Tabulation-Based Hashing

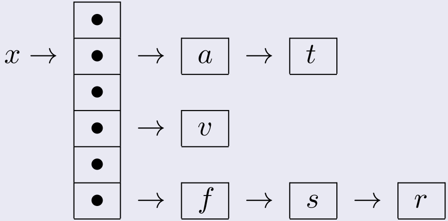
Mihai Pătraşcu   Mikkel Thorup

April 23, 2010

Applications of Hashing

Hash tables:

- chaining



Applications of Hashing

Hash tables:

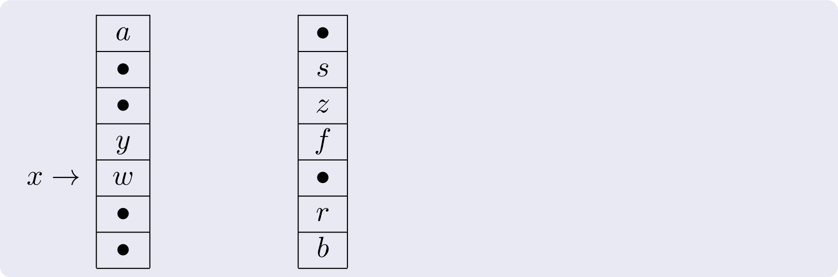
- chaining
- linear probing

$\bullet$	m	$\bullet$	c	a	f	t	$\bullet$	x	y
				$\uparrow$	$\uparrow$	$\uparrow$			
				t					

Applications of Hashing

Hash tables:

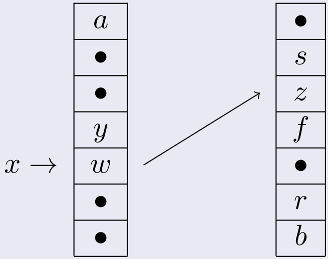
- chaining
- linear probing
- cuckoo hashing



Applications of Hashing

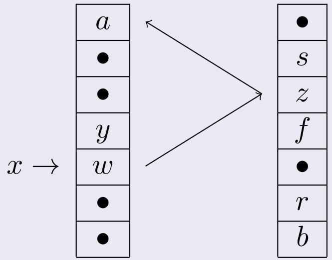
Hash tables:

- chaining
- linear probing
- cuckoo hashing



Hash tables:

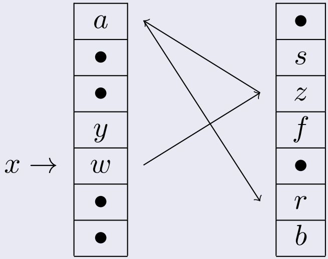
- chaining
- linear probing
- cuckoo hashing



Applications of Hashing

Hash tables:

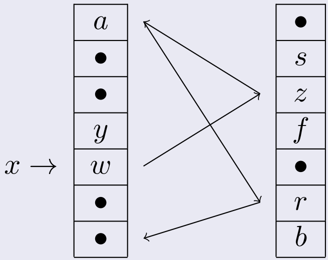
- chaining
- linear probing
- cuckoo hashing



Applications of Hashing

Hash tables:

- chaining
- linear probing
- cuckoo hashing



Applications of Hashing

Hash tables:

- chaining
- linear probing
- cuckoo hashing

Sketching and streaming:

- moment estimation: $F_2(\bar{x}) = \sum_i x_i^2$

Applications of Hashing

Hash tables:

- chaining
- linear probing
- cuckoo hashing

Sketching and streaming:

- moment estimation: $F_2(\bar{x}) = \sum_i x_i^2$
- sketch A and B to later find $\frac{|A \cap B|}{|A \cup B|}$

Applications of Hashing

Hash tables:

- chaining
- linear probing
- cuckoo hashing

Sketching and streaming:

- moment estimation: $F_2(\bar{x}) = \sum_i x_i^2$
- sketch A and B to later find $\frac{|A \cap B|}{|A \cup B|}$
- etc, etc.

Minwise independence

Hash each set through h, keen the minimum

$$\frac{|A \cap B|}{|A \cup B|} = \Pr_h[\min h(A) = \min h(B)]$$

- repeat with k different h;
- keep smallest k items with one h

Minwise independence

Hash each set through h, keen the minimum

$$\frac{|A \cap B|}{|A \cup B|} = \Pr_h[\min h(A) = \min h(B)]$$

- repeat with k different h;
- keep smallest k items with one h

The guarantee we need on h: “minwise independence”

$$(\forall) S, x : \qquad \Pr[x < \min h(S)] = \tfrac{1}{|S|+1}$$

Minwise independence

Hash each set through h, keen the minimum

$$\frac{|A \cap B|}{|A \cup B|} = \Pr_h[\min h(A) = \min h(B)]$$

- repeat with k different h;
- keep smallest k items with one h

The guarantee we need on h: “minwise independence”

$$(\forall) S, x : \qquad \Pr[x < \min h(S)] = \tfrac{1}{|S|+1}$$

Not feasible... Approximate:

$$(\forall) S, x : \qquad \Pr[x < \min h(S)] = \tfrac{1 \pm \varepsilon}{|S|+1}$$

Approximation = $\varepsilon + f\left(\#\ \text{repetitions}\right)$

Minwise independence

Hash each set through h, keen the minimum

$$\frac{|A \cap B|}{|A \cup B|} = \Pr_h[\min h(A) = \min h(B)]$$

- repeat with k different h;
- keep smallest k items with one h

The guarantee we need on h: "minwise independence"

$$(\forall)S, x: \qquad \Pr[x < \min h(S)] = \tfrac{1}{|S|+1}$$

Not feasible... Approximate:

$$(\forall)S, x: \qquad \Pr[x < \min h(S)] = \tfrac{1 \pm \varepsilon}{|S|+1}$$

Approximation = $\varepsilon + f\left(\# \text{ repetitions}\right)$
NB: for weighted A, B the generalization is priority sampling

Carter & Wegman (1977)

A family $\mathcal{H} = \{h : [u] \to [b]\}$ is k-independent iff:

- $(\forall) x \in u$, $h(x)$ is uniform in $[b]$;
- $(\forall) x_1, \ldots, x_k \in [u]$, $h(x_1), \ldots, h(x_k)$ are independent.

Carter & Wegman (1977)

A family $\mathcal{H} = \{h : [u] \to [b]\}$ is k-independent iff:

- $(\forall) x \in u$, $h(x)$ is uniform in $[b]$;
- $(\forall) x_1, \ldots, x_k \in [u]$, $h(x_1), \ldots, h(x_k)$ are independent.

Prototypical example: degree k polynomial

- u prime;
- choose $a_0, a_1, \ldots, a_{k-1}$ randomly in $[u]$;
- $h(x) = \left(a_0 + a_1 x + \cdots + a_{k-1} x^{k-1}\right) \bmod b$.

How much independence?

Chaining	2	
Linear probing	≤ 5 [Pagh^2, Ružić'07]	≥ 5 **[PT'10]**
Cuckoo hashing	$O(\lg n)$	≥ 6 [Cohen, Kane'05]
F_2 estimation	4 [Thorup, Zhang'04]	
ε-minwise indep.	$O(\lg \frac{1}{\varepsilon})$ [Indyk'99]	$\Omega(\lg \frac{1}{\varepsilon})$ **[PT'10]**

How much independence?

Chaining	2	
Linear probing	≤ 5 [Pagh^2, Ružić'07]	≥ 5 **[PT'10]**
Cuckoo hashing	$O(\lg n)$	≥ 6 [Cohen, Kane'05]
F_2 estimation	4 [Thorup, Zhang'04]	
ε-minwise indep.	$O(\lg \frac{1}{\varepsilon})$ [Indyk'99]	$\Omega(\lg \frac{1}{\varepsilon})$ **[PT'10]**

Chaining: $\text{time} = \#\{x \mid h(x) = h(\text{query})\}$

$$\mathbf{E}[\text{time}] = n \cdot \Pr[h(x) = h(\text{query})] = n \cdot \frac{1}{b} = O(1)$$

How much independence?

Chaining	2	
Linear probing	≤ 5 [Pagh^2, Ružić'07]	≥ 5 **[PT'10]**
Cuckoo hashing	$O(\lg n)$	≥ 6 [Cohen, Kane'05]
F_2 estimation	4 [Thorup, Zhang'04]	
ε-minwise indep.	$O(\lg \frac{1}{\varepsilon})$ [Indyk'99]	$\Omega(\lg \frac{1}{\varepsilon})$ **[PT'10]**

Chaining: $\mathrm{time} = \#\{x \mid h(x) = h(\mathrm{query})\}$

$$\mathbf{E}[\mathrm{time}] = n \cdot \Pr[h(x) = h(\mathrm{query})] = n \cdot \frac{1}{b} = O(1)$$

Cuckoo hashing:

components in random graphs have size $O(\lg n)$

How much independence?

Chaining	2	
Linear probing	≤ 5 [Pagh^2, Ružić'07]	≥ 5 **[PT'10]**
Cuckoo hashing	$O(\lg n)$	≥ 6 [Cohen, Kane'05]
F_2 estimation	4 [Thorup, Zhang'04]	
ε-minwise indep.	$O(\lg \frac{1}{\varepsilon})$ [Indyk'99]	$\Omega(\lg \frac{1}{\varepsilon})$ **[PT'10]**

Chaining: $\text{time} = \#\{x \mid h(x) = h(\text{query})\}$

$$\mathbf{E}[\text{time}] = n \cdot \Pr[h(x) = h(\text{query})] = n \cdot \frac{1}{b} = O(1)$$

Cuckoo hashing:

components in random graphs have size $O(\lg n)$

Minwise independence:

k-level inclusion/exclusion estimates probabilities to $\pm 2^{-k}$.

Linear probing

Implementing k-independence

Goals:

- constant time for $\omega(1)$ independence
- practical solution?

Implementing k-independence

Goals:

- constant time for $\omega(1)$ independence
- practical solution?

Lower bound [Siegel'90s]:
With space $u^{1/q}$, query time $\geq \min\{k, q\}$.

Implementing k-independence

Goals:

- constant time for $\omega(1)$ independence
- practical solution?

Lower bound [Siegel'90s]:
With space $u^{1/q}$, query time $\geq \min\{k, q\}$.

Tabulation hashing:

- q basic characters: $x \mapsto (x_1, \ldots, x_q)$
- d derived characters: $y_i = f_i(x_1, \ldots, x_q)$
- store $q + d$ random tables $T_i\left[u^{1/q}\right]$
- $h(x) = T_1[q_1] \oplus \cdots \oplus T_q[x_q] \oplus T_{q+1}[y_1] \oplus \cdots$

Tabulation-Based Hashing

	Independence	# characters
[Carter, Wegman'77]	3	$q \quad (\star)$
[Siegel'90s]	$n^{\Omega(1)}$	$q^{O(q)}$
[Dietzf., Woelfel'03]	k	$k \cdot q$
[Thorup, Zhang'04]	k	$(k-1)(q-1)$
[Thorup, Zhang'10]	5	$2q-1$
recent	$\omega(1)$	$O(q^2)$

$(\star)$ simple tabulation (no derived characters)

Peeling $(q = 2, k = 3)$

$(x_1, x_2) \mapsto T_1[x_1] \oplus T_2[x_2]$

Let's prove independence of $\{a, b, c\}$.

Peeling $(q = 2, k = 3)$

$$(x_1, x_2) \mapsto T_1[x_1] \oplus T_2[x_2]$$

Let's prove independence of $\{a, b, c\}$.

a_1	a_2
b_1	b_2
c_1	c_2

Peeling:
If a_i is unique $(a_i \neq b_i, c_i)$
$\implies h(a)$ independent of $h(b), h(c)$

Peeling $(q = 2, k = 3)$

$(x_1, x_2) \mapsto T_1[x_1] \oplus T_2[x_2]$

Let's prove independence of $\{a, b, c\}$.

a_1	a_2
b_1	b_2
c_1	c_2

Peeling:
If a_i is unique ($a_i \neq b_i, c_i$)
$\implies h(a)$ independent of $h(b), h(c)$

Any set of ≤ 3 keys is peelable, thus independent.

Peeling $(q = 2, k = 4)$

$$(x_1, x_2) \mapsto T_1[x_1] \oplus T_2[x_2] \oplus T_3[x_1 + x_2]$$

Let's prove $\{a, b, c, d\}$ are independent.

- if we can peel, reduce to 3-independence.
- the only non-peelable configuration:

Peeling $(q = 2, k = 4)$

$$(x_1, x_2) \mapsto T_1[x_1] \oplus T_2[x_2] \oplus T_3[x_1 + x_2]$$

Let's prove $\{a, b, c, d\}$ are independent.

- if we can peel, reduce to 3-independence.
- the only non-peelable configuration:

x	s	$x+s$
x	t	$x+t$
y	s	$y+s$
y	t	$y+t$

Peeling $(q = 2, k = 4)$

$$(x_1, x_2) \mapsto T_1[x_1] \oplus T_2[x_2] \oplus T_3[x_1 + x_2]$$

Let's prove $\{a, b, c, d\}$ are independent.

- if we can peel, reduce to 3-independence.
- the only non-peelable configuration:

x	s	$x + s$
x	t	$x + t$
y	s	$y + s$
y	t	$y + t$

Only possible equalities: $x + s = y + t$ or $x + t = y + s$.
Both cannot hold, so we have peeling in derived character.

5-independence [PT'10]

Theorem: Any 4-independent tabulation is 5-independent!
Among any 5 keys, one is independent in the basic characters.

5-independence [PT'10]

Theorem: Any 4-independent tabulation is 5-independent!
Among any 5 keys, one is independent in the basic characters.

- any unique character $\Rightarrow$ peel

5-independence [PT'10]

Theorem: Any 4-independent tabulation is 5-independent!
Among any 5 keys, one is independent in the basic characters.

- any unique character $\Rightarrow$ peel
- otherwise, any dimension looks like: three "0", two "1"

5-independence [PT'10]

Theorem: Any 4-independent tabulation is 5-independent!
Among any 5 keys, one is independent in the basic characters.

- any unique character $\Rightarrow$ peel
- otherwise, any dimension looks like: three "0", two "1"
- two columns have Hamming distance = 4

$a \mapsto$	0	1	...
$b \mapsto$	0	1	...
$c \mapsto$	1	0	...
$d \mapsto$	1	0	...
$e \mapsto$	1	1	...

5-independence [PT'10]

Theorem: Any 4-independent tabulation is 5-independent!
Among any 5 keys, one is independent in the basic characters.

- any unique character $\Rightarrow$ peel
- otherwise, any dimension looks like: three "0", two "1"
- two columns have Hamming distance = 4

$a \mapsto$	0	1	...
$b \mapsto$	0	1	...
$c \mapsto$	1	0	...
$d \mapsto$	1	0	...
$e \mapsto$	1	1	...

- all columns at Hamming distance = 2

$a \mapsto$	0	0	0	...
$b \mapsto$	0	1	1	...
$c \mapsto$	1	0	1	...
$d \mapsto$	1	1	0	...
$e \mapsto$	1	1	1	...

NB: $h(a) = h(b) \oplus h(c) \oplus h(d)$
If e independent of b, c, d, also independent of $f(b, c, d)$.

Putting it together

Algorithm	Indep.
chaining	2
F_2 estimation	4
linear probing	5
ε-minwise	$\Theta(\lg \frac{1}{\varepsilon})$
cuckoo hashing	$O(\lg n)$?

Scheme	Indep.	Characters
simple tabulation	3	q
[Thorup, Zhang'04]	4	$2q-1$
"$4 \to 5$"	5	$2q-1$
[Thorup, Zhang'04]	k	$(k-1)(q-1)$
[Siegel'90s]	$n^{\Omega(1)}$	$q^{O(q)}$

Putting it together

Algorithm	Indep.
chaining	2
F_2 estimation	4
linear probing	5
ε-minwise	$\Theta(\lg \frac{1}{\varepsilon})$
cuckoo hashing	$O(\lg n)$?

Scheme	Indep.	Characters
simple tabulation	3	q
[Thorup, Zhang'04]	4	$2q-1$
"$4 \to 5$"	5	$2q-1$
[Thorup, Zhang'04]	k	$(k-1)(q-1)$
[Siegel'90s]	$n^{\Omega(1)}$	$q^{O(q)}$

What exactly are we doing here?

The Power of Simple Tabulation

Algorithm	Indep.
chaining	2
F_2 estimation	4
linear probing	5
ε-minwise	$\Theta(\lg \frac{1}{\varepsilon})$
cuckoo hashing	$O(\lg n)$?

Scheme	Indep.	Characters
simple tabulation	3	q
simple tabulation	3	q
simple tabulation	3	q
simple tabulation	3	q
maybe...		

The Power of Simple Tabulation

Algorithm	Indep.
chaining	2
F_2 estimation	4
linear probing	5
ε-minwise	$\Theta(\lg \frac{1}{\varepsilon})$
cuckoo hashing	$O(\lg n)$?

Scheme	Indep.	Characters
simple tabulation	3	q
simple tabulation	3	q
simple tabulation	3	q
simple tabulation	3	q
maybe...		

Simple tabulation:

The Power of Simple Tabulation

Algorithm	Indep.
chaining	2
F_2 estimation	4
linear probing	5
ε-minwise	$\Theta(\lg \frac{1}{\varepsilon})$
cuckoo hashing	$O(\lg n)$?

Scheme	Indep.	Characters
simple tabulation	3	q
simple tabulation	3	q
simple tabulation	3	q
simple tabulation	3	q
maybe...		

Simple tabulation:

The Power of Simple Tabulation

Algorithm	Indep.
chaining	2
F_2 estimation	4
linear probing	5
ε-minwise	$\Theta(\lg \frac{1}{\varepsilon})$
cuckoo hashing	$O(\lg n)$?

Scheme	Indep.	Characters
simple tabulation	3	q
simple tabulation	3	q
simple tabulation	3	q
simple tabulation	3	q
maybe...		

Simple tabulation:

- preserves 4^{th} moment bound $\Rightarrow$ F_2 estimation

The Power of Simple Tabulation

Algorithm	Indep.
chaining	2
F_2 estimation	4
linear probing	5
ε-minwise	$\Theta(\lg \frac{1}{\varepsilon})$
cuckoo hashing	$O(\lg n)$?

Scheme	Indep.	Characters
simple tabulation	3	q
simple tabulation	3	q
simple tabulation	3	q
simple tabulation	3	q
maybe...		

Simple tabulation:

- preserves 4$^{\text{th}}$ moment bound $\Rightarrow$ F_2 estimation
- 1-in-5 indep. $\Rightarrow$ linear probing in expected $O(1)$ time

The Power of Simple Tabulation

Algorithm	Indep.
chaining	2
F_2 estimation	4
linear probing	5
ε-minwise	$\Theta(\lg \frac{1}{\varepsilon})$
cuckoo hashing	$O(\lg n)$?

Scheme	Indep.	Characters
simple tabulation	3	q
simple tabulation	3	q
simple tabulation	3	q
simple tabulation	3	q
maybe...		

Simple tabulation:

- preserves 4th moment bound $\Rightarrow$ F_2 estimation
- 1-in-5 indep. $\Rightarrow$ linear probing in expected $O(1)$ time
- minwise independence with $\varepsilon = \varepsilon(n) = o(1)$.

The Power of Simple Tabulation

Algorithm	Indep.
chaining	2
F_2 estimation	4
linear probing	5
ε-minwise	$\Theta(\lg \frac{1}{\varepsilon})$
cuckoo hashing	$O(\lg n)$?

Scheme	Indep.	Characters
simple tabulation	3	q
simple tabulation	3	q
simple tabulation	3	q
simple tabulation	3	q
maybe...		

Simple tabulation:

- preserves 4th moment bound $\Rightarrow$ F_2 estimation
- 1-in-5 indep. $\Rightarrow$ linear probing in expected $O(1)$ time
- minwise independence with $\varepsilon = \varepsilon(n) = o(1)$.
- Chernoff concentration $\Rightarrow$ $O(\lg n)$ query time w.h.p.

The Power of Simple Tabulation

Algorithm	Indep.
chaining	2
F_2 estimation	4
linear probing	5
ε-minwise	$\Theta(\lg \frac{1}{\varepsilon})$
cuckoo hashing	$O(\lg n)$?

Scheme	Indep.	Characters
simple tabulation	3	q
simple tabulation	3	q
simple tabulation	3	q
simple tabulation	3	q
maybe...		

Simple tabulation:

- preserves 4th moment bound $\Rightarrow$ F_2 estimation
- 1-in-5 indep. $\Rightarrow$ linear probing in expected $O(1)$ time
- minwise independence with $\varepsilon = \varepsilon(n) = o(1)$.
- Chernoff concentration $\Rightarrow$ $O(\lg n)$ query time w.h.p.
- preserve moments in linear probing, chaining: F_p w/ simple tabulation $=$ F_p w/ truly random $+\, o(1)$

Simple tabulation as a PRG

Pseudorandom numbers $\approx h(0), h(1), h(1), \ldots$

	$f(0)$	$f(1)$	$\ldots$	$f(S-1)$
$g(0)$	$h(0)$	$h(1)$	$\ldots$	$h(S-1)$
$g(1)$	$h(S)$	$h(S+1)$	$\ldots$	$h(2S-1)$
$g(2)$	$h(2S)$	$h(2S+1)$	$\ldots$	$h(3S-1)$
$\ldots$				

Simple tabulation as a PRG

Pseudorandom numbers $\approx h(0), h(1), h(1), \ldots$

	$f(0)$	$f(1)$	$\ldots$	$f(S-1)$
$g(0)$	$h(0)$	$h(1)$	$\ldots$	$h(S-1)$
$g(1)$	$h(S)$	$h(S+1)$	$\ldots$	$h(2S-1)$
$g(2)$	$h(2S)$	$h(2S+1)$	$\ldots$	$h(3S-1)$
$\ldots$				

Use $S = O(\lg n)$ truly random numbers.
Compute $\lg n$ independent $g(\cdot)$, but rarely.

PRG has:

- concentration (load balancing, ...)
- minwise independence (treaps, ...)

Wait, there's more!

What more can we ask for?

Wait, there's more!

What more can we ask for?

- minwise independence with $\varepsilon = \varepsilon(n) = o(1)$
  $\longrightarrow$ minwise independence with $\varepsilon = \varepsilon(u) = o(1)$

Wait, there's more!

What more can we ask for?

- minwise independence with $\varepsilon = \varepsilon(n) = o(1)$
  $\longrightarrow$ minwise independence with $\varepsilon = \varepsilon(u) = o(1)$

- linear probing/chaining $O(1)$ exp. time, $O(\lg n)$ w.h.p.
  $\longrightarrow$ for $k \geq \lg n$, any k operations work in $O(k)$ time w.h.p.

Wait, there's more!

What more can we ask for?

- minwise independence with $\varepsilon = \varepsilon(n) = o(1)$
  $\longrightarrow$ minwise independence with $\varepsilon = \varepsilon(u) = o(1)$
  Experiments: $\{0,1\}^q$ is counterexample?
- linear probing/chaining $O(1)$ exp. time, $O(\lg n)$ w.h.p.
  $\longrightarrow$ for $k \geq \lg n$, any k operations work in $O(k)$ time w.h.p.

Wait, there's more!

What more can we ask for?

- minwise independence with $\varepsilon = \varepsilon(n) = o(1)$
  $\longrightarrow$ minwise independence with $\varepsilon = \varepsilon(u) = o(1)$
  Experiments: $\{0, 1\}^q$ is counterexample?
- linear probing/chaining $O(1)$ exp. time, $O(\lg n)$ w.h.p.
  $\longrightarrow$ for $k \geq \lg n$, any k operations work in $O(k)$ time w.h.p.
  We only get $k = n^\varepsilon$ for any $\varepsilon > 0$.
  Counterexample for $n^{o(1)}$.

Wait, there's more!

What more can we ask for?

- minwise independence with $\varepsilon = \varepsilon(n) = o(1)$
  $\longrightarrow$ minwise independence with $\varepsilon = \varepsilon(u) = o(1)$
  Experiments: $\{0,1\}^q$ is counterexample?
- linear probing/chaining $O(1)$ exp. time, $O(\lg n)$ w.h.p.
  $\longrightarrow$ for $k \geq \lg n$, any k operations work in $O(k)$ time w.h.p.
  We only get $k = n^\varepsilon$ for any $\varepsilon > 0$.
  Counterexample for $n^{o(1)}$.

Simple++:

- $h_1 : [u] \to [b]$, $h_2 : [u] \to [u^{1/q}]$. Just simple tabulation...
- $h(x) = h_1(x) \oplus T[h_2(x)]$.

All previous properties, plus:

- minwise independence with $\varepsilon(u) = o(1)$.
- linear probing/chaining with buffer $k = O(\lg n)$.

$\mathcal{THE\ END}$