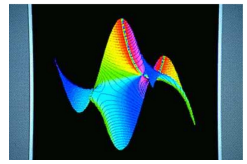


A unified approach to linear probing hashing

Svante Janson

Uppsala University

Alfredo Viola

Universidad de la República

FoCM 2014

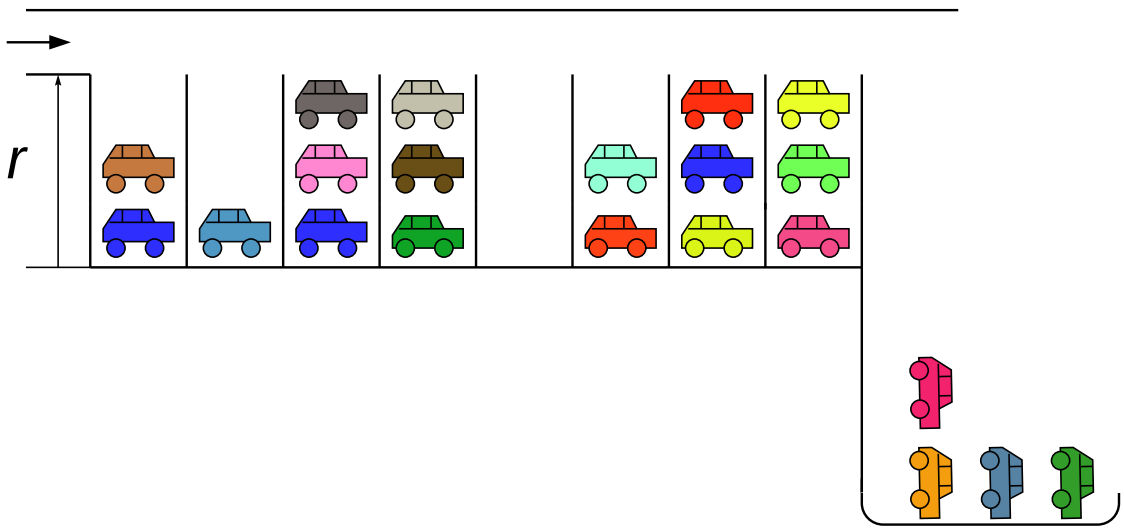
Dedicated to Philippe Flajolet

Further research: Bucket parking scheme

Bucket parking scheme

Blake and Konheim [1976]:

- Each parking lots can hold up to r cars
- Related to analysis of bucket hashing algorithms

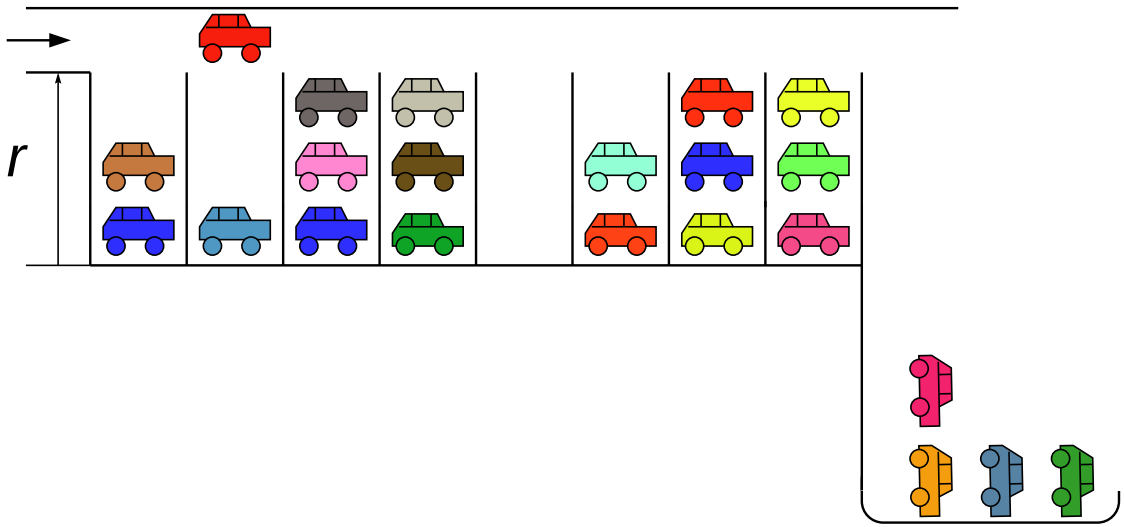


Further research: Bucket parking scheme

Bucket parking scheme

Blake and Konheim [1976]:

- Each parking lots can hold up to r cars
- Related to analysis of bucket hashing algorithms

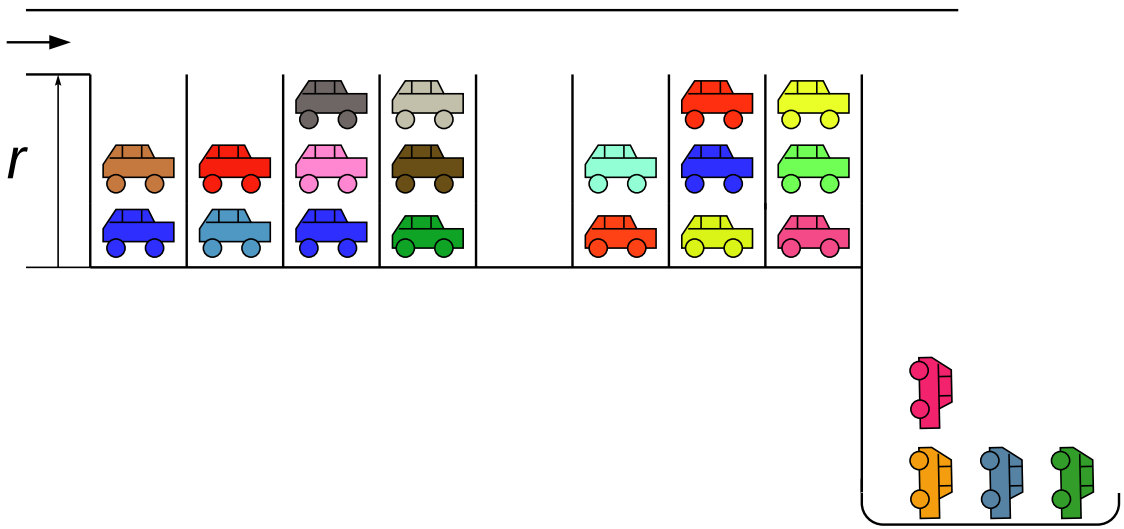


Further research: Bucket parking scheme

Bucket parking scheme

Blake and Konheim [1976]:

- Each parking lots can hold up to r cars
- Related to analysis of bucket hashing algorithms

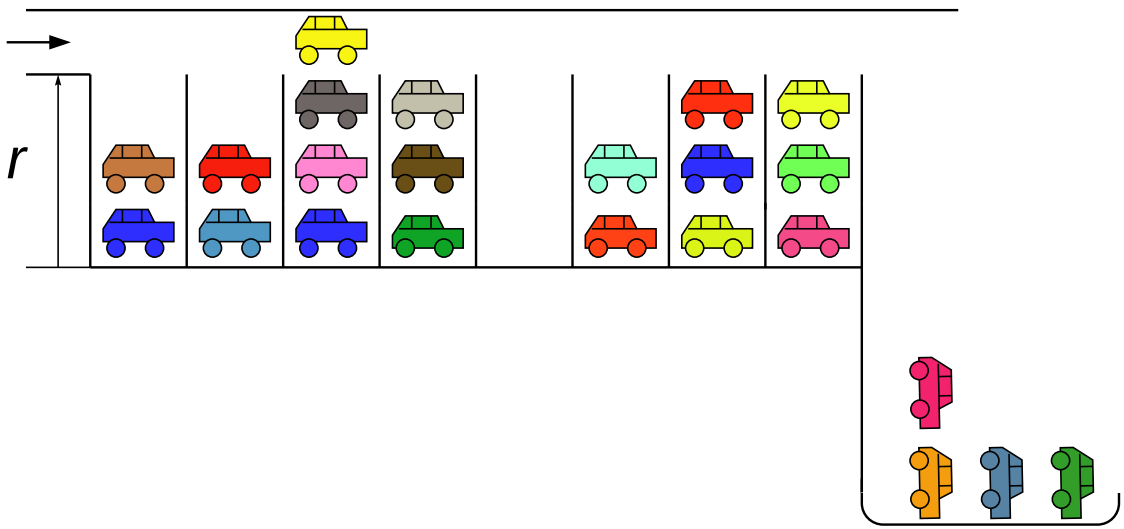


Further research: Bucket parking scheme

Bucket parking scheme

Blake and Konheim [1976]:

- Each parking lots can hold up to r cars
- Related to analysis of bucket hashing algorithms

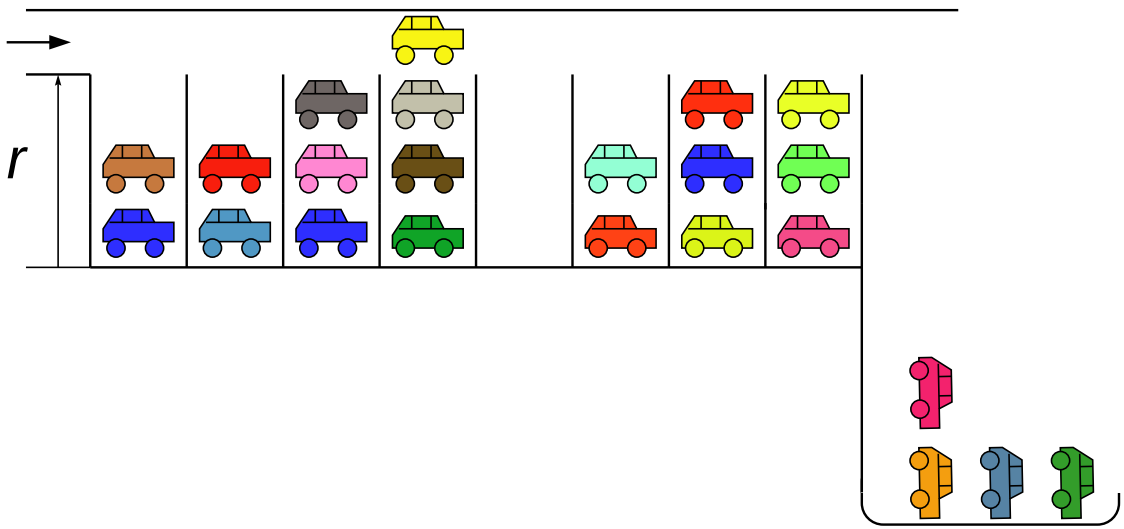


Further research: Bucket parking scheme

Bucket parking scheme

Blake and Konheim [1976]:

- Each parking lots can hold up to r cars
- Related to analysis of bucket hashing algorithms

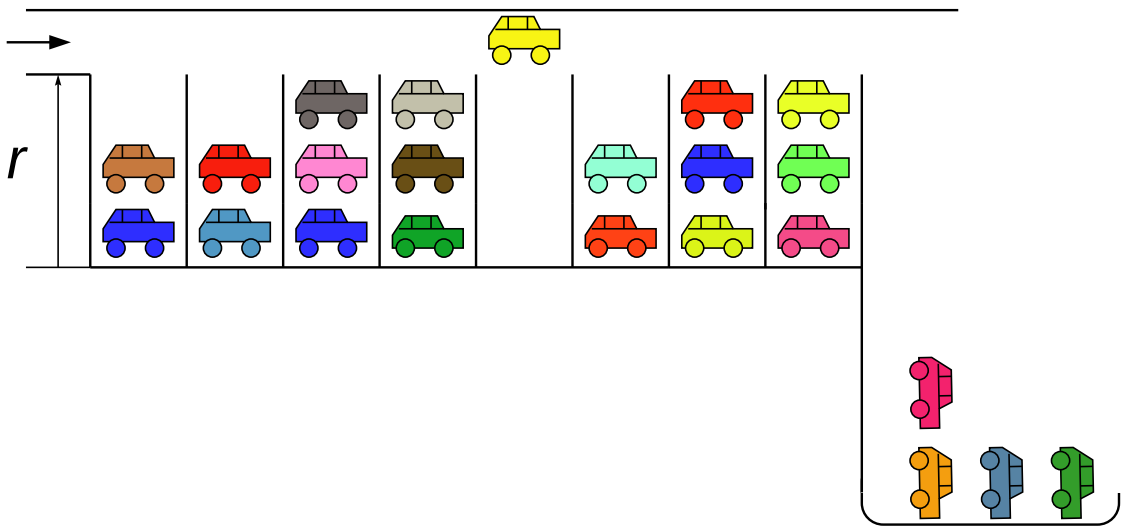


Further research: Bucket parking scheme

Bucket parking scheme

Blake and Konheim [1976]:

- Each parking lots can hold up to r cars
- Related to analysis of bucket hashing algorithms

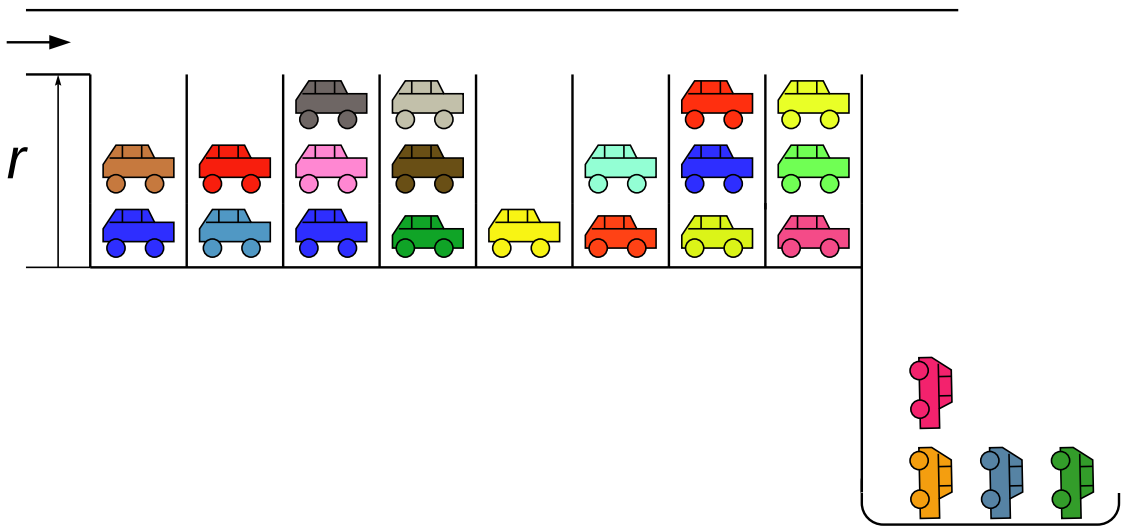


Further research: Bucket parking scheme

Bucket parking scheme

Blake and Konheim [1976]:

- Each parking lots can hold up to r cars
- Related to analysis of bucket hashing algorithms

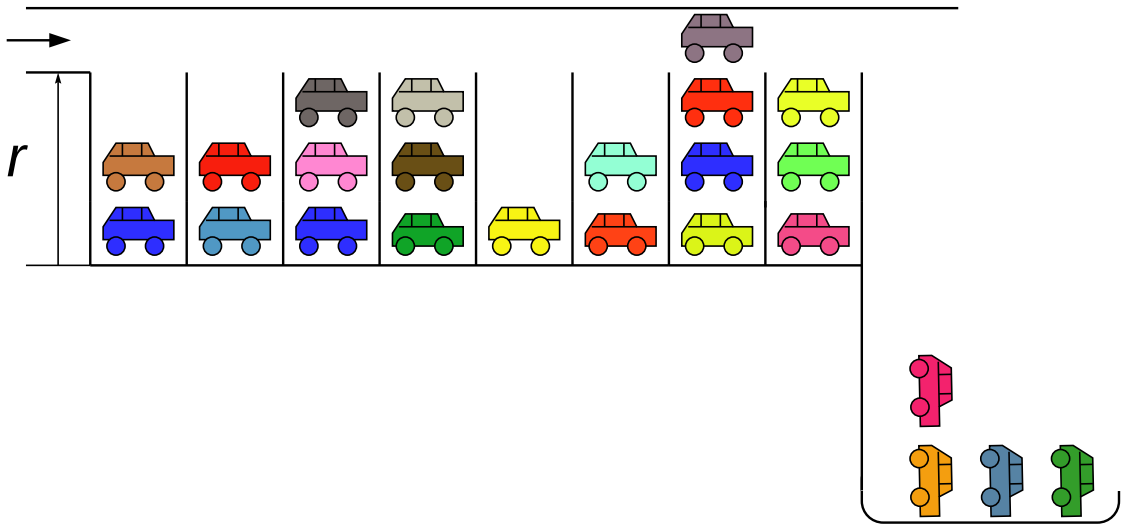


Further research: Bucket parking scheme

Bucket parking scheme

Blake and Konheim [1976]:

- Each parking lots can hold up to r cars
- Related to analysis of bucket hashing algorithms

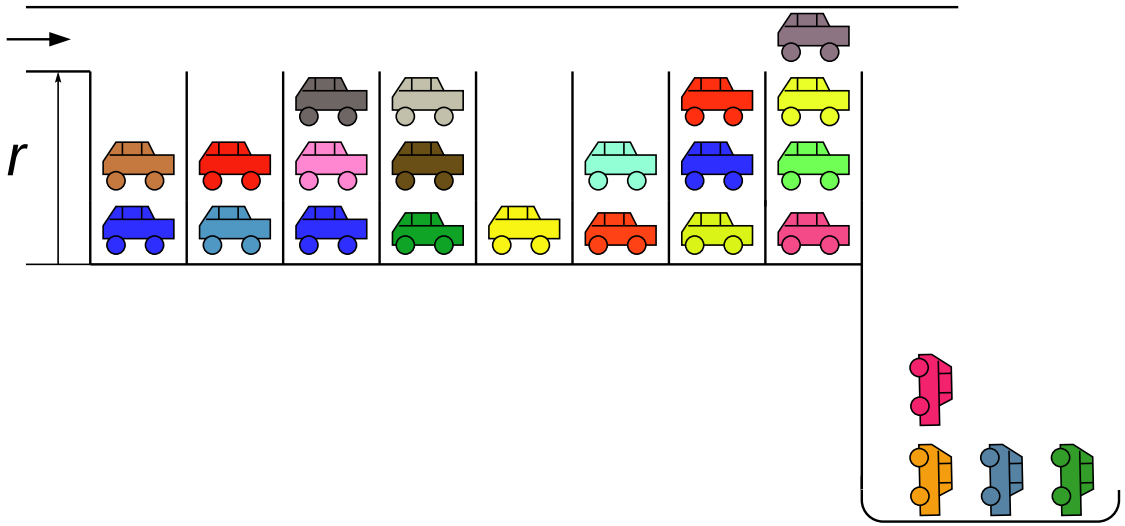


Further research: Bucket parking scheme

Bucket parking scheme

Blake and Konheim [1976]:

- Each parking lots can hold up to r cars
- Related to analysis of bucket hashing algorithms

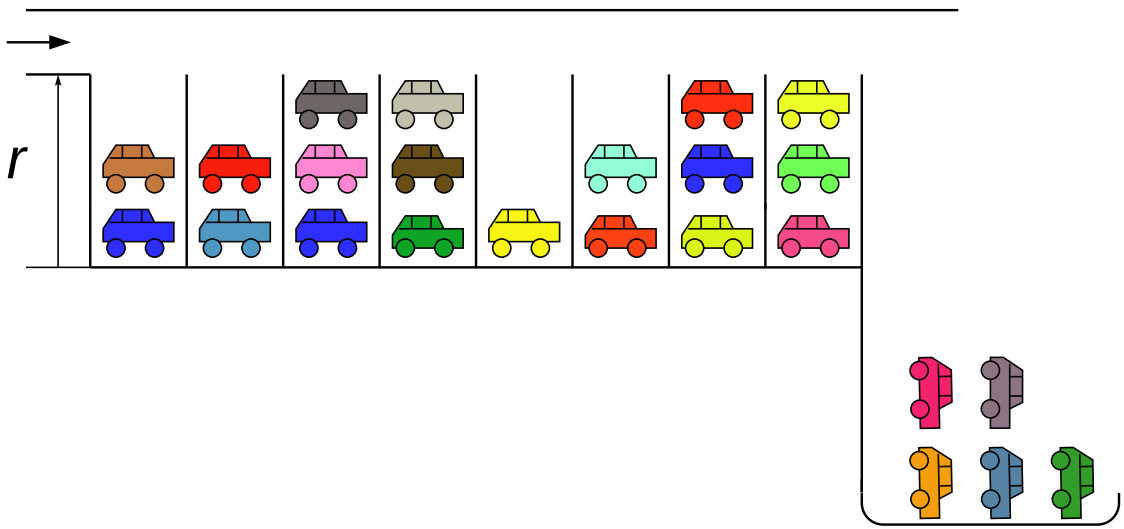


Further research: Bucket parking scheme

Bucket parking scheme

Blake and Konheim [1976]:

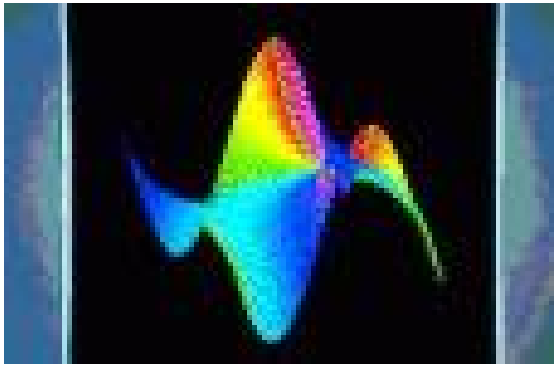
- Each parking lots can hold up to r cars
- Related to analysis of bucket hashing algorithms



The problem.

- The table has m places of capacity b to hash (park) from 0 to $m-1$, and n inserted elements (cars).
- Each element is given a hash value (preferred parking lot).
- If place is not full, then the element is stored there.
- Otherwise, looks sequentially for an empty place.
- If no empty place up to the end of the table, the search follows at location 0.
- Several R.V. to study, mainly related with cost of individual searches and total construction cost.
- Very important special case: **Parking Problem**.
- In parking the car is lost if no available place to park up to the end of the table.
- Main R.V. is the number of lost cars.

Linear Probing Hashing.



MATHÉMATIQUES ET INFORMATIQUE

Hachage, arbres, chemins & graphes

Philippe Chassaing[†] et Philippe Flajolet[‡]

The mathematical beauty of Linear Probing!

Mathématiques discrètes et continues se rencontrent et se complètent volontiers harmonieusement. C'est cette thèse que nous voudrions illustrer en discutant un problème classique aux ramifications nombreuses—l'analyse du hachage avec essais linéaires. L'exemple est issu de l'analyse d'algorithmes, domaine fondé par Knuth et qui se situe lui-même « à cheval » entre l'informatique, l'analyse combinatoire, et la théorie des probabilités. Lors de son traitement se croisent au fil du temps des approches très diverses, et l'on rencontrera des questions posées par Ramanujan à Hardy en 1913, un travail d'été de Knuth datant de 1962 et qui est à l'origine de l'analyse d'algorithmes en informatique, des recherches en analyse combinatoire du statisticien Kreweras, diverses rencontres avec les modèles de graphes aléatoires au sens d'Erdös et Rényi, un peu d'analyse complexe et d'analyse asymptotique, des arbres qu'on peut voir comme issus de processus de Galton-Watson particuliers, et, pour finir, un peu de processus, dont l'ineffable mouvement Brownien! Tout ceci contribuant *in fine* à une compréhension très précise d'un modèle simple d'aléa discret.

1962: Summer work by Don Knuth ...

[My first analysis of an algorithm, originally done during summer 1962 in Madison.]

8354

NOTES ON "OPEN" ADDRESSING. D. Knuth 7/22/63

1. Introduction and Definitions. Open addressing is a widely-used technique for keeping "symbol tables." The method was first used in 1954 by Samuel, Amdahl, and Boehme in an assembly program for the IBM 701. An extensive discussion of the method was given by Peterson in 1957 [1], and frequent references have been made to it ever since (e.g. Schay and Spruth [2], Iverson [3]). However, the timing characteristics have apparently never been exactly established, and indeed the author has heard reports of several reputable mathematicians who failed to find the solution after some trial. Therefore it is the purpose of this note to indicate one way by which the solution can be obtained.

We will use the following abstract model to describe the method: N is a positive integer, and we have an array of N variables $x_1, x_2, \ldots, x_N$. At the beginning, $x_i = 0$, for $1 \leq i \leq N$.

To "enter the k-th item in the table," we mean that an integer a_k is calculated, $1 \leq a_k \leq N$, depending only on the item, and the following process is carried out:

1. Set $j = a_k$.
2. "The comparison step." If $x_j = 0$, set $x_j = 1$ and stop; we say "the k-th item has fallen into position x_j."
3. If $j = N$, go to step 5.
4. Increase j by 1 and return to step 2.
5. "The overflow step." If this step is entered twice, the table is full, i.e. $x_i = 1$ for $1 \leq i \leq N$. Otherwise set j to 1 and return to step 2.

Observe the cyclic character of this algorithm.

We are concerned with the statistics of this method, with respect to the number of times the comparison step must be executed. More precisely, we consider all of the N^k possible sequences $a_1, a_2, \ldots a_k$ to be equally probable, and we ask, "What is the probability that the comparison step is used precisely m times when the k-th item is placed?"

2. Non-overflow (self-contained) sequences.

Let $\left[{n \atop k} \right]$ denote the number of sequences $a_1, a_2, \ldots a_k$ $(1 \leq a_i \leq n)$ in which no overflow step occurs during the entire process of placing k items, if the algorithm is used for $N = n$. (By convention, we set

$$\left[{n \atop 0} \right] = 1.)$$

Lemma 1: If $0 \leq k \leq n+1$, then $\left[{n \atop k} \right] = (n+1)^k - k(n+1)^{k-1}$.

Proof: This proof is based on the fact that $\left[{n \atop k} \right]$ is precisely the number of sequences $b_1, b_2, \ldots b_k$ $(1 \leq b_i \leq n+1)$ in which, if the algorithm is carried out for $N = n+1$, then $x_{n+1} = 0$ at the end of the operation. This follows because every sequence of the former type is one of the latter, and conversely the condition implies in particular that $1 \leq b_i \leq n$, and that no overflow step occurs.

But sequences of the latter type are easily enumerated, because the algorithm has circular symmetry; of the $(n+1)^k$ possible sequences $b_1, b_2, \ldots b_k$, exactly $k/(n+1)$ of these leave $x_{n+1} \neq 0$. This shows that

$$\left[{n \atop k} \right] = (n+1)^k \left(1 - \frac{k}{n+1} \right).$$

Original results.

- Let a hash table with m positions and n inserted elements.
- Let $P_{m,n}$ the probability of the last position being empty.
  - $P_{m,n} = \left(1 - \frac{n}{m}\right)$.
- Let $C_{m,n}$ the R.V. for the number of successful searches of a random element.
  - $E\left[C_{m,n}\right] = \frac{1}{2}\left(1 + Q_0((m, n-1)\right)$.
  - $E\left[C_{m,\alpha m}\right] = \frac{1}{2}\left(1 + \frac{1}{1-\alpha}\right)$ with $0 \leq \alpha < 1$.
  - $E\left[C_{n,n}\right] = \sqrt{\frac{\pi n}{8}} + O(1)$ (proved on may 20, 1965).
- Let $U_{m,n}$ the R.V. for the number of unsuccessful searches of a random element.
  - $E\left[U_{m,n}\right] = \frac{1}{2}\left(1 + Q_1((m, n-1)\right)$.
  - $E\left[U_{m,\alpha m}\right] = \frac{1}{2}\left(1 + \frac{1}{(1-\alpha)^2}\right)$ with $0 \leq \alpha < 1$.
- The Ramanujan Q function is the special case $Q_0(n, n)$ of

$$Q_r(m, n) = \sum_{k=0}^{n} \binom{k+r}{k} \frac{n^{\underline{k}}}{m^n}.$$

Linear Probing and the Symbolic Method (I).

Linear Probing and Graphs

Donald E. Knuth, Stanford University

Dedicated to Philippe Patrick Michel Flajolet

Abstract. Mallows and Riordan showed in 1968 that labeled trees with a small number of inversions are related to labeled graphs that are connected and sparse. Wright enumerated sparse connected graphs in 1977, and Kreweras related the inversions of trees to the so-called "parking problem" in 1980. A combination of these three results leads to a surprisingly simple analysis of the behavior of hashing by linear probing, including higher moments of the cost of successful search.

Linear Probing and the Symbolic Method (I).

The purpose of this note is to exhibit a surprisingly simple solution to a problem that appears in a recent book by Sedgewick and Flajolet [9]:

> **Exercise 8.39** Use the symbolic method to derive the EGF of the number of probes required by linear probing in a successful search, for fixed M.

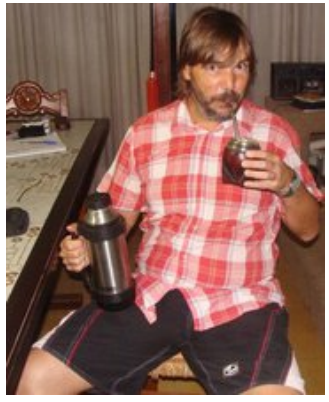
The authors admitted that they did not know how to solve the problem, in spite of the fact that a "symbolic method" was the key to the analysis of all the other algorithms in their book. Indeed, the second moment of the distribution of successful search by linear probing was unknown when [9] was published in 1996.

Conclusions (Knuth).

7. Personal remarks. The problem of linear probing is near and dear to my heart, because I found it immensely satisfying to deduce (5.4) when I first studied the problem in 1962. Linear probing was the first algorithm that I was able to analyze successfully, and the experience had a significant effect on my future career as a computer scientist. None of the methods available in 1962 were powerful enough to deduce the expected square displacement, much less the higher moments, so it is an even greater pleasure to be able to derive such results today from other work that has enriched the field of combinatorial mathematics during a period of 35 years.

The reader will note that Sedgewick and Flajolet's exercise 8.39 has not truly been solved, strictly speaking, because we have not found the EGF $\sum_{n=0}^{m-1} F_{mn}(x)\, z^n/n!$ as requested. However, Sedgewick and Flajolet should be happy with any analysis of linear probing that uses symbolic methods associated with generating functions in an informative way.

Linear Probing and the Symbolic Method (II).



Algorithmica (1998) 22: 490–515

Algorithmica

On the Analysis of Linear Probing Hashing[1]

P. Flajolet,[2] P. Poblete,[3] and A. Viola[4]

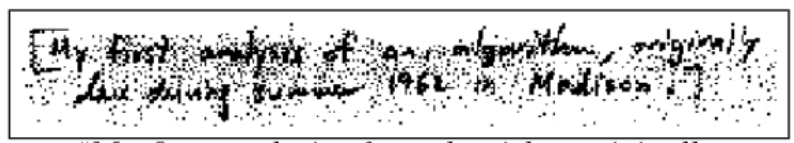
Combinatorial interpretation.

29	69	10		24		36	77	18	58
49	79					56	97	38	78
0	1	2	3	4	5	6	7	8	9

- Any Linear Probing Hash table can be seen as a sequence of almost full tables (a subtable with all but the last bucket full).
- **Example:** [3-3],[4-4],[5-5],[6-2].
- This interpretation can be nicely handled by Analytic Combinatorics, since for example, it implies that it is enough to study almost full tables, and then use the *sequence* construction.

ON THE ANALYSIS OF LINEAR PROBING HASHING

Philippe Flajolet, INRIA Rocquencourt
(France)



"My first analysis of an algorithm originally done during Summer 1962 at Madison."

CONSTRUCTIONS

Dictionary (I)

$$\mathcal{F} \quad \mapsto \quad \{f_n\} \quad \mapsto \quad f(z) = \sum_n f_n \frac{z^n}{n!}.$$

$$\frac{1}{1-f} = 1 + f + f^2 + f^3 + \cdots$$

$$\exp(f) = 1 + f + \frac{1}{2!}f^2 + \frac{1}{3!}f^3 + \cdots$$

$$\begin{aligned}
\mathbf{A} \cup \mathbf{B} &\mapsto A(z) + B(z) \\
\mathbf{A} \times \mathbf{B} &\mapsto A(z) \times B(z) \\
\text{Seq } \mathbf{A} &\mapsto \frac{1}{1 - A(z)} \\
\text{Set } \mathbf{A} &\mapsto \exp(A(z)) \\
\text{Cycle } \mathbf{A} &\mapsto \log \frac{1}{1 - A(z)}
\end{aligned}$$

4 L.P.H.: Generating functions

Almost-full tables $n = m - 1$ have **tree decomposition**.

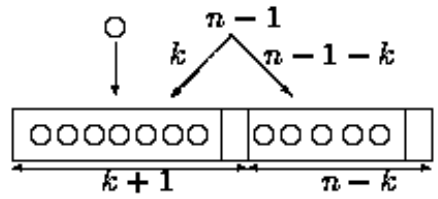


FIGURE 1. The binary tree decomposition of almost full tables.

```
-------------------------------------
|  <Full> := <Full> * <Last> * <Full> |
|            ^^^^^^  with Position    |
-------------------------------------
```

Dictionary: Products $\mapsto$ Products, $C_n = \sum \binom{n}{k} A_k B_{n-k}$

$$\mathcal{C} = \mathcal{A} \star \mathcal{B} \qquad \mapsto \qquad C(z) = A(z) \cdot B(z)$$

Dictionary: Adding an element $\mapsto \int$, $C_n = A_{n-1}$

$$\mathcal{C} = \mathsf{Add}(\mathcal{A}) \qquad \mapsto \qquad C(z) = \int_0^z A(w)\, dw.$$

Dictionary: Choosing a position $\mapsto \partial$, $C_n = (n+1)A_n$

$$\mathcal{C} = \mathsf{Pos}(\mathcal{A}) \qquad \mapsto \qquad \frac{\partial}{\partial z}\left(z\, A(z)\right).$$

A nonlinear ODE translates Linear Probing

$$\begin{aligned} \mathit{Table} &= \mathit{Table} \text{ `}\star\text{' } \mathit{Table} \\ F &= \int (zF)' F \\ T'(1-\tfrac{1}{T}) &= \tfrac{1}{z} \qquad \text{with } T = zF \end{aligned}$$

Lemma 1.

$$F(z) = \frac{1}{z} T(z) \quad \text{where} \quad T = ze^T$$

Lemma 2. [Lagrange + Eisenstein + Cayley]

$$T(z) = z + 2\frac{z^2}{2!} + 9\frac{z^3}{3!} + \cdots = \sum_n n^{n-1} \frac{z^n}{n!}$$

Defines the Tree function $T(z)$

```
Tree = Root * Set(Tree)
```

[Knuth63] The number of almost full tables (n keys)

$$F_n = (n+1)^{(n-1)}$$

Distributional equations (almost full tables)

- *construction cost* = *total displacement*

```
-------------------------------------
|  <Full> := <Full> * <Last> * <Full> |
|             ^^^^^^  with Position   |
-------------------------------------
```

$$F_n(q) = \sum_{k=0}^{n-1} \binom{n-1}{k} F_k(q)(1+q+\cdots+q^k)F_{n-1-k}(q).$$

- q–Calculus: The construction cost is worst-case quadratic

$$\sum_n q^{n^2} z^2$$

Q-calculus

$$
\begin{array}{lcl}
n & \mapsto & [n] = 1 + q + q^2 + \cdots q^{n-1} = \dfrac{1-q^n}{1-q} \\
n! & \mapsto & [n]! = [1] \cdot [2] \cdots [n] \\
\sum (n+1) f_n z^n & \mapsto & \sum [n+1] f_n z^n \\
\dfrac{\partial}{\partial z}(z f(z)) & \mapsto & \mathrm{H}[f(z)] = \dfrac{F(z) - qF(qz)}{1-q}
\end{array}
$$

Many "combinatorial identities" survive in the q-world, with $q^\star \mapsto 1$

EXAMPLE. [Euler] $(\exp(z))^{-1} = (\exp(-z))$

$$\left(\sum \frac{z^n}{n!} \right)^{-1} = \left(\sum \frac{(-z)^n}{n!} \right)$$

$$\left(\sum \frac{z^n}{[n]!} \right)^{-1} = \left(\sum q^{n(n-1)/2} \frac{(-z)^n}{[n]!} \right)$$

$$\partial_z F(z) = \partial_z(zF(z)) \cdot F(z)$$

$$\partial_z F(z,q) = \left(\frac{F(z,q) - qF(qz,q)}{1-q}\right) \cdot F(z,q)$$

$$\partial_z F = \mathrm{H}\, F \cdot F$$

Moments result from

— ∂_q, differentiation w.r.t. q

— U, setting $q = 1$

For the r-th moment, apply $U\,\partial_q^r$ to difference-differential equation

Lemma. Commutation rule with $\mathrm{Z}[f] = z \cdot f$

$$\begin{aligned}
\mathrm{U}\,\mathrm{H} &= \partial_z \mathrm{Z} \\
\mathrm{U}\,\partial_q\,\mathrm{H} &= \mathrm{U}\,\partial_z \mathrm{Z} \partial_q + \frac{1}{2}\mathrm{Z}\partial_z^2 Z, \\
\mathrm{U}\,\partial_q^2\,\mathrm{H} &= \partial_z \mathrm{Z}\,\mathrm{U}\,\partial_q^2 + \mathrm{Z}\partial_z^2 \mathrm{Z}\,\mathrm{U}\,\partial_q + \tfrac{1}{3}\mathrm{Z}^2\partial_z^3 \mathrm{Z}\,\mathrm{U}
\end{aligned}$$

Proof. $\partial_q[n+1]$, Leibniz, &c.

Decomposition.

- Let F_{bi+d} be the number of ways to construct an almost full table of length $i+1$ and size $bi+d$ with $0 \leq d \leq b-1$. Then,

$$F_d(u) := \sum_{i \geq 0} F_{bi+d} \frac{u^{bi+d}}{(bi+d)!}, \quad N_d(z,w) := \sum_{s=0}^{b-1-d} w^{b-s} F_s(zw).$$

- $N_d(z,w)$ is the generating function for the number of almost full tables with more than d empty locations in the last bucket [Blake and Konheim 1977], [V. 2010].

$$\sum_{d=0}^{b-1} F_d(bz) x^d = x^b - \prod_{j=0}^{b-1} \left(x - \frac{T(\omega^j z)}{z} \right),$$

$$N_d(bz,w) = \left[x^d \right] \frac{\prod_{j=0}^{b-1} \left(1 - x \frac{T(\omega^j zw)}{z} \right) - \prod_{j=0}^{b-1} \left(1 - \frac{T(\omega^j zw)}{z} \right)}{1-x},$$

where T is the Tree function and ω is a b-th root of unity.

Decomposition.

- Let $Q_{m,n,d}$ be the number of ways of inserting n elements into m buckets, where the last bucket has more than d empty slots.
- By a direct application of the sequence construction we have

$$\Lambda_d(bz, w) := \sum_{m \geq 0} \sum_{n \geq 0} Q_{m,n,d} \frac{(bz)^n}{n!} w^{bm} = 1 + \frac{N_d(bz, w)}{1 - N_0(bz, w)}.$$

- $\Lambda_0(z, w)$ is the generating function for the number of ways to construct hash tables such that their last bucket is not full.

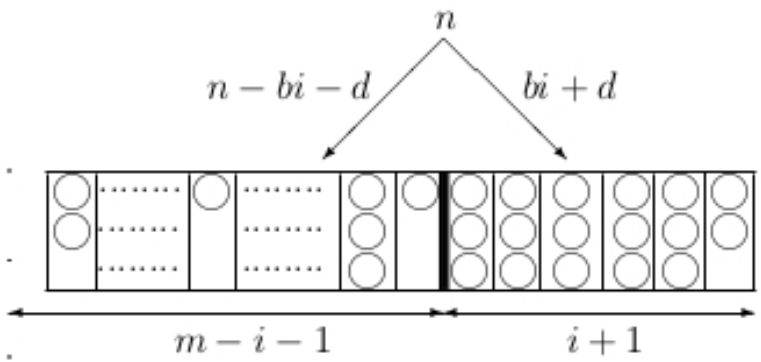


Fig. 1: A decomposition for $b = 3$ and $d = 2$.

Decomposition.

- Let P be a property (e.g. cost of a successful search or block length).

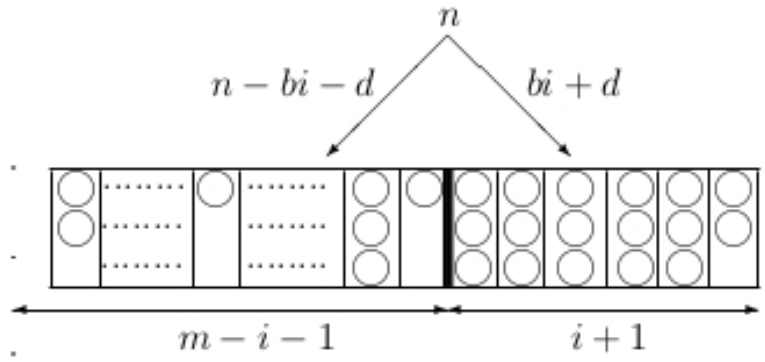


Fig. 1: A decomposition for $b = 3$ and $d = 2$.

- Let $p_{bi+d}(q)$ be the probability generating function of P calculated in the last cluster. Then

$$p_{m,n}(q) = \sum_{d=0}^{b-1} \sum_{i \geq 0} \binom{n}{bi+d} Q_{m-i-1,n-bi-d,0} \; F_{bi+d} \; p_{bi+d}(q).$$

Decomposition.

- As a consequence,

$$P(z,w,q) := \sum_{m,n\geq 0} p_{m,n}(q)\; w^{bm}\frac{z^n}{n!} = \frac{\hat{N}_0(z,w,q)}{1-N_0(z,w)}, \qquad \text{with}$$

$$\hat{N}_0(z,w,q) := \sum_{d=0}^{b-1} w^{b-d} \sum_{i\geq 0} F_{bi+d}\,\frac{(zw)^{bi+d}}{(bi+d)!}\; p_{bi+d}(q),$$

  which could be directly derived with the sequence construction.
- $\hat{N}_0(z,w,1) = N_0(z,w)$, $P(z,w,1) = \Lambda_0(z,w) - 1$.
- Moreover, by singularity analysis of $\Lambda_0(\alpha, y^{1/b}e^{-\alpha}, q)$ at $y = 1$,

$$\lim_{m\to\infty} \mathbf{P}_m[Q_{m,n,0}/m^n; b\alpha] = T_0(b\alpha) = \frac{b(1-\alpha)}{\prod_{j=1}^{b-1}\left(1-\frac{T(\omega^j \alpha e^{-\alpha})}{\alpha}\right)}.$$

- As a consequence,

$$\lim_{m\to\infty} \mathbf{P}_m[p_{m,n}(q)/m^n; b\alpha] \;=\; T_0(b\alpha)\hat{N}_0(b\alpha, e^{-\alpha}, q).$$

Summary of the q-calculus.

Marking a position $\mapsto \partial_w$ $\mathcal{C} = \mathrm{Pos}(\mathcal{A})$	$C_{bn+d} = (n+1)A_{bn+d}$ $C(z,w) = \frac{w}{b}\frac{\partial}{\partial w}(A(z,w))$
Adding a key $\mapsto \int$ $\mathcal{C} = \mathrm{Add}(\mathcal{A})$	$C_{bn+d} = A_{bn+d-1}$ $C(z,w) = \int_0^z A(u,w)du$
Bucketing $\mapsto \exp$ $\mathcal{C} = \mathrm{Bucket}(\mathcal{Z})$	$C_{m,n} = \delta(m,1)$ $C(z,w) = w^b \exp(z)$
Marking a key $\mapsto \partial_z$ $\mathcal{C} = \mathrm{Mark}(\mathcal{A})$	$C_{m,n} = nA_{m,n}$ $C(z,w) = z\frac{\partial}{\partial z}A(z,w)$

$$n \mapsto [n] = 1 + q + q^2 + \ldots + q^{n-1} = \frac{1-q^n}{1-q}.$$

$$\sum (n+1) f_n z^n \mapsto \sum [n+1] f_n z^n.$$

$$\frac{w}{b}\frac{\partial}{\partial w}A(z,w) \mapsto \mathrm{H}[A(z,w)] := \frac{A(z,w) - A(z,wq^{\frac{1}{b}})}{1-q}$$

$$z\frac{\partial}{\partial z}A(z,w) \mapsto \hat{\mathrm{H}}[A(z,w)] := \frac{A(z,w) - A(qz,w)}{1-q}.$$

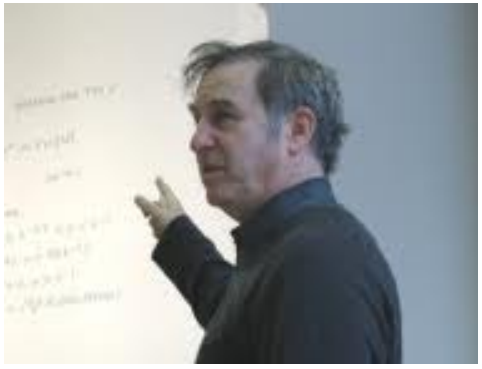
Two models to analyze the problem.

- **Exact filling model.**
  - A fixed number of keys n, are distributed among m locations, and all m^n possible arrangements are equally likely to occur.

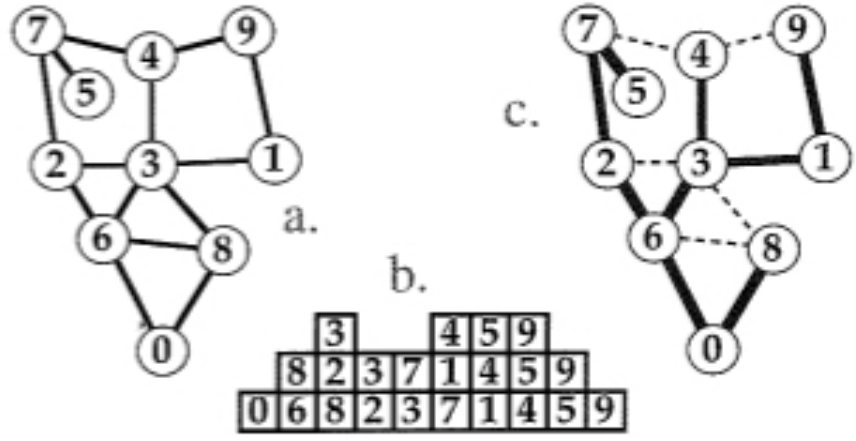
- **Poisson model.**
  - Each location receives a number of keys that is Poisson distributed with parameter $b\alpha$, and is **independent** of the number of keys going elsewhere. This implies that the total number of keys, N, is itself a Poisson distributed random variable with parameter $b\alpha m$:

$$Pr[N = n] = \frac{e^{-b\alpha m}(b\alpha m)^n}{n!}.$$

Parking, random graphs and random trees.



[Spencer 1997].



- BFS traversal of a random graph γ with vertices $\{0, 1, \ldots, n\}$.
- Induces a queue $(H_i(\tau))_{1\leq i\leq n}$ and a spanning tree τ.
- BFS induces a parking sequence $(X_i(\tau))_{1\leq i\leq n}$.
- Ex: $(X_i(\tau)) = \{\{6,8\},\{2,3\},\phi,\{7\},\{1,4\},\{5\},\{9\},\phi,\phi\}$.
- $x_i(\tau) = |X_i(\tau)|$. Ex: $(x_i(\tau)) = \{2,2,0,1,2,1,1,0,0\}$.
- $y_i(\tau) = x_1(\tau) + x_2(\tau) + \ldots + x_i(\tau) - i + 1$, size of queue $(H(\tau))$ before step i. Ex: $(y_i(\tau)) = \{2,3,2,2,3,3,3,2,1\}$.
- $y_1(\tau) + \ldots y_n(\tau) - n$ is the total displacement. Ex: 12.
- BFS induces a random walk excursion. Ex: b.

Ideas about the probabilistic approach.

- Let X_i be the number of elements that have hash address i.
- Let H_i be the total number of elements that try bucket i.
- let Q_i be the *overflow* from bucket i.
- We thus have the equations

$$H_i = X_i + Q_{i-1}, \qquad Q_i = (H_i - b)_+.$$

- The number of elements stored in bucket i is $Y_i = \min(H_i, b)$.
- The bucket is full if and only if $H_i \geq b$.

Finite and infinite hash tables.

Lemma

Let X_i, $i \in \mathfrak{T}$, be given non-negative integers. If $\mathfrak{T} = \{1, \ldots, m\}$ or $\mathbb{N}$, then the equations (??), for all $i \in \mathfrak{T}$, have a unique solution given by, considering $j \geq 0$,

$$H_i = \max_{j<i} \sum_{k=j+1}^{i} (X_k - b) + b, \qquad Q_i = \max_{j \leq i} \sum_{k=j+1}^{i} (X_k - b)$$

- Properly defined, the result can be extended to infinite tables.

Convergence to an infinite hash table.

- We are interested in hashing on Z_m with n elements having independent uniformly random hash addresses.
- $X_1, \ldots, X_m$ have a multinomial distribution with parameters n and $(1/m, \ldots, 1/m)$. (We denote these X_i by $X_{m,n;i}$.)
- We denote the profile of this hash table by $H_{m,n;i}$, where as above $i \in Z_m$ but we also can allow $i \in \mathbb{Z}$ in the obvious way.
- We consider a limit with $m, n \to \infty$ and $n/bm \to \alpha \in (0, 1)$.
- The appropriate limit object is an infinite hash table on $\mathbb{Z}$ with $X_i = X_{\alpha;i}$ that are independent and identically distributed (i.i.d.) with the Poisson distribution $X_i \sim \mathrm{Po}(\alpha b)$.
- We denote the profile of this hash table by $H_{\alpha;i}$.

Lemma

Let $m, n \to \infty$ with $n/bm \to \alpha$ for some α with $0 < \alpha < 1$. Then $(H_{m,n;i})_{i=-\infty}^{\infty} \xrightarrow{\mathrm{d}} (H_{\alpha;i})_{i=-\infty}^{\infty}$.

Poisson Transform.

- Results in one model can be transfered into the other model by the Poisson Transform:

$$P_m[f_{m,n}; b\alpha] = \sum_{n \geq 0} Pr[N = n] f_{m,n} = e^{-b\alpha m} \sum_{n \geq 0} \frac{(b\alpha m)^n}{n!} f_{m,n}.$$

Inversion Theorem:
[Gonnet and Munro 1984]

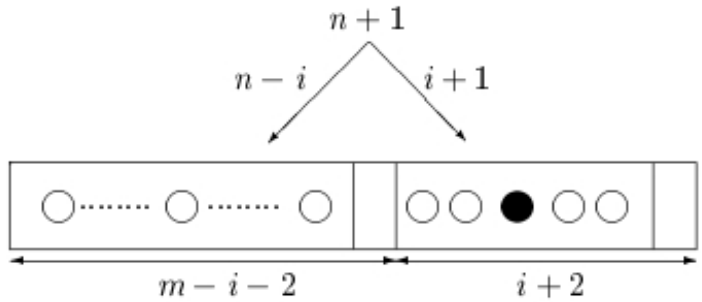
$$\text{If } \mathbf{P}_m[f_{m,n}; b\alpha] = \sum_{k \geq 0} a_{m,k} (bm\alpha)^k \text{ then } f_{m,n} = \sum_{k \geq 0} a_{m,k} \frac{n^{\underline{k}}}{(bm)^k}.$$

- The Poisson model is an approximation of the exact filling model when $n, m \to \infty$ with $n/m = b\alpha$ with $0 \leq \alpha < 1$.

Diagonal Poisson Transform.

,

[Munro, Poblete, Viola 1997]



- Let a hash table of size m, with $n+1$ keys, and let P be a property for $\bullet$ (chosen uniformily at random).
- Let $f_{m,n}$ be the result of applying a linear operator (e.g. an expected value) to the probability generating function of P.

$$f_{m,n} = \sum_{i \geq 0} \Pr[\bullet \in \text{cluster of size } i+1]\ f_{i+2,i}$$

$$= \sum_{i \geq 0} \binom{n}{i} \frac{(m-i-2)^{n-i-1}(m-n-2)(i+2)^i}{m^n} f_{i+2,i}.$$

- Then $\mathcal{P}_m[f_{m,n};\alpha] = \mathcal{D}_2[f_{n+2,n};\alpha]$ with

$$\mathcal{D}_c[f_n;\alpha] = (1-\alpha) \sum_{n \geq 0} e^{-(n+c)\alpha} \frac{((n+c)\alpha)^n}{n!} f_n.$$

Decomposition.

- As a consequence,

$$P(z,w,q) := \sum_{m,n\geq 0} p_{m,n}(q)\; w^{bm}\frac{z^n}{n!} = \frac{\hat{N}_0(z,w,q)}{1-N_0(z,w)}, \qquad \text{with}$$

$$\hat{N}_0(z,w,q) := \sum_{d=0}^{b-1} w^{b-d} \sum_{i\geq 0} F_{bi+d}\,\frac{(zw)^{bi+d}}{(bi+d)!}\; p_{bi+d}(q),$$

  which could be directly derived with the sequence construction.
- $\hat{N}_0(z,w,1) = N_0(z,w)$, $P(z,w,1) = \Lambda_0(z,w) - 1$.
- Moreover, by singularity analysis of $\Lambda_0(\alpha, y^{1/b}e^{-\alpha}, q)$ at $y = 1$,

$$\lim_{m\to\infty} \mathbf{P}_m[Q_{m,n,0}/m^n; b\alpha] = T_0(b\alpha) = \frac{b(1-\alpha)}{\prod_{j=1}^{b-1}\left(1-\frac{T(\omega^j \alpha e^{-\alpha})}{\alpha}\right)}.$$

- As a consequence,

$$\lim_{m\to\infty} \mathbf{P}_m[p_{m,n}(q)/m^n; b\alpha] \;=\; T_0(b\alpha)\hat{N}_0(b\alpha, e^{-\alpha}, q).$$

Generating Functions and the Poisson Transform.

- Let $P_{m,n}(q)$ be the generating function of a cumulated value of a RV χ in a hash table of size m with n elements. Let

$$P(z,w,q) = \sum_{m\geq 0} w^{bm} \sum_{n\geq 0} \frac{P_{m,n}(q)}{m^n} \frac{(bmz)^n}{n!}.$$

- Then, for a fixed $0 < \alpha < 1$,

$$\begin{aligned} P(\alpha, y^{1/b}e^{-\alpha}, q) &= \sum_{m\geq 0} y^m \left(e^{-bm\alpha} \sum_{n\geq 0} \frac{P_{m,n}(q)}{m^n} \frac{(bm\alpha)^n}{n!} \right) \\ &= \sum_{m\geq 0} y^m P_m \left[\frac{P_{m,n}(q)}{m^n}; b\alpha \right]. \end{aligned}$$

- Results for the probability generating function in the Poisson Model ($n, m \to \infty, 0 \leq n/bm = \alpha < 1$) can be found by singularity analysis from $P(\alpha, y^{1/b}e^{-\alpha}, q)$. In our problems, the dominant singularity is at $y = 1$.

The overflow (parking problem).

- Let $N_{m,n,k}$ be the number of tables of length m with n elements and overflow k and

$$\Omega(z,w,q) := \sum_{m\geq 0}\sum_{n\geq 0}\sum_{k\geq 0} N_{m,n,k} w^{bm}\frac{z^n}{n!}q^k.$$

Theorem ([Seitz and Panholzer 2009])

$$\Omega(bz,w,q) = \frac{1}{q^b - w^b e^{qbz}} \cdot \frac{\prod_{j=0}^{b-1}\left(q - \frac{T(\omega^j zw)}{z}\right)}{\prod_{j=0}^{b-1}\left(1 - \frac{T(\omega^j zw)}{z}\right)}.$$

Proof.

[Sketch]

$$\Omega(z,w,q) = 1 + \Omega(z,w,q)\frac{w^b e^{zq}}{q^b} + \sum_{s=0}^{b-1}(1-q^{s-b})O_s(z,w),$$

with

$$O_s(z,w) = \frac{F_s(zw)w^{b-s}}{1 - N_0(z,w)}.$$

The overflow (parking problem).

- Let $Q_{m,n}$ denote the overflow in a random hash table with m buckets and n keys.

Corollary

$$\mathbb{E}\, Q_{m,n} = m^{-n} \sum_{j=0}^{n} \sum_{k=1}^{\lfloor j/b \rfloor} \binom{n}{j} (j - kb) k^{j-1} (m-k)^{n-j}.$$

The parking problem.

Theorem

Let $0 < \alpha < 1$. The probability generating functions $\psi_H(z)$ and $\psi_Q(z)$ extend to meromorphic functions given by

$$\psi_H(z) = \frac{b(1-\alpha)(z-1)}{z^b e^{\alpha b(1-z)} - 1} \frac{\prod_{\ell=1}^{b-1}(z - T(\omega^\ell \alpha e^{-\alpha})/\alpha)}{\prod_{\ell=1}^{b-1}(1 - T(\omega^\ell \alpha e^{-\alpha})/\alpha)},$$

$$\psi_Q(z) = \frac{b(1-\alpha)(z-1)}{z^b - e^{\alpha b(z-1)}} \frac{\prod_{\ell=1}^{b-1}(z - T(\omega^\ell \alpha e^{-\alpha})/\alpha)}{\prod_{\ell=1}^{b-1}(1 - T(\omega^\ell \alpha e^{-\alpha})/\alpha)}.$$

Corollary

In the infinite Poisson model, the probability of no overflow from a given bucket is

$$\Pr(Q_i = 0) = e^{b\alpha} T_{b-1}(b\alpha).$$

This is the asymptotic probability of success in the parking problem, as $m, n \to \infty$ with $n/m \to \alpha$,

The parking problem.

Corollary

For $k = 0, \ldots, b-1$,

$$\Pr(Y_i = k) = \Pr(H_i = k) = -b(1-\alpha)\frac{[z^k]\prod_{\ell=0}^{b-1}(z - T(\omega^\ell \alpha e^{-\alpha})/\alpha)}{\prod_{\ell=1}^{b-1}(1 - T(\omega^\ell \alpha e^{-\alpha})/\alpha)}.$$

Furthermore, the probability that a bucket is not full is given by

$$\Pr(Y_i < b) = \Pr(H_i < b) = T_0(b\alpha) = \frac{b(1-\alpha)}{\prod_{\ell=1}^{b-1}(1 - T(\omega^\ell \alpha e^{-\alpha})/\alpha)}$$

and thus

$$\Pr(Y_i = b) = \Pr(H_i \geq b) = 1 - T_0(b\alpha).$$

Robin Hood: an example (b=2).

Keys inserted:

- 36, 77, 24, 79, 56, 69, 49, 18, 38, 97, 78, 10, 58.

Hash function

- $h(x) = x \bmod 10$.

a

49	79			24		36	77	18	58
69	10					56	97	38	78
0	1	2	3	4	5	6	7	8	9

- What happens when 29 is inserted?

a

29	69	10		24		36	77	18	58
49	79					56	97	38	78
0	1	2	3	4	5	6	7	8	9

Properties of Robin Hood Hashing.

a

29	69	10		24		36	77	18	58
49	79					56	97	38	78
0	1	2	3	4	5	6	7	8	9

- At least one record is in its home bucket.
- The keys are stored in nondecreasing order by hash value, starting at some location k and wrapping around. In our example, $k = 5$ (the first slot of the third bucket).
- If a fixed rule is used to break ties among the candidates to probe their next probe bucket (eg: by sorting these keys in increasing order), then the resulting table is independent of the order in which the records were inserted. Then, we may insert the elements in any order, and study the behavior of the last one inserted!.

Robin Hood displacement.

- W.l.o.g. we search for a record that hashes to bucket 0.
- We have to probe buckets occupied by the elements that would have gone to the overflow area.
- Then consider collisions with all the elements that hash to 0.
- Let D^{RH} be the displacement of a given element x.
- Let C^{RH} be the number of elements that win over x in the competition for slots in the buckets. Then $D^{\mathrm{RH}} = \lfloor C^{\mathrm{RH}}/b \rfloor$.
- The specification is

$$C^{\mathrm{RH}} = Q_{-1} + V = \mathrm{Overflow} * \mathrm{Mark}(\mathrm{Bucket})$$

  of the number $Q_{-1} = Q_{m-1}$ of keys that overflow into 0 and the number V of keys that hash to 0 that win over x.
- The number Q_{m-1} of keys that overflow does not change when the keys that hash to 0 are removed. This is thus independent of V.

Robin Hood displacement.

- We consider the displacement D^{RH} of a marked key $\bullet$.

$$RH(z,w,q) := \sum_{m\geq 0}\sum_{n\geq 0}\sum_{k\geq 0} CRH_{m,n,k} w^{bm}\frac{z^n}{n!}q^k,$$

where $CRH_{m,n,k}$ is the number of hash tables of length m with n keys (one of them marked as $\bullet$) such that $\bullet$ hashes to the first bucket and the displacement D^{RH} of $\bullet$ equals k.

Theorem

$$RH(bz,w,q) = \frac{1}{b}\sum_{d=0}^{b-1} C\left(bz,w,\omega^d q^{1/b}\right)\sum_{p=0}^{b-1}\left(\omega^d q^{1/b}\right)^{-p},$$

with

$$C(bz,w,q) = \frac{w^b(e^{bz}-e^{bzq})}{(1-q)(q^b-w^b e^{bzq})}\frac{\prod_{j=0}^{b-1}\left(q-\frac{T(\omega^j zw)}{z}\right)}{\prod_{j=0}^{b-1}\left(1-\frac{T(\omega^j zw)}{z}\right)}.$$

Robin Hood displacement.

- Note that the expectation of the displacement (but not the variance) is the same for any insertion heuristic. We let $D_{m,n}$ denote the displacement of a random element in a hash table with m buckets and n keys.

Lemma

For linear probing with the Robin Hood, FCFS or LCFS (or any other) heuristic,

$$\mathbb{E}\, D_{m,n} = \frac{m}{n}\, \mathbb{E}\, Q_{m,n}.$$

Proof.

For any hash table, and any linear probing insertion policy, the sum of the n displacements of the keys equals the sum of the m overflows Q_i. Take the expectation. □

Robing Hood displacement.

Theorem

Let $0 < \alpha < 1$. In the infinite Poisson model, the variable V_α, the number of keys that win over the new key C_α^{RH} and its Robin Hood displacement D_α^{RH} have the probability generating functions

$$\psi_V(q) = \frac{1 - e^{b\alpha(q-1)}}{b\alpha(1-q)}$$

$$\psi_C(q) = \psi_Q(q)\psi_V(q) = \frac{1-\alpha}{\alpha}\,\frac{1 - e^{b\alpha(q-1)}}{e^{b\alpha(q-1)} - q^b}\,\frac{\prod_{\ell=1}^{b-1}(q-\zeta_\ell)}{\prod_{\ell=1}^{b-1}(1-\zeta_\ell)}.$$

$$\psi_{\mathrm{RH}}(q) = \frac{1}{b}\sum_{j=0}^{b-1}\psi_C(\omega^j q^{1/b})\,\frac{1-q^{-1}}{1-\omega^{-j}q^{-1/b}}.$$

As $m, n \to \infty$ with $n/bm \to \alpha$, $V_{m,n} \xrightarrow{\mathrm{d}} V_\alpha$, $C_{m,n}^{\mathrm{RH}} \xrightarrow{\mathrm{d}} C_\alpha^{\mathrm{RH}}$ and $D_{m,n}^{\mathrm{RH}} \xrightarrow{\mathrm{d}} D_\alpha^{\mathrm{RH}}$, with convergence of all moments; furthermore, for some $\delta > 0$, the corresponding probability generating functions converge, uniformly for $|q| \le 1 + \delta$.

Robing Hood displacement.

Corollary

As $m, n \to \infty$ *with* $n/bm \to \alpha \in (0,1)$,

$$\mathbb{E}\, C_{m,n}^{\mathsf{RH}} \to \mathbb{E}\, C_{\alpha}^{\mathsf{RH}} = \frac{1}{2(1-\alpha)} - \frac{b}{2} + \sum_{\ell=1}^{b-1} \frac{1}{1-\zeta_\ell},$$

$$\mathbb{E}\, D_{m,n}^{\mathsf{RH}} \to \mathbb{E}\, D_{\alpha}^{\mathsf{RH}} = \frac{1}{2b\alpha}\left(\frac{1}{1-\alpha} - b - b\alpha\right) + \frac{1}{b\alpha}\sum_{\ell=1}^{b-1} \frac{1}{1-\zeta_\ell}.$$

with

$$\zeta_\ell := T\left(\omega^\ell \alpha e^{-\alpha}\right)/\alpha.$$

Block length.

- Let F_{bi+d} be the number of ways to construct an almost full table of length $i+1$ and size $bi+d$ with $0 \le d \le b-1$. Then,

$$F_d(u) := \sum_{i \ge 0} F_{bi+d} \frac{u^{bi+d}}{(bi+d)!}, \quad N_0(z,w) := \sum_{d=0}^{b-1} w^{b-d} F_d(zw).$$

- In an almost full table the length of the block is marked by w in $N_0(bz,w)$.
- The generating function $B(z,w,q)$ for the block length is

$$B(bz,w,q) = \Lambda_0(bz,w) N_0(bz, wq^{1/b}) = \frac{1 - \prod_{j=0}^{b-1} \left(1 - \frac{T(\omega^j z w q^{1/b})}{z}\right)}{\prod_{j=0}^{b-1} \left(1 - \frac{T(\omega^j z w)}{z}\right)}.$$

Block length.

Let $B_{m,n}$ be the length of a random block, chosen uniformly among all blocks in all hash tables with m buckets and n keys. This is the same as the length of the last block in a uniformly random hash table such that the rightmost bucket is not full. Recall that we denote the number of such hash tables by $Q_{m,n,0}$.

Corollary

If $0 \leq n < bm$, *then*

$$\mathbb{E}\, B_{m,n} = \frac{m^n}{Q_{m,n,0}}.$$

Proof.

The sum of the block lengths in any hash table is m, and thus the sum of the lengths of all blocks in all tables is $m \cdot m^n$, while the number of blocks ending with a given bucket is $Q_{m,n,0}$ and thus the total number of blocks is $m \cdot Q_{m,n,0}$. □

Block length.

- Let B be the length of the first block, i.e.,

$$B := \min\{i \geq 1 : Y_i < b\} = \min\{i \geq 1 : H_i < b\}.$$

- Hence, B is the first positive index i such that the number of elements $S_i = X_1 + \cdots + X_i$ hashed to the i first buckets is less than the capacity bi of these buckets, i.e.,

$$B = \min\{i \geq 1 : S_i < bi\}.$$

Theorem

The probability generating function $\psi_B(z) := \mathbb{E}\, z^B$ of B is given by

$$\psi_B(z) = 1 - \prod_{\ell=0}^{b-1} \left(1 - T(\omega^\ell \alpha e^{-\alpha} z^{1/b})/\alpha\right),$$

for $|q| \leq R$ for some $R > 1$.

Block length.

Corollary

The random block length $B = B_\alpha$ defined above has expectation

$$\mathbb{E}\, B_\alpha = \frac{1}{T_0(b\alpha)}$$

and variance,

$$\mathbb{V}[B]_\alpha = \frac{1}{b(1-\alpha)^2 T_0(b\alpha)} - \frac{2}{bT_0(b\alpha)} \sum_{\ell=1}^{b-1} \frac{\zeta_\ell}{(1-\zeta_\ell)(1-\alpha\zeta_\ell)} - \frac{1}{T_0(b\alpha)^2}.$$

Block length.

- The length of the block $\hat{B}_i$ containing a given bucket i has a different, size-biased distribution.

Theorem

In the infinite Poisson model, $\hat{B} = \hat{B}_\alpha$ has the size-biased distribution

$$\Pr(\hat{B}_\alpha = k) = \frac{k \Pr(B_\alpha = k)}{\mathbb{E}\, B_\alpha} = T_0(b\alpha) k \Pr(B_\alpha = k)$$

and thus the probability generating function

$$\psi_{\hat{B}}(q) = T_0(b\alpha) q \psi'_B(q) = T_0(b\alpha) q \sum_{\ell=0}^{b-1} \zeta'_\ell(q) \prod_{j \neq \ell} (1 - \zeta_j(q)).$$

As $m, n \to \infty$ with $n/bm \to \alpha$, $\hat{B}_{m,n} \xrightarrow{\mathrm{d}} \hat{B}_\alpha$ with convergence of all moments; furthermore, for some $\delta > 0$, the probability generating function converges to $\psi_{\hat{B}}(q)$, uniformly for $|q| \leq 1 + \delta$.

Block length.

Corollary

As $m, n \to \infty$ *with* $n/bm \to \alpha \in (0,1)$,

$$\mathbb{E}\,\hat{B}_{m,n} \to \mathbb{E}\,\hat{B}_\alpha = \frac{1}{b(1-\alpha)^2} - \frac{2}{b}\sum_{\ell=1}^{b-1} \frac{\zeta_\ell}{(1-\zeta_\ell)(1-\alpha\zeta_\ell)}.$$

Theorem

In the exact model, $\hat{B}_{m,n}$ *has the size-biased distribution*

$$\Pr(\hat{B}_{m,n} = k) = \frac{k \Pr(B_{m,n} = k)}{\mathbb{E}\,B_{m,n}} = \frac{Q_{m,n,0}}{m^n} k \Pr(B_{m,n} = k).$$

Theorem

For the exact model, as $m, n \to \infty$ *with* $n/bm \to \alpha$, $B_{m,n} \xrightarrow{\mathrm{d}} B_\alpha$ *with convergence of all moments.*

Unsuccessful search.

- In a cluster with n keys, the number of visited buckets in a unsuccessful search, is the same as the one needed to insert the $(n+1)$st element.
- Then, the specification $\mathrm{Pos}(\mathcal{C})$ (marking the position of this inserted element) leads to

$$\begin{aligned} U(bz,w,q) &= \sum_{m\geq 1} w^{bm} \sum_{n\geq 0} \frac{(bmz)^n}{n!} P_{m,n}(q) \\ &= \Lambda_0(bz,w)\frac{N_0(bz,w)-N_0(bz,wq^{1/b})}{1-q} \\ &= \frac{\prod_{j=0}^{b-1}\left(1-\frac{T(\omega^j z w q^{1/b})}{z}\right)-\prod_{j=0}^{b-1}\left(1-\frac{T(\omega^j z w)}{z}\right)}{(1-q)\prod_{j=0}^{b-1}\left(1-\frac{T(\omega^j z w)}{z}\right)}, \end{aligned}$$

where $P_{m,n}(q)$ is the probability generating function for the displacement of the $(n+1)$st inserted element.

FCFS displacement.

- We consider the displacement of a marked key $\bullet$, which we by symmetry may assume hashes to the first bucket.
- Thus, let

$$FCFS(z, w, q) := \sum_{m \geq 1} \sum_{n \geq 1} \sum_{k \geq 0} FCFS_{m,n,k} w^{bm} \frac{z^n}{n!} q^k,$$

  where $FCFS_{m,n,k}$ is the number of hash tables of length m with n keys (one of them marked as $\bullet$) such that $\bullet$ hashes to the first bucket and the displacement D^{FC} of $\bullet$ equals k.
- For a given m and n with $1 \leq n \leq bm$, there are nm^{n-1} such tables (n choices to select $\bullet$ and m^{n-1} choices to place the other $n - 1$ elements).

FCFS displacement.

- Thus, if $d_{m,n}(q)$ is the probability generating function for the displacement of a random key

$$FCFS(z,w,q) = \sum_{m\geq 1}\sum_{n=1}^{bm} nm^{n-1}d_{m,n}(q)w^{bm}\frac{z^n}{n!}$$
$$= z\sum_{m\geq 1}\sum_{n=0}^{bm-1} d_{m,n+1}(q)w^{bm}\frac{m^n z^n}{n!}.$$

Theorem

$$FCFS(bz,w,q) = b\int_0^z U(bt, we^{z-t}, q)\,dt$$

up to terms $z^n w^m q^k$ with $n > bm$.

FCFS displacement.

Proof.

The probability generating function for the displacement of a random key when having n keys in the table is

$$d_{m,n}(q) = \frac{1}{n} \sum_{i=0}^{n-1} u_{m,i}(q),$$

Then, for all $m \geq 1$ and $n \geq 0$,

$$u_{m,n}(q) = (n+1)d_{m,n+1}(q) - nd_{m,n}(q),$$

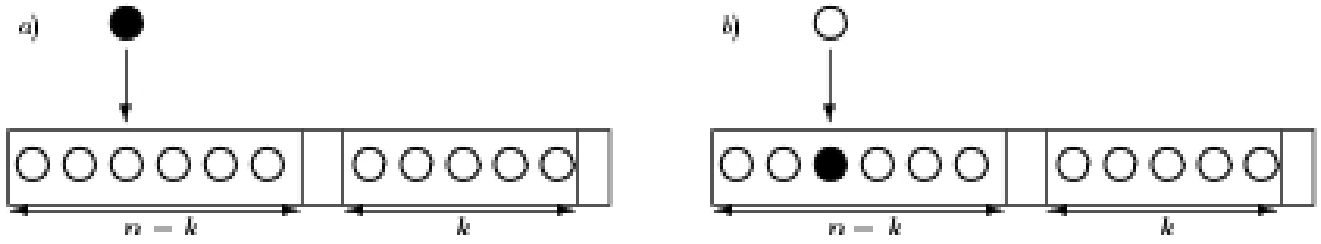
and so,

$$\frac{\partial}{\partial z} FCFS(bz, w, q) - w \frac{\partial}{\partial w} FCFS(bz, w, q) = bU(bz, w, q).$$

This differential equation together with the boundary condition $F(0, w, q) = 0$ leads to the solution. □

Specification for FCFS displacement (b=1).

- Let $FC(z, w, q)$ be the generating function for the cost of a successful search in an almost full table when $n+1$ elements are inserted and one element $\bullet$ is marked.
- $AF(z) = \frac{T(z)}{z}$ where $T(z)$ is the tree function.



- Then, <FC> = <Add(AF)>*<AF> + <Pos(FC)>*<AF> + <Pos(AF)>*<FC> leads to

$$\partial_z(FC) = H[AF] * AF + \frac{\partial}{\partial z}(zFC) * AF + \frac{\partial}{\partial z}(zAF) * FC.$$

$$FCFS(z, w, q) = \frac{((1 - T(zwq))^2 - (1 - T(zw))^2)T(zw)}{2z(1-q)(1-T(zw))}.$$

Unsuccessful search and FCFS displacement.

Theorem

The probability generating function $\psi_U(z) := \mathbb{E}\, z^{U_i}$ *of* U_i *is given by*

$$\psi_U(z) = \frac{T_0(b\alpha)}{1-z} \prod_{\ell=0}^{b-1} \left(1 - T\left(\omega^\ell \alpha e^{-\alpha} z^{1/b}\right)/\alpha\right).$$

Theorem

The probability generating function $\psi_{\mathsf{FC}}(z) := \mathbb{E}\, z^{D_i^{\mathsf{FC}}}$ *of* D_i^{FC} *is given by*

$$\psi_{\mathsf{FC}}(z;\alpha) = \frac{1}{\alpha}\int_0^\alpha \psi_U(z;\beta)\,\mathrm{d}\beta = \frac{1}{\alpha}\int_0^\alpha \frac{\tau(\beta)}{1-z}\prod_{\ell=0}^{b-1}(1-\zeta_\ell(z;\beta))\,\mathrm{d}\beta$$

$$= \frac{1}{\alpha}\int_0^\alpha \frac{b(1-\beta)\prod_{\ell=0}^{b-1}(1-\zeta_\ell(z;\beta))}{(1-z)\prod_{\ell=1}^{b-1}(1-\zeta_\ell(1;\beta))}\,\mathrm{d}\beta.$$

Some final considerations.

- Problem with a very rich history.
- Paradigm of a problem that nicely integrates analytical, combinatorial and probabilistic approaches.
- This integration (together with the use of symbolic methods!) has allowed the understanding of deep relations with other important problem.
- A unified analysis of several important random variables related with linear probing: **symbolic methods + random walks (linked by the Poisson Transform)**.

Some ongoing work.

- Total displacement with buckets. Relation with other problems as for b=1?
- Number of movements in deletion algorithm.