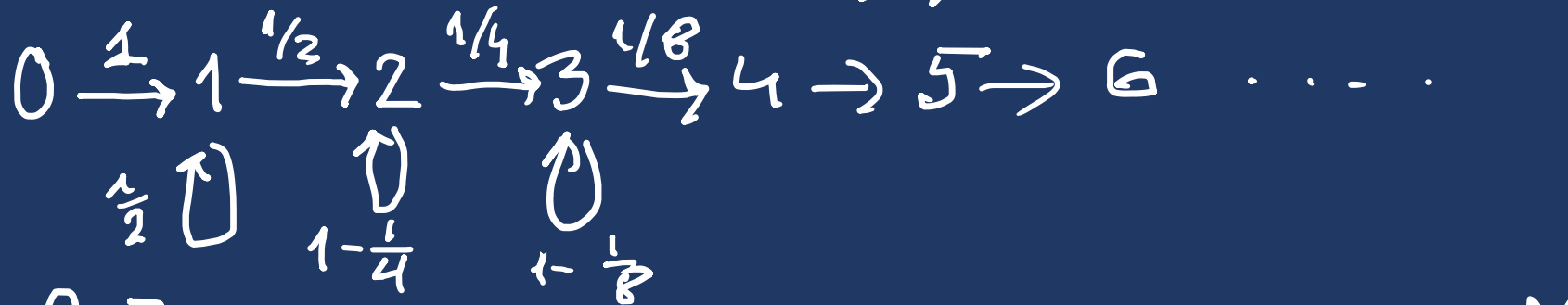
Morris Counter:

$\approx$ 1955

```
C = 0.
onInc() { if(random() < (1/2)^C) { C++; } }
```



$E[2^{C_n}] = n + 1$

C_n = the value of C after n increments

$$E\left[2^{C_0}\right] = 2^0 = 1.$$

$$E\left[2^{C_{n+1}}\right] = \sum_k 2^k P\left[C_{n+1} = k\right] =$$

$$= \sum_k 2^k \Big(P\left[C_{n+1} = k \mid C_n = k\right] P\left[C_n = k\right] + P\left[C_{n+1} = k \mid C_n = k-1\right] P\left[C_n = k-1\right] \Big) =$$

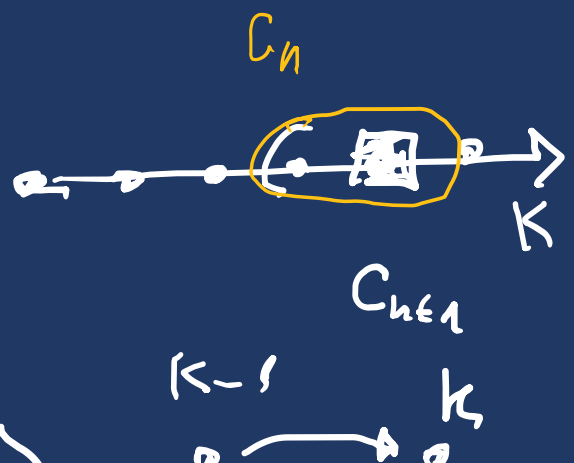
$$= \sum_k 2^k \left(\left(1 - \frac{1}{2^k}\right) P\left[C_n = k\right] + \frac{1}{2^{k-1}} P\left[C_n = k-1\right] \right)$$

$$= \sum_k 2^k P\left[C_n = k\right] - \sum_k P\left[C_n = k\right] + \sum_k 2 \cdot P\left[C_n = k-1\right] =$$

$l = k-1$

$$= E\left[2^{C_n}\right] - 1 + 2 \sum_l P\left[C_n = l\right] = E\left[2^{C_n}\right] + 1.$$

$k < 0 : P\left[C_n = k\right] = 0$



$$\begin{cases} \cdot\ \mathbb{E}[2^{C_0}] = 1 \\ \cdot\ \mathbb{E}[2^{C_{n+1}}] = \mathbb{E}[2^{C_n}] + 1 \end{cases}$$

$$\mathbb{E}[2^{C_3}] = \mathbb{E}[2^{C_2}] + 1 = \mathbb{E}[2^{C_1}] + 1 + 1 =$$
$$= \mathbb{E}[\cdot 2^{C_0}] + \underbrace{1+1+1}_{3} = 1 + 3$$

$$\boxed{\mathbb{E}[2^{C_n}] = n+1}$$

IF we know C_n then

$$\mathbb{E}[2^{C} - 1] = n$$

So we may take

$$\boxed{\hat{n} = 2^{C} - 1}$$

$$\mathbb{E}[\hat{n}] = n$$

EXERCISE: count $\mathrm{var}(2^{C_n})$.

- $\mathrm{var}(2^{C_n}) = E[(2^{C_n})^2] - (E[2^{C_n}])^2$

$\mathrm{var}(X) = E(X^2) - (E(X))^2$.

$(2^{C_n})^2 = 2^{2 \cdot C_n} = 4^{C_n}$

Try to count

$$E[4^{C_{n+1}}] = \sum_k 4^k P[C_{n+1} = k] = \dots$$

- Tchebyscev inequality:

$$P[|2^{C_n} - E[2^{C_n}]| \geq \alpha \cdot E[2^{C_n}]) \leq \frac{\mathrm{var}(2^{C_n})}{\alpha^2 (n+1)^2}$$

GEOMETRIC COUNTER

$\{a, a, b, a, b, b, c, a, \ldots\} \longrightarrow \{a, b, c\}$

how many distinct elements was in the stream (till now)

- use good hash function h

$\{a, a, b, a, b, b, \ldots\} \rightsquigarrow \{h(a), h(a), h(b) \ldots\}$

$a \neq b \longrightarrow P[h(a) = h(b)]$ is very small!

$h : \Sigma^* \longrightarrow \{0,1\}^{128}$

$a \neq b \longrightarrow P[h(a) = h(b)] \simeq \frac{1}{2^{128}}$

take h from universal family

- $L = 0$;
- onUpdate(a) {

$h(a) = (x_1 x_2 \dots x_\ell)$;

first one → $\omega(h(a)) = \min\{i : h(a)_i = 1\}$;

$L = \max(L, \omega(h(a)))$;

}

e.g. $\ell = 128 \dots$

$h(a) = (0,0,0,1,0,0,\dots)$

4

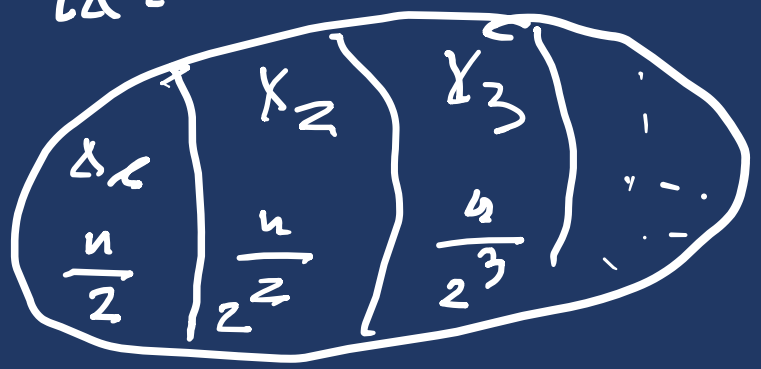
a, ~~b~~ b, a, b, c, c, c,

- $h(a) = (0\,0\,0\,1\,0\,0\,0)$: $c = 4$
- $h(b) = (0\,1\,1\,0\,0\,1\,0)$: $c = 4$

— $h(a) = (0\,0\,0\,1\dots)$: $c = 4$

- $h(c) = (0\,0\,0\,0\,1\,0\dots)$: $c = 5$

$|X| = n$ X - set



$\omega(a) = \min\{k : h(a)_k = 1\}$

$X_i^0 = \{x \in X : \omega(x) = i\}$

$X_1 = \{x \in X : h(x) = "1*"\}$ $\quad |X_1| \simeq \frac{n}{2}$

$X_2 = \{x \in X : h(x) = "01*"\}$ $\quad |X_2| \simeq \frac{n}{2^2}$

$X_3 = \{x \in X : h(x) = "001*"\}$ $\quad |X_3| \simeq \frac{n}{2^3}$

when $|X_k| \leqslant 1$? $\quad \frac{n}{2^k} \leqslant 1$

$n \leqslant 2^k$; $\quad k \geqslant \log_2 n$!!!

L_n = the value of counter after observing n dist. elements.

$$P[L_n \leq k] = \left(1 - \frac{1}{2^k}\right)^n$$

- $X_1, \dots, X_n$ – indep. rand. variables with $Geo\left(\frac{1}{2}\right)$
- $\tilde{L} = \max\{X_1, \dots, X_n\}$
- $L_n \sim \tilde{L}$.
- $E[\tilde{L}] = \frac{1}{2} + \frac{1+\gamma}{\ln 2} + r_n \approx \log_2 n$.

quite difficult

γ

$|r_n| < 0.01$

fix n

$\left(1 - \frac{1}{2^{\log_2 n}}\right)^n = \left(1 - \frac{1}{n}\right)^n \approx \frac{1}{e}$

monotonicity.

IDEA : use $\hat{n} = 2^L$ as an estimate of n

$E[L_n] \approx \log_2 n$

$L \approx \log_2 n$

$n \approx 2^L$

Problem : precision

solution :
- try to use many such counters $L_1, L_2, \ldots, L_m$

each counter $\leftrightarrow$ hash function

try use

$$\frac{L_1 + L_2 + \ldots + L_m}{m} = \text{mean}(L_1, L_2, \ldots, L_m)$$

many calculations

How to use only one hash function?

$$L = \underbrace{[L_0, L_1 \dots, L_{m-1}]}_{m}$$

$m \simeq 64$
$\simeq 128$
$m \simeq 1024$

$$x = h(a)$$

$$x = (x_1 x_2 \dots x_L) = \underbrace{(x_1 \dots x_M)}_{\text{coll. pos.}} \| (x_{M+1} \dots x_L)$$

$m = 2^M$

$$L_{(x_1 \dots x_M)_{(2)}} = \max\left(L_{(x_1 \dots x_M)_{(2)}}, \omega(x_{M+1} \dots x_L)\right)$$

```
onUpdate(a) {
    s = h(a)
    s = s_1 || s_2 ; |[s_1]| = M
    i = (s_1)_(2)
    L[i] = ω*(s_2)
}
```

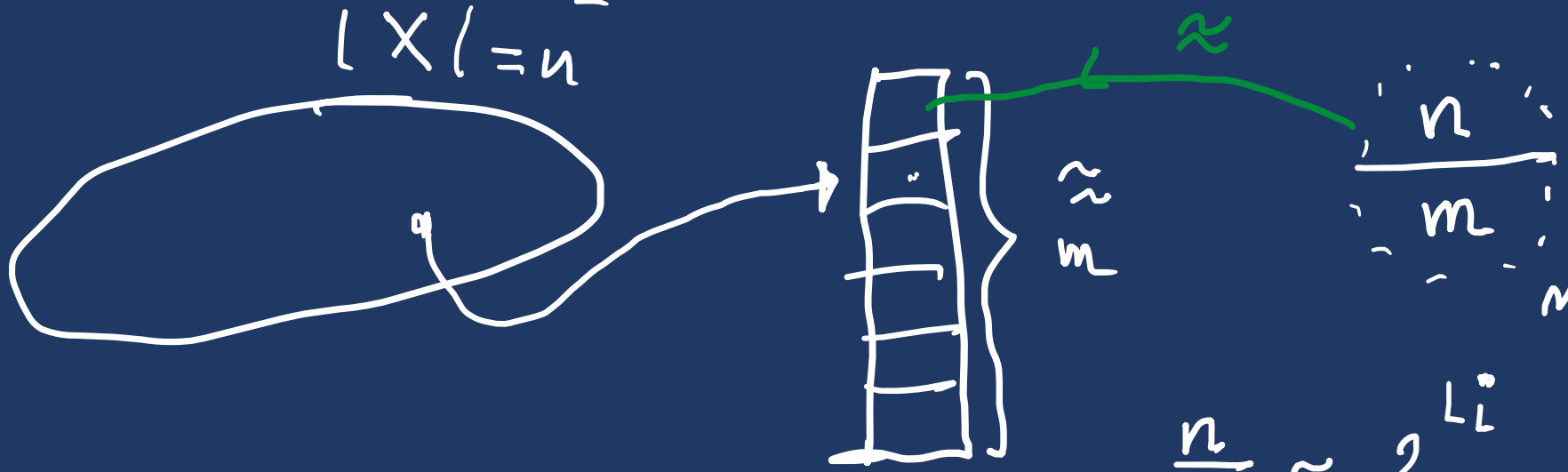
W. Rotts

$h : \Sigma \to \{0,1\}^L$

$$\omega^*(s) = \begin{cases} \min\{k : s_k = 1\} & : (\exists i)(s_i = 1) \\ (L - M) + 1 & : (\forall i)(s_i = 0) \end{cases}$$

HYPER-LOG-LOG

$\log_2(\log_2 n)$



$i = 0, \dots, m-1$

$\frac{n}{m} \simeq 2^{L_i}$ $\quad \hat{m}_1 \simeq 2^{L_i}$

$n \simeq m \cdot 2^{L_i} \quad i = 0, 1, \dots, m-1$

Try to use

$\text{mean}(m 2^{L_0}, m 2^{L_1}, \dots m 2^{L_{m-1}})$

as an estimator of n ?

$x_1 \dots x_n > 0$

Means

$$\frac{n}{\frac{1}{x_1} + \dots + \frac{1}{x_n}} \leq \sqrt[n]{x_1 \cdot \dots \cdot x_n} \leq \frac{x_1 + \dots + x_n}{n}$$

harmonic mean — geometric mean — arithmetic mean

$$\frac{n}{\frac{1}{n \cdot 2^{L_0}} + \frac{1}{n \cdot 2^{L_1}} + \dots + \frac{1}{n \, 2^{L_{n-1}}}} = \frac{n^2}{\left(\frac{1}{2}\right)^{L_0} + \dots + \left(\frac{1}{2}\right)^{L_{n-1}}}$$

this is our estimate

Hyperloglog Algorithm

HLL.

onGetEstimate() {

$$Z = m^2 / \sum_{c=0}^{m-1} \left(\frac{1}{2}\right)^{L[c]};$$

return ($\alpha_m \cdot Z$)

}

$$\alpha_m = \left(m \int_0^{\infty} \left(\log_2 \left(\frac{u+2}{u+1} \right) \right)^m du \right)^{-1}$$

m	α_m
16	0.673
32	0.69
64	0.709
≥ 128	$\frac{0.7213}{1 + \frac{1.079}{m}}$

FINAL IMPROVEMENTS

- count $z = |\{i : L[i] = 0\}|$
  IF $z \neq 0$
  use Linear Counting, e.g.
  return : $m \ln \frac{m}{z}$,
- if some some i : $L_i > \frac{2^{64}}{30}$
  correction the $L_i \leftarrow -2^{64} \ln\left(1 - \frac{L_i}{2^{64}}\right)$
  return $\left(\alpha_m \cdot Z\right)$

EXER
Implement it

HYPERLOGLOG

précision : $\frac{1.04}{\sqrt{m}}$

$m = 124$: $\sigma \approx \frac{1}{10} = 10\%$

$m = 1024$: $\sigma \approx 3\%$ ∎