

Quality Measurements for Hash Functions

Andres Valloud

Slide of Contents

1. Why measure?
2. Quick review of hashing.
3. Measurements and their motivations.
4. Hash Analysis Tool demo.
5. Questions.

Why bother... erm, measure?

- Performance performance performance.
- Small changes can lead to huge speedups.
 - Production application example: 17x.

But why is this not usually done?

- No strong theory for quick, good quality hash functions.
 - Testing theory exists, creation theory does not.
 - No available off the shelf quality gauge.
- Hash seen as magic.
- But also... current CS education does not foster an interest in this kind of stuff. Sad.

Quick review of hashing

- Three sets and two integer maps.

Data superset D		Integers X		Integers mod p Y
d_1		x_1		y_1
d_2	\Rightarrow	x_2	\Rightarrow	y_2
d_3	(hash)	x_3	(collection)	y_3
...	
d_n		x_n		y_n

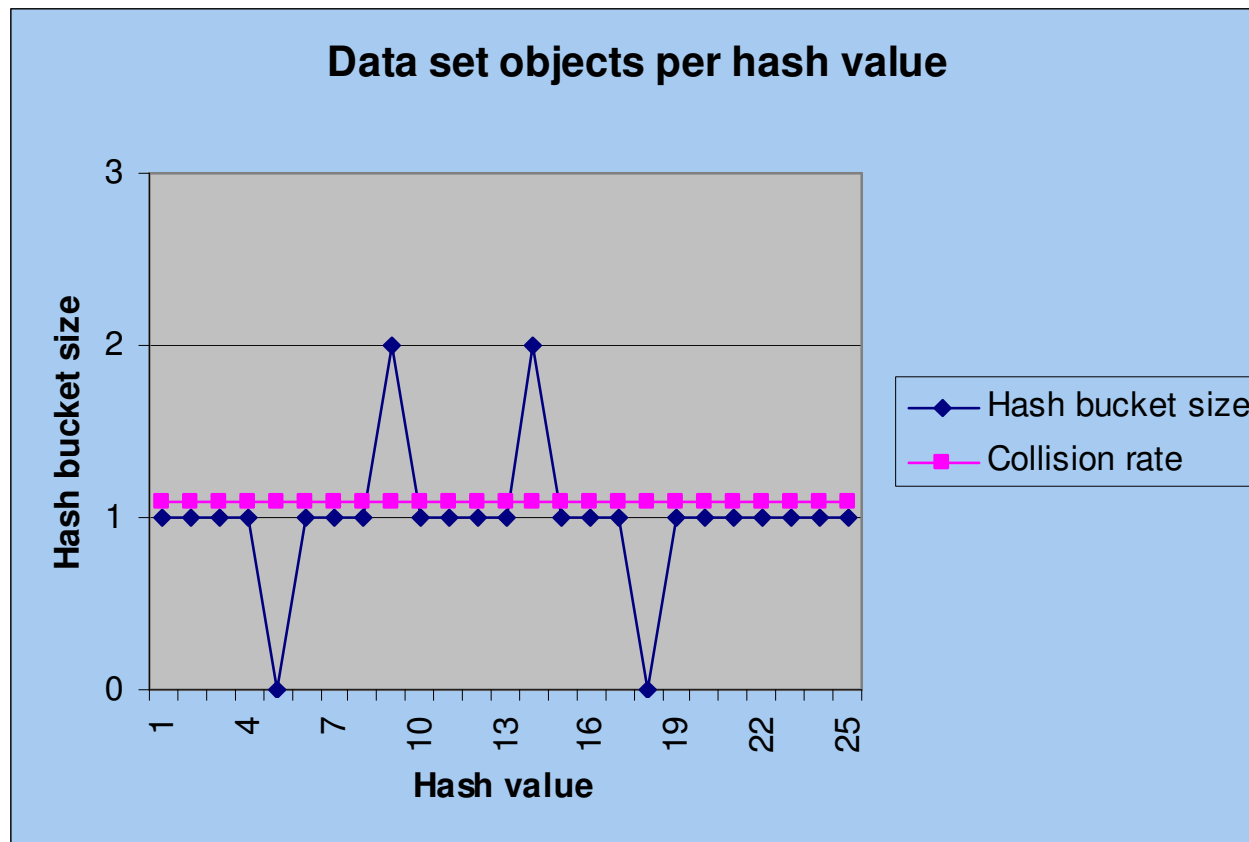
Quick review of hashing

- Sources of collisions:
 - $|D| > |X|$. Inevitable. Leads to “scrambling” properties desired for hash functions.
 - $|X| > |Y|$. Mostly inevitable. Leads to more “scrambling” efforts.
- Too many collisions for your data set / hash function pair, and hash collection access becomes **linear search**.

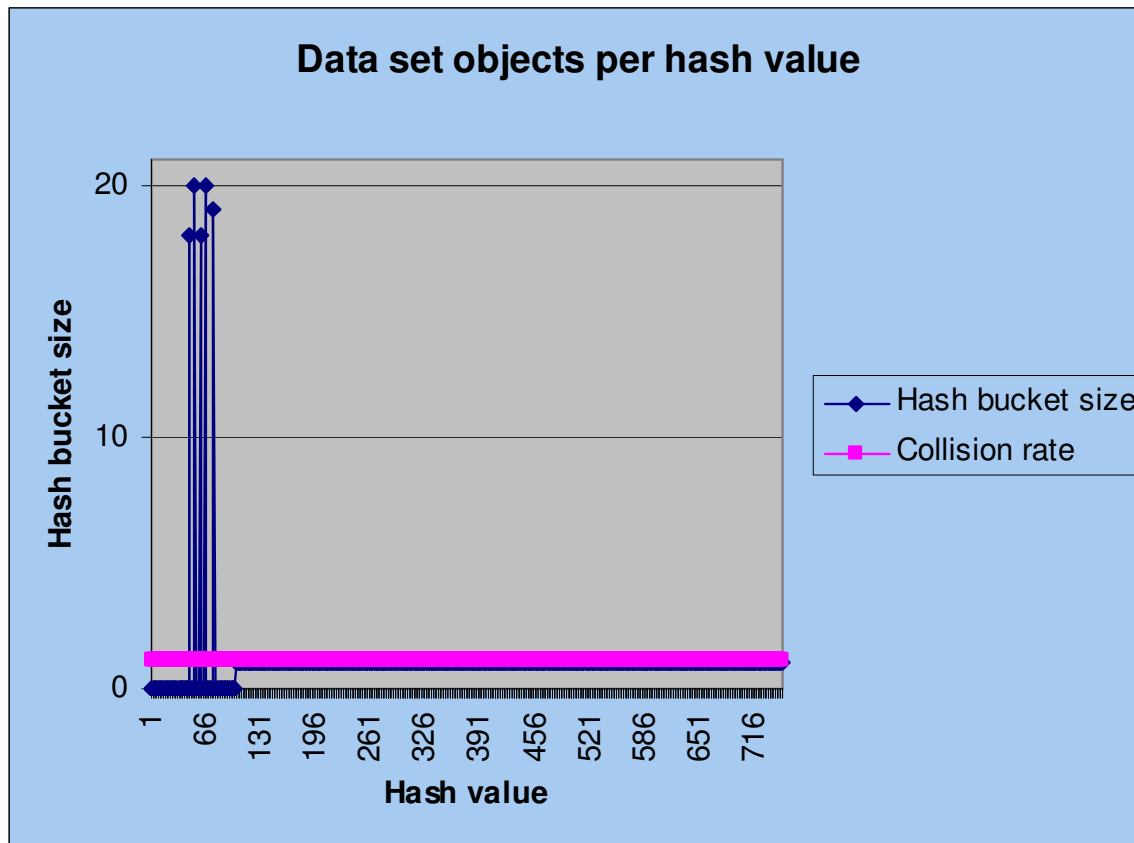
Measurements

- If collisions are bad, then measure average collisions per data set object: *collision rate*.
 - CR = 1: no collisions, good!
 - CR = 2: 1 hash value per 2 data set objects, really bad already!
- Good hash functions: $1 \leq CR \leq 1.01$

Two collisions



But... many collisions...



χ^2 test detects spikes

- CR is an average, and so patterns can hide.
- χ^2 test will catch such problems.
 - $\chi^2(\mathbf{D}) = |\mathbf{D}|^{-1} \sum_{v \in f(\mathbf{D})} |\varphi(v) - 1|^2$
- $\chi^2(\mathbf{D}) = 0$: ideal value.
- $\chi^2(\mathbf{D}) = 1$: implies $2 < \text{CR} < 3$ under best of circumstances.
 - Previous example... $\text{CR} \sim 1.15$, $\chi^2(\mathbf{D}) \sim 1.9$

$\chi^2(D)$ is still not enough

- Integer >> hash implemented as...
 - self * someFactor bitAnd: 16rFFFFFFFF
- CR = 1, $\chi^2(D) = 0$, for small data sets.
- Looks fine, but:
 - all hash values are multiples of some factor!
 - bias against changes in most significant bits!

$$\chi^2(\mathbf{D}, p)$$

- Evaluate collision behavior in hashed collections, and measure spikes there.
 - $\chi^2(\mathbf{D}, p) = |\mathbf{D}|^{-1} \sum_{v \in f(\mathbf{D}, p)} |\varphi(v, p) - 1|^2$
- Test for many $p > |\mathbf{D}|$.
- $\chi^2(\mathbf{D}, p)$ should behave like the load factor or better. No spikes. No large value inversions.

Hash Analysis Tool v2.x

- Demo!

Curious about hash functions?

- Hashing in Smalltalk: Theory and Practice
 - 450 pages, 220+ exercises.
- Hash Analysis Tool bundle in public Store repository.

Questions?

Thanks for coming!

- Useful links:
 - <http://groups.google.com/group/hash-functions>
 - <http://planet.smalltalk.org>
 - Google: cincom smalltalk community resources
 - IRC channel: <irc.parcplace.net>, #smalltalk
 - Brought to you courtesy of Peter Hatch