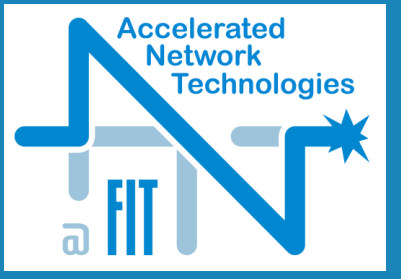
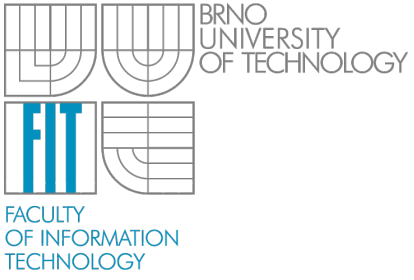
Perfektní hašovací funkce



Jan Kaštil

Brno University of Technology, Faculty of Information Technology
Božetěchova 2, 612 00 Brno, CZ
www.fit.vutbr.cz/~ikastil



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Obsah

Hašovací funkce

Haš funkce je matematická funkce pro převod libovolných vstupních dat do intervalu.

Uniformní Hašovací funkce

Hašovací funkce, pro kterou platí, že pro náhodný vstupní řetězec je pravděpodobnost každého čísla s výstupního intervalu stejná.

2-nezávislá Hašovací funkce

Hašovací funkce je 2-nezávislá, právě když pro dva různé vstupy hašovací funkce x,y a dvě různá čísla s výstupního intervalu a,b platí:

$$Pr[h(x) = a \wedge h(y) = b] = \frac{1}{R^2} \quad (1)$$

Perfektní Hašovací Funkce

Hašovací funkce se nazývá Perfektní, pokud každému klíči z množiny S přiřadí jedinečnou hodnotu s výstupního intervalu.

Minimální Perfektní Hašovací Funkce

Perfektní hašovací funkce se nazývá minimální, pokud platí, že velikost množiny klíčů (S) je rovna velikosti výstupního intervalu.

Statická Perfektní Hašovací Funkce

Perfektní hašovací funkce se nazývá Statická, pokud po celou dobu trvání výpočtu je množina S konstantní.

Dynamická Perfektní Hašovací Funkce

Perfektní hašovací funkce se nazývá dynamická, pokud je v průběhu výpočtu možno přidávat a odebírat klíče z množiny S.

Algoritmy konstrukce PHF: FCH

Základní vlastnosti

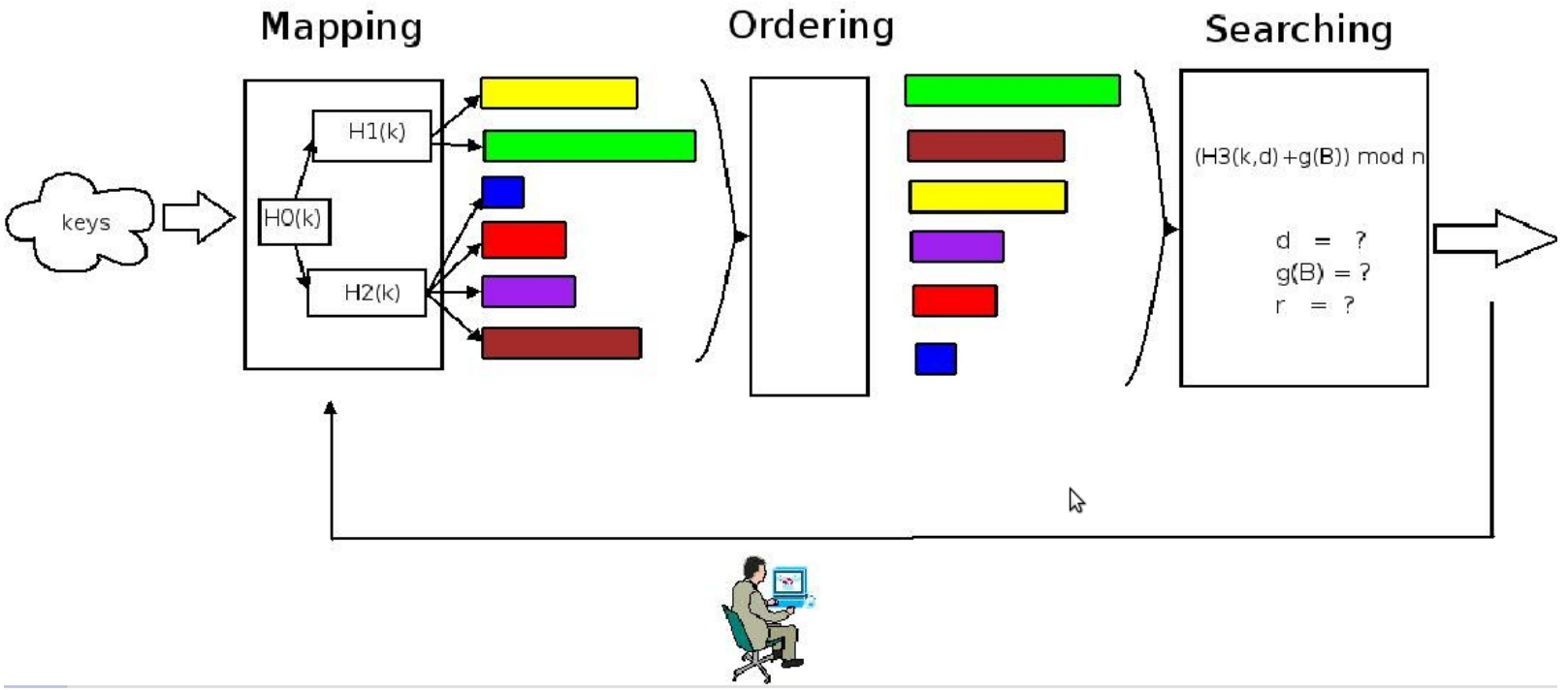
- Lineární paměťová složitost – 2,6 bitů na klíč
- Exponenciální časová složitost v nejhorším případě
- Lineární časová složitost v běžných případech

Skládání haš funkcí

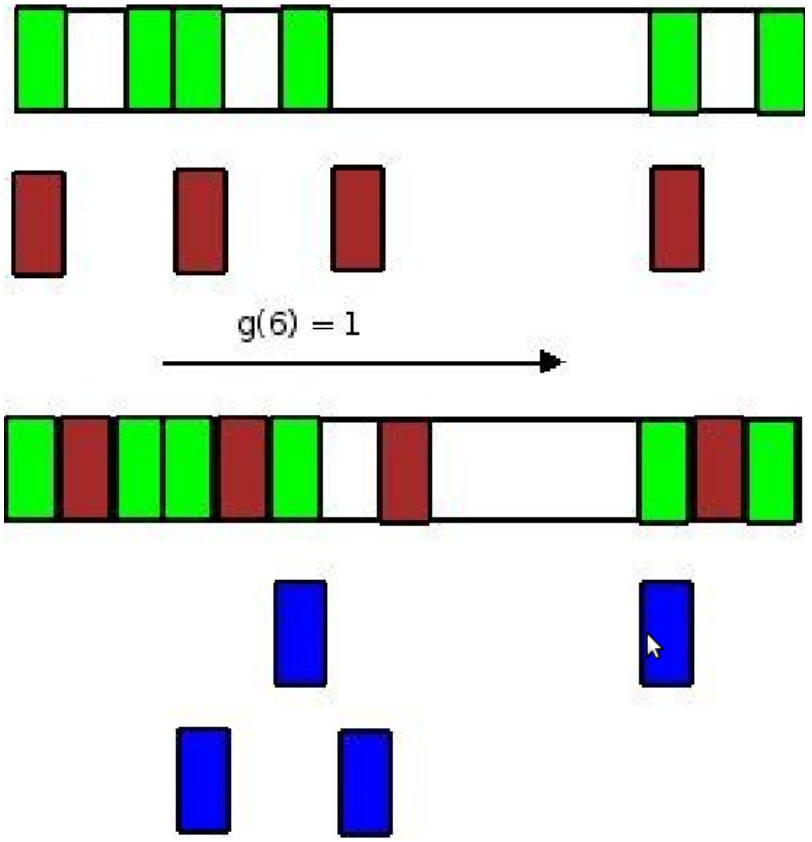
- První haš zobrazí klíče do číselného intervalu
- Další dvě haš funkce zobrazí číselnou reprezentaci klíče na číslo koše
- Poslední parametrizovatelná haš funkce zobrazí číselnou reprezentaci klíče na unikátní hodnotu podle parametrů asociovaných s košem

Konstrukce PHF

- Mapping – Zobrazení klíčů do košů
- Orderring – Seřazení košů podle velikosti
- Searching – Stanovení parametrů pro jednotlivé koše



- Mapping umístí 60% klíčů do 30% košů – zbývající koše jsou malé
- Ordering zajistí, aby se první našli parametry pro největší koš
- Searching je prohledávání stavového prostoru parametrů košů



Searching

- Hledání posunu jednotlivých košů ve výstupním intervalu
- Začínáme od největšího koše
- Pokud neexistuje vhodné posunutí, změníme hodnotu d – modrý koš
- Poslední možností je vygenerování jiných haš funkcí

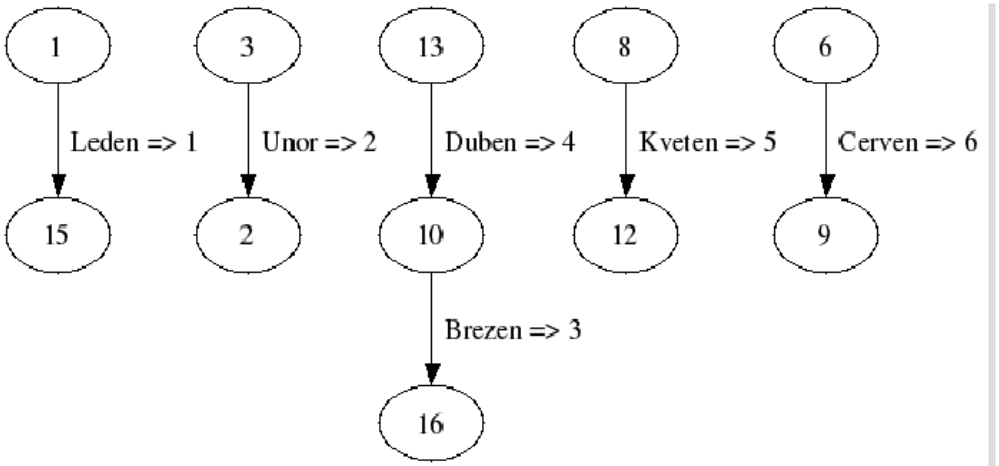
Tvorba náhodného grafu

Náhodný graf

Mějme neprázdnou množinu vrcholů V a prázdnou množinu hran E. Náhodný graf vznikne přidáváním hran do množiny E tak, že každá přidávaná hrana spojuje dva vrcholy, které byly vybrány náhodně.

Tvorba náhodného grafu

- Náhodně zvolíme dvě hashovací funkce
- Na uniformní hashovací funkce je generátor pseudonáhodných čísel s uniformním rozložením
- Hashovací funkce přidají pro každý vstup jednu hranu do grafu
- Pro množinu vstupů S tedy vznikne náhodný graf G, kde každá hrana odpovídá jednomu prvku množiny S
- Náhodně zvolíme dvě hashovací funkce



Vlastnosti grafu

- Graf je acyklický a beze smyček
- Z teorie pravděpodobnosti plyne, že by graf měl mít 2x více vrcholů než hran
- Ohodnocení hran je součet ohodnocení vrcholů

Vylepšení algoritmu CHM

Použití cyklického grafu

- Acykličnost je požadována pro usnadnění ohodnocení vrcholů
- Acykličnost je nutná pokud chceme klíčům přiřadit konkrétní hodnotu, není nutná pro unikátnost
- Cyklický graf vyžaduje méně vrcholů – menší paměťová složitost
- Pokud je cyklická maximálně polovina grafu, lze vrcholy ohodnotit bez nárůstu časové složitosti

Použití acyklického hypergrafu

- Hypergraf asociuje více vrcholů ke každé hraně
- Pravděpodobnost najití acyklického grafu je největší při použití 3 haš funkcí
- Místo hodnoty kliče se ukládá číslo haš funkce, která je pro daný klíč unikátní

A nyní diskuze!