametry mají „kopcovitý“ charakter. Rostoucí část křivek pravděpodobně způsobuje tato podmínka, poněvadž nosnost je pro tyto hodnoty parametru velice malá, a tudíž dochází k onomu ořezávání stavového prostoru. Na druhou stranu klesající část křivek nejspíše způsobuje podmínka na **min. cenu**.

**2**

Graf:



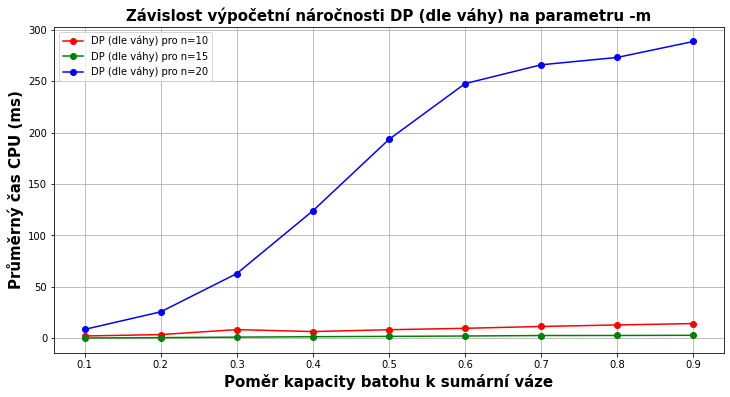
Data:



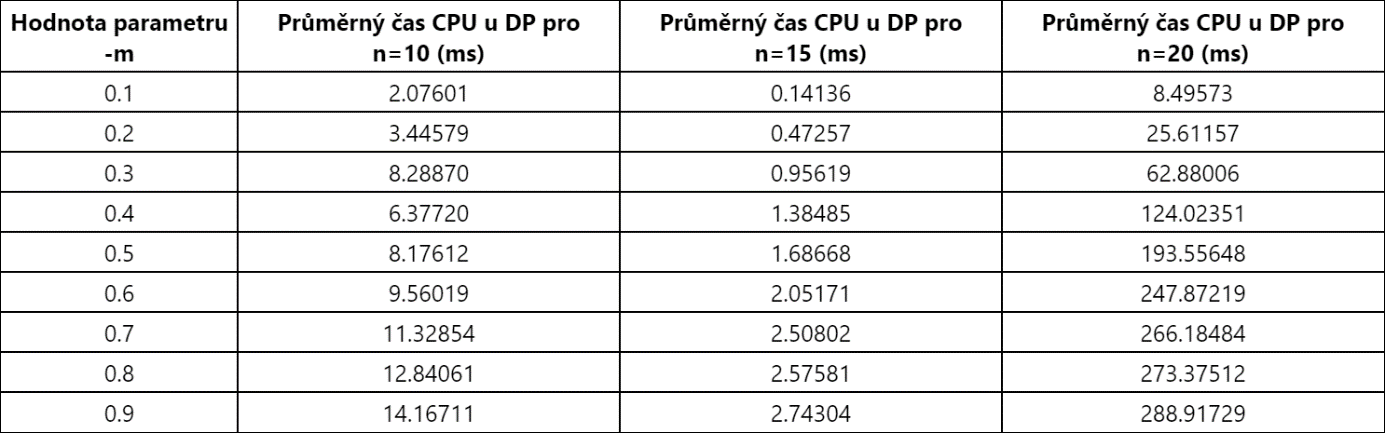
Komentář autora experimentu: V případě vynechání podmínky na **max. váhu** mají všechny křivky čistě klesající charakter. Zároveň dochází i ke zhoršení výpočetní náročnosti. Ve všech dalších experimentech se používá B&B **s hmotnostní podmínkou**.

**3**

Graf:



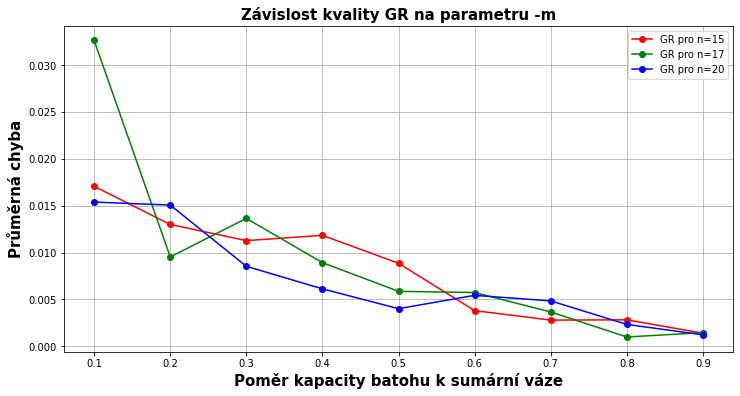
Data:



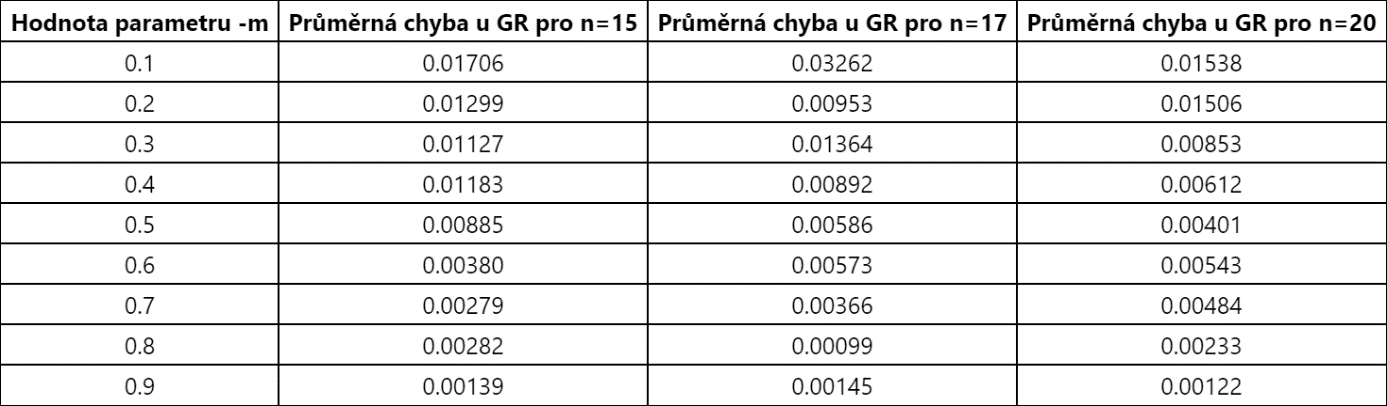
Komentář autora experimentu: S rostoucí hodnotou poměru lze předpokládat zvětšování nosnosti batohu. Vzhledem k tomu, že DP pracuje s 2D polem mezivýsledků, jehož rozměry závisí na počtu věcí a **nosnosti**, lze předpokládat narůstání výpočetní náročnosti.

**4**

Graf:



Data:



Komentář autora experimentu: Ze samotného principu, jak GR pracuje (tj. seřazení věcí od nejhodnotnějších a postupné přidávání těchto věcí, dokud se vejdou batohu), lze usoudit, že s rostoucím poměrem klesá chybovost, poněvadž se do batohu povede vměstnat postupně více a více věcí.

**[KORELACE VÁHY S CENOU]**

Pilotní experimenty odhalily pouze závislost **kvality** GR na tomto parametru. U algoritmů BF, B&B, DP (dle váhy) ani GR nebyla zjištěna žádná časová závislost. Pro experimenty v této sekci bylo použito následující nastavení generátoru:

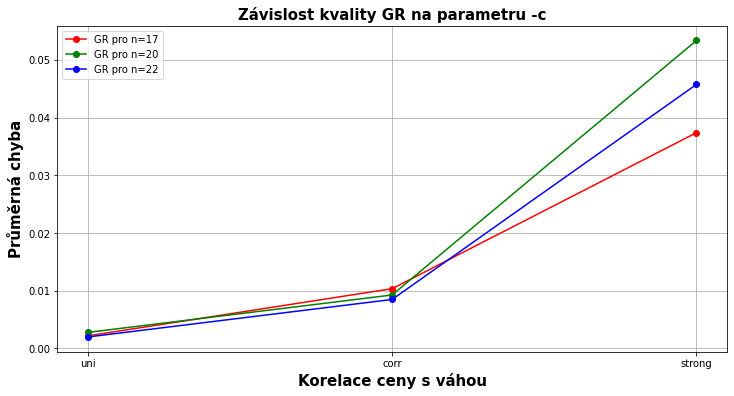
./kg2 -n 17 -N 100 -W 100 -C 1000 -c uni/corr/strong

./kg2 -n 20 -N 100 -W 500 -C 500 -c uni/corr/strong

./kg2 -n 22 -N 100 -W 1000 -C 100 -c uni/corr/strong

**1**

Graf:

****

Data:

****

Komentář autora experimentu: Vzhledem k tomu, že GR při své činnosti seřadí věci podle poměru cena/váha, lze s vyšší korelací těchto dvou parametrů očekávat, že toto seřazení pravděpodobně proběhne špatně, což implikuje vyšší chybovost.

**[GRANULARITA]**

Pilotní experimenty odhalily závislost **výpočetní náročnosti** algoritmu DP (dle váhy) a **kvality** GR na tomto parametru. Naopak u algoritmů BF, B&B a GR nebyla zjištěna žádná podobná časová závislost. Pro experimenty v této sekci bylo použito následující nastavení generátoru:

./kg2 -n 10 -N 100 -W 100 -C 1000 -w light -k 1.0–9.0

./kg2 -n 12 -N 100 -W 500 -C 500 -w light -k 1.0–9.0

./kg2 -n 15 -N 100 -W 1000 -C 100 -w light -k 1.0–9.0

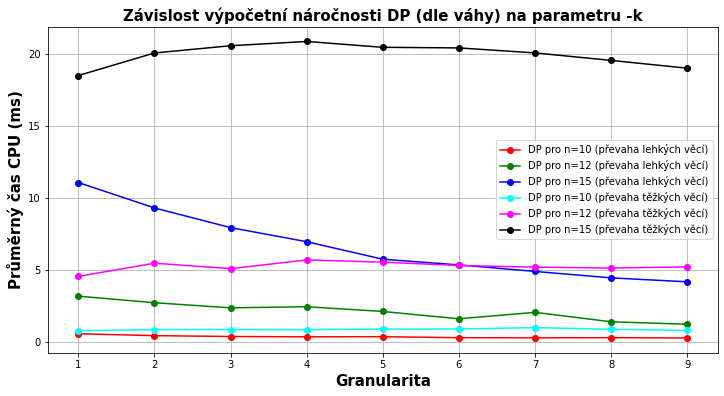
./kg2 -n 10 -N 100 -W 100 -C 1000 -w heavy -k 1.0–9.0

./kg2 -n 12 -N 100 -W 500 -C 500 -w heavy -k 1.0–9.0

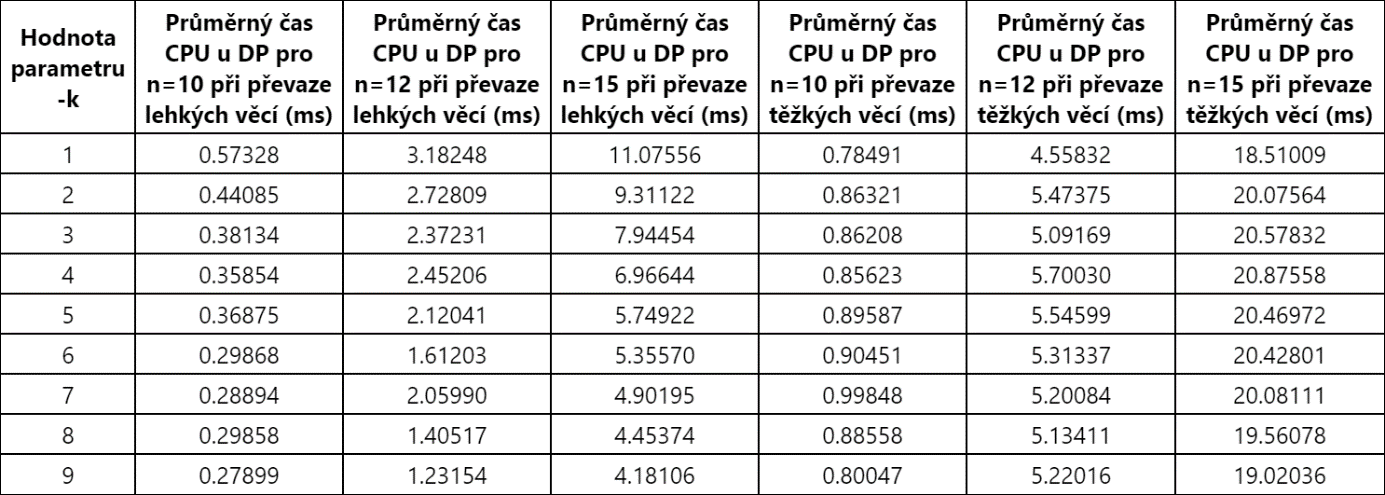
./kg2 -n 15 -N 100 -W 1000 -C 100 -w heavy -k 1.0–9.0

**1**

Graf:

****

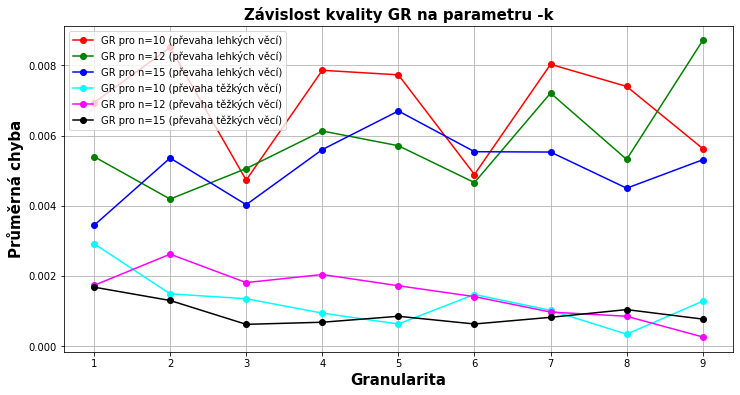
Data:

****

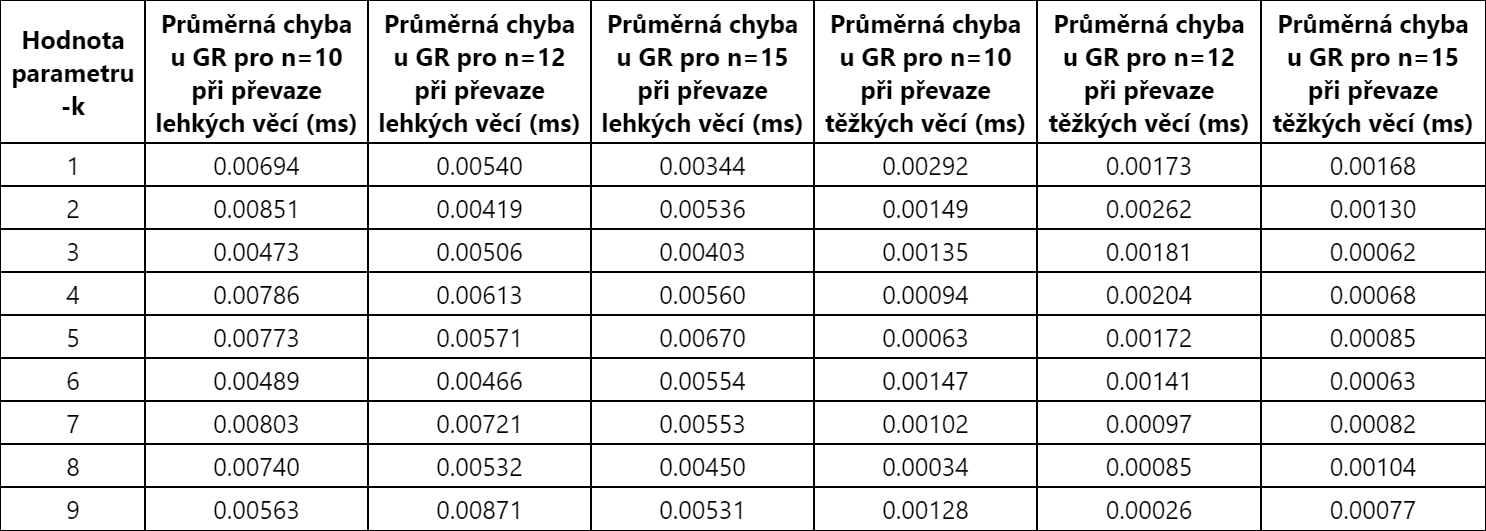
Komentář autora experimentu: Zatímco DP (dle váhy) má při převaze těžších věcí víceméně konstantní charakter, DP (dle váhy) při převaze lehčích věcí s rostoucí hodnotou exponentu granularity postupně klesá. To může být pravděpodobně vysvětleno větším využíváním mezivýsledků z 2D pole, poněvadž pro rostoucí hodnotu exponentu granularity lze předpokládat zmenšování průměru vah všech věcí.

**2**

Graf:

****

Data:

****

Komentář autora experimentu: DP (dle váhy) při převaze lehkých věcí také způsobuje mírný růst průměrné chyby algoritmu GR, zatímco DP (dle váhy) při převaze těžších věcí naopak způsobuje nepatrný pokles této chyby.

**[MAXIMÁLNÍ VÁHA]**

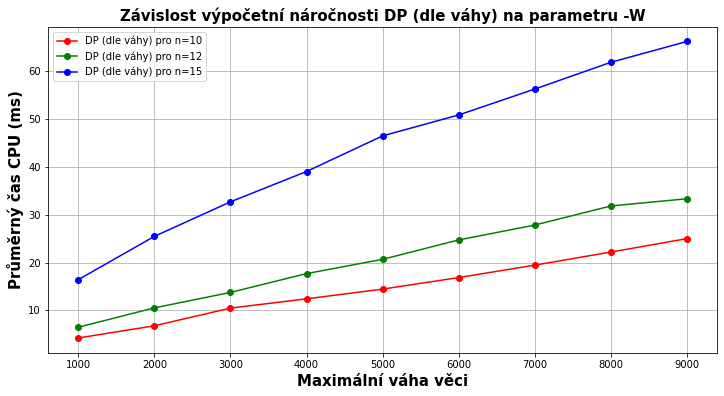
Pilotní experimenty ověřily podezření na závislost **výpočetní náročnosti** algoritmu DP (dle váhy) na parametru maximální váhy, příp. ceny. V případě maximální váhy byla objevena časová závislost, v případě maximální ceny nikoliv. Pro experimenty v této sekci bylo použito následující nastavení generátoru:

./kg2 -n 10 -N 50 -W 1000–9000 -C 1000

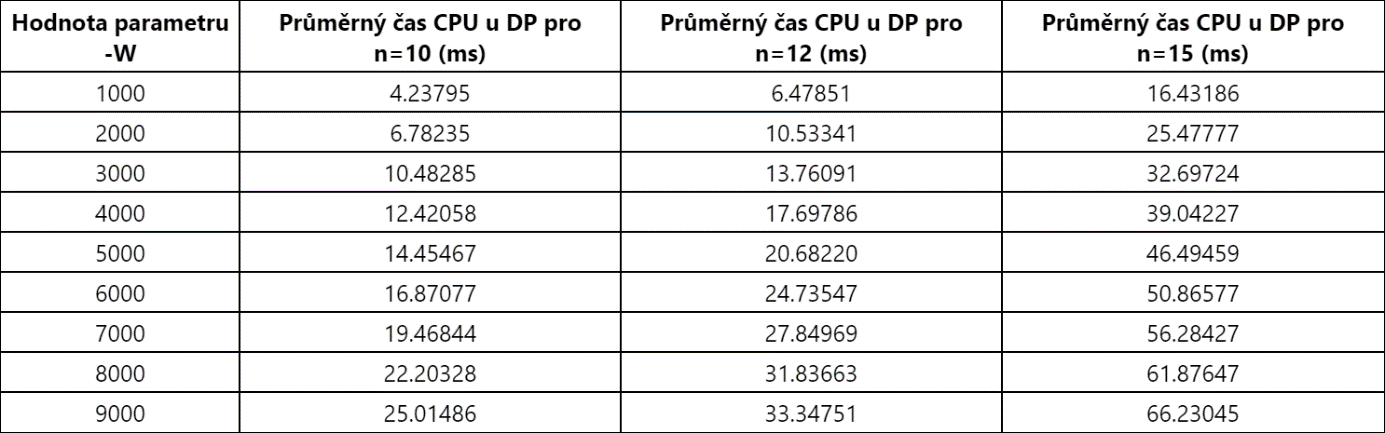
./kg2 -n 12 -N 50 -W 1000–9000 -C 500

./kg2 -n 15 -N 50 -W 1000–9000 -C 100

**1**

Graf:

Data:



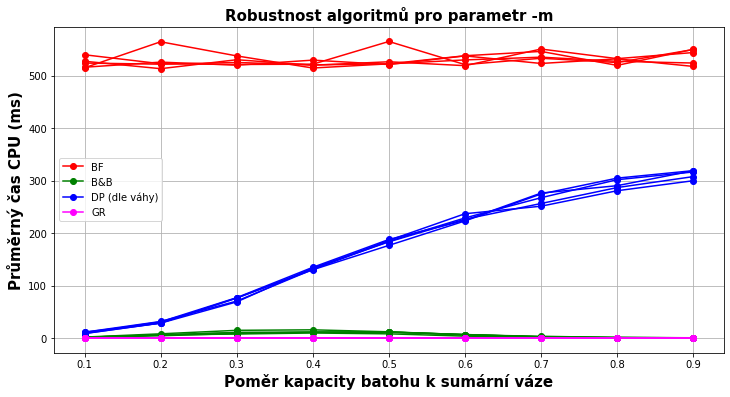
Komentář autora experimentu: Pro rostoucí maximální váhu věci (tzn. také nosnost) lze předpokládat zvětšování 2D pole mezivýsledků (jeho velikost závisí na **nosnosti** a **počtu věcí**), se kterým DP (dle váhy) pracuje. Tím pádem se zvyšuje výpočetní náročnost.

**[ROBUSTNOST]**

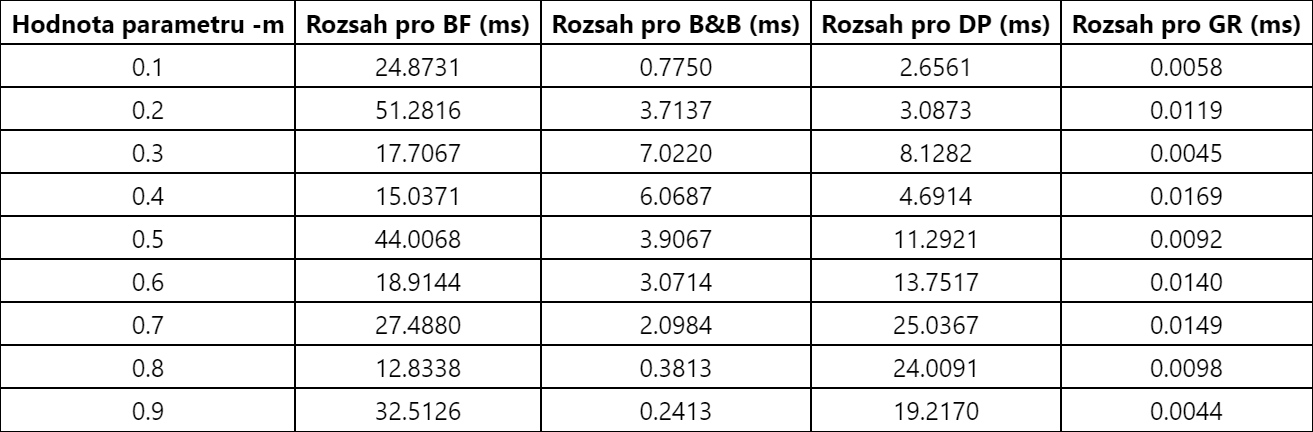
Pro každý testovaný parametr z předchozích experimentů byla ověřena **robustnost** všech algoritmů (tj. nezávislost na pořadí věcí v jednotlivých instancích). Pro algoritmus GR lze zcela jistě tvrdit, že je robustní, protože při něm vždy dochází k **setřídění věcí** podle poměru cena/váha. Naopak na metodě BF tolik **nezáleží**, poněvadž vždy projde celý stavový prostor. Pro každý experiment byla použita jediná instance, která byla celkem **pětkrát** permutovaná.

**1**

Graf:

****

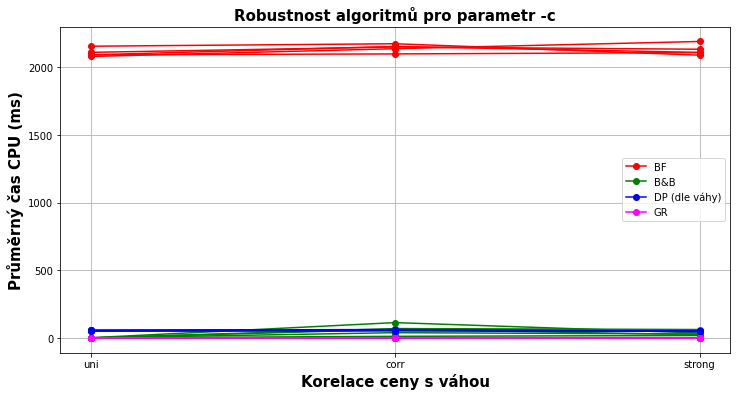
Data:

****

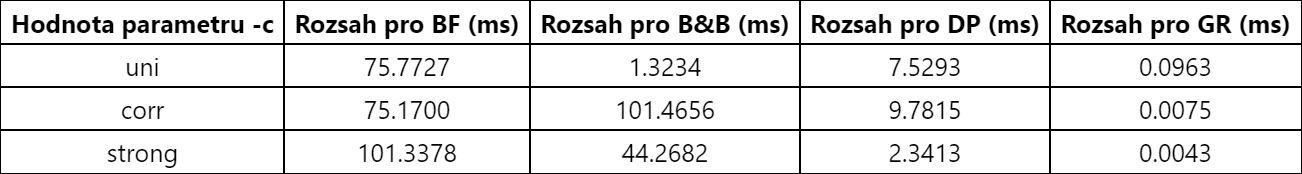
Komentář autora experimentu: U algoritmů B&B a DP (dle váhy) lze sledovat dříve naměřené závislosti. Různé permutace té samé instance se udržují na stejném místě v grafu. Největší časové rozpětí bylo naměřeno u BF a DP (dle váhy). Naopak nejmenší takové rozpětí má GR.

**2**

Graf:



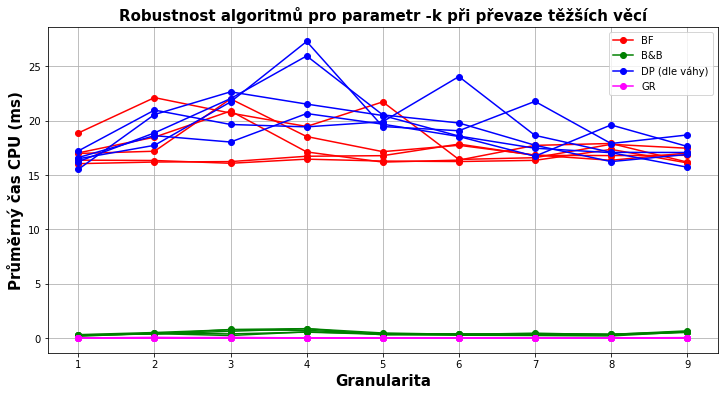
Data:

****

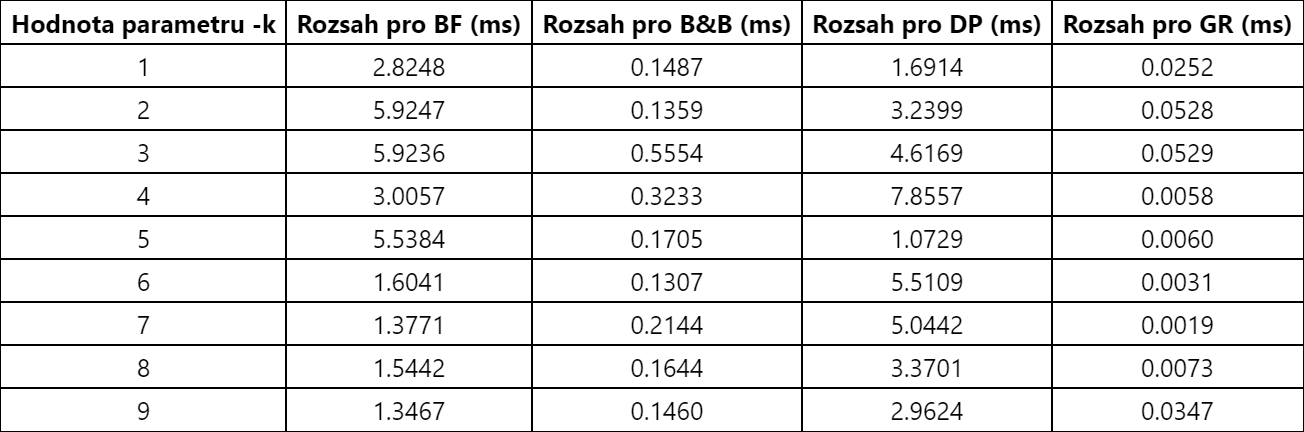
Komentář autora experimentu: Pro tento parametr nebyla naměřena žádná časová závislost. Výjimkou je pouze kvalita algoritmu GR, který je ovšem robustní.

**3**

Graf:

****

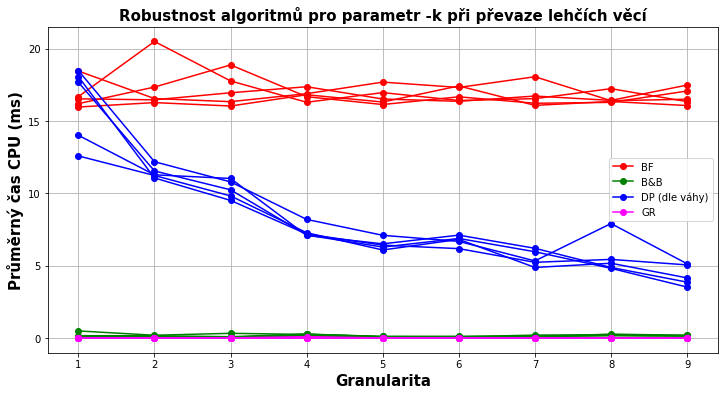
Data:

****

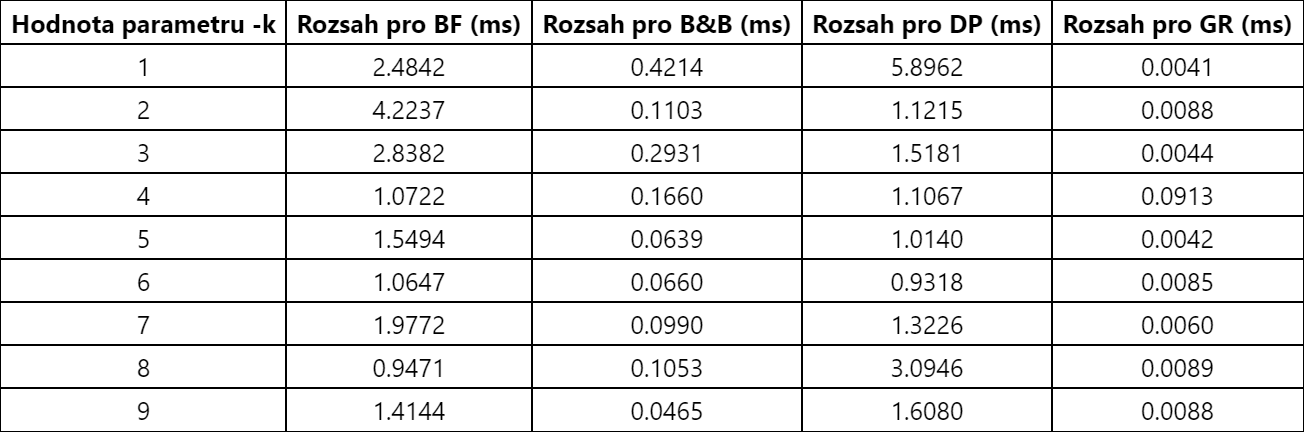
Komentář autora experimentu: Pro tento parametr opět nebyla naměřena žádná časová závislost. Jednotlivé algoritmy se víceméně udržují na stejných hodnotách.

**4**

Graf:

****

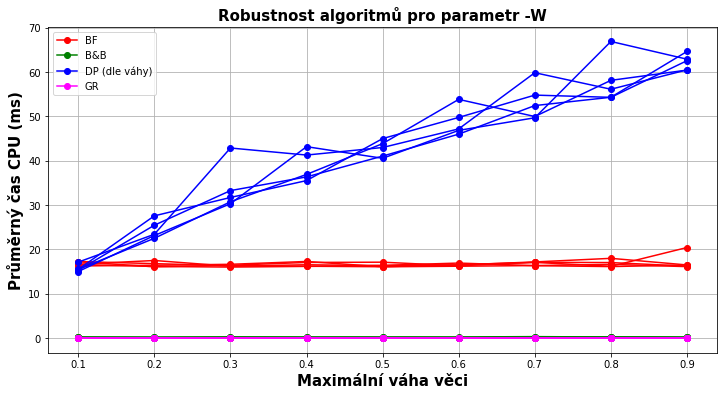
Data:

****

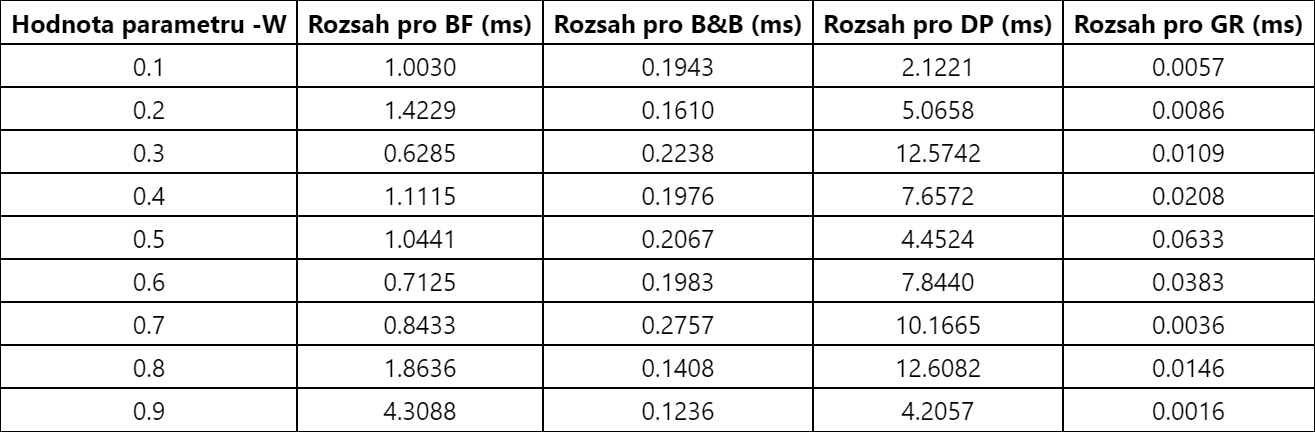
Komentář autora experimentu: U algoritmu DP (dle váhy) lze sledovat dříve naměřenou závislost. Různé permutace té samé instance se opět udržují na stejném místě v grafu. Největší časové rozpětí bylo opět naměřeno u BF a DP (dle váhy).

**5**

Graf:

****

Data:

****

Komentář autora experimentu: U algoritmu DP (dle váhy) lze sledovat dříve naměřenou závislost. Různé permutace té samé instance se opět udržují na stejném místě v grafu. Největší časové rozpětí bylo tentokrát naměřeno u DP (dle váhy).

**ZÁVĚR**

V předchozích experimentech byly otestovány celkem čtyři hodnoty generátoru instancí – **poměr kapacity batohu k sumární váze**, **korelace ceny s váhou**, **granularita** a **maximální váha věci**. Ve všech případech byla nalezena alespoň jedna časová/kvalitativní závislost. Dále byla ověřena **robustnost** algoritmů. Různé permutace té samé instance se při výpočtu vždy udržovaly na stejných nebo přibližných hodnotách. Nejmenší časový rozsah byl naměřen u algoritmu GR, který je nejvíce robustní. Naopak největší rozsah byl naměřen u algoritmu BF, u kterého ovšem na robustnosti tolik nezáleží, poněvadž vždy projde všechny konfigurace.