Computing integrated costs of sequences of operations with application to dictionaries

Philippe Flajolet, Jean Vuillemin, Jean Francon

IRIA
laboria

Institut de Recherche
d'Informatique
et d'Automatique

Domaine de Voluceau
Rocquencourt
B. P. 105 78150 - Le Chesnay
France
Tél.: 954 90 20

laboratoire de recherche
en informatique
et automatique

COMPUTING INTEGRATED COSTS OF SEQUENCES OF OPERATIONS WITH APPLICATION TO DICTIONARIES

14p.

Philippe FLAJOLET
Jean FRANCON
Jean VUILLEMIN

Rapport de Recherche N° 346

Février 1979

COMPUTING INTEGRATED COSTS OF SEQUENCES OF OPERATIONS WITH APPLICATION TO DICTIONARIES

Philippe Flajolet, Jean Françon, Jean Vuillemin

Résumé :

Nous introduisons une notion de coût intégré des structures de dictionnaires. Ce coût est défini comme le coût moyen d'une suite d'adjonctions, de suppressions et de consultations. Nous donnons des expressions sous forme de fractions continues des séries génératrices et nous obtenons une représentation intégrale explicite des coûts intégrés de trois implantations usuelles de la structure de dictionnaire.

Abstract :

We introduce a notion of integrated cost of a dictionary, as average cost of sequences of search, insert and delete operations. We express generating functions of these sequences in terms of continued fractions ; from this we derive an explicit integral expression of integrated costs for three common reprsentations of dictionaries.

COMPUTING INTEGRATED COSTS OF SEQUENCES OF OPERATIONS

WITH APPLICATION TO DICTIONARIES

Philippe Flajolet, Jean Françon*, Jean Vuillemin**

IRIA - Laboria

78150 Le Chesnay (France)

Article présenté à la 11è Conférence de l'ACM "Symposium on Theory of Computing" (1979).

I - BACKGROUND

I.1) Data Organisations

A data organisation is a class of data structures, together with a (finite) collection of algorithms manipulating such data. Various data organisations exist for representing :

- dictionaries where the basic operations are
  - A for adjunction, i-e, insertion ;
  - S for suppression, i-e, deletion ;
  - Q^+ for positive query, i-e successfull search ;
  - Q^- for negative query, i-e unsuccessfull search ;
- partitions where the basic operations on a collection of disjoint sets are search and union.
- priority queues, etc... .

For example, dictionaries can be represented by : Unsorted linear lists (ULL), Sorted linear lists (SLL), Binary search trees (BST), (see [Knu. 73] for a description).

I.2) Cost of sequences of operations

In comparing the cost (with respect to some measure like time, space, OS charge,...) of two data organisations A and B representing a dictionary, we cannot merely compare the cost of individual operations on data of a given size : A may be better than B on some data, and conversely on others ; operation 1 may be more efficient in A whereas operation 2 is more efficient in B.

---

* Centre de Calcul du CNRS, B.P. 20 CRO 67037 Strasbourg (France).

** Université de Paris-Sud, Informatique, Bâtiment 490 91405 Orsay (France).

A reasonable way to measure the efficiency of a data organisation is to consider sequences of operations on the structure ; the cost of each such sequence is the sum of the costs of individual operations. Given a finite set of sequences of fixed length, we can define the maximal, minimal and average cost over the set in an obvious way.

I.3) Previous work

This general problem has already generated interesting work pertaining to the cost of sequences of operations for partition structures, where both maximal cost ([Fis. 72], [A.,H.,U. 74], [Tar. 75]) and average cost ([D.,R. 76], [Yao. 76], [K.,S. 77]) have been studied.

About dictionaries, the "Knott paradox" [Kno. 75] for binary search trees ([Knu. 73], [Hib. 62]) has stimulated a general methodological reflexion by [Knu. 77] and a difficult mathematical analysis [J.,K. 78]. Interesting results about maximal cost have been obtained in [Sny. 77] under hypotheses very similar to ours (see section 2).

A more recent work [Fra. 78], motivated by combinatorial results of [F.,V. 78], has made general methodological proposals for defining the integrated cost of data organizations and has obtained explicit results for specific dictionary representations. The richness and interest of the structures studied by [Fra. 78] is further uncovered by [Fla. 78] who provides an expression in terms of continued fraction for the generating function of sequences of operations, under very general hypotheses.

I.4) Contributions of this paper

In this paper, we build upon the work of [Fra. 78] and [Fla. 78]. Using continued fractions provides general algebraic expressions of integrated costs for data organizations. We then specialize this general expression to the case of dictionaries ; we show that in terms of generating functions, the passage from unitary costs of operations to integrated cost over sequences of operations is expressed by a linear integral transform.

As a showcase, we make complete, à la Knuth ([Knu. 68]) analyses of the integrated cost of dictionaries represented by unsorted linear lists, sorted linear lists and binary search trees, under arbitrary sequences of operations A, S, Q^+ and Q^-.

It is now clear that studying sequences of operations on arbitrary data organisations raises rich and interesting questions related to the fields of :

A) continued fractions, orthogonal polynomials, special functions ;

B) combinatorial study of permutations, trees, and plane paths ;

C) analysis of algorithms, definition of relevant complexity measures over sequences of operations.

A more complete attempt at discussing these issues is made by [F.,F.,V.,V. 79]. In the present work, we content ourselves with describing the aspects or the general theory which are relevant to computing integrated costs of dictionaries. It is nevertheless clear that the techniques introduced here have a theoretical and practical interest that goes beyond that of completely analysing toy examples.

II) INTEGRATED COSTS OF DICTIONARIES

The set K of keys (real , alphanumerical,...) manipulated by our dictionaries is assumed to be infinite and totally ordered.

A sequence of operations is a sequence of the form $O_1(k_1);O_2(k_2);\ldots;O_n(k_n)$ where $n\geq 0$, and, for each $1\leq i\leq n$, $k_i \in K$ and $O_i \in \{A,S,Q^+,Q^-\}$.

Informally, a legal sequence of operation is one where operations A and Q^- are performed on keys which are not yet in the structure, while S and Q^+ must only be performed on keys in the structure.

The content F_i of the structure (or file) at time (stage) i, $1\leq i\leq n$, is defined by :

1) $F_0 = \emptyset$;

2) $F_{i+1} = F_i$ if ($k_i \in F_i$ $O_i = Q^+$) or ($k_i \notin F_i$ $O_i = Q^-$);
3) $F_{i+1} = F_i \cup \{k_i\}$ if ($k_i \notin F_i$ $O_i = A$);
4) $F_{i+1} = F_i \backslash k_i$ if ($k_i \in F_i$ $O_i = S$);
5) F_{i+1} = undefined in all the other cases.

The sequence is then legal if F_{n+1} is well defined by the above rules.

For example, (the keys being rational numbers) A(1.4);A(3.1);Q^-(1.7);A(0.5);Q^+(3.1);S(0.5);Q^+(0.5);S(3.1); S(0.5) is a legal sequence of operations and the associated file contents are : $\emptyset$, {1.4}, {1.4,3.1}, {1.4,3.1}, {0.5,1.4,3.1}, {0.5,1.4,3.1}, {0.5,3.1}, {0.5,3.1}, {0.5}, $\emptyset$.

II.1) Histories

If $F \subset K$ is a finite set $F = \{k_1,\ldots,k_p\}$ of keys, canonically numbered $1,\ldots,p$ in increasing order (according to the ordering on K), we define the rank *rank*(k,F) of a key $k \in K$ in F by :

1) if $k \in F$ and $k = k_i$ then *rank*$(k,F) = i-1$.
2) if $k \notin F$ and $k_i < k < k_{i+1}$ then *rank*$(k,F) = i$ (with the convention that $k_0 < k < k_{p+1}$ for all $k \in K$).

To each legal sequence $L = O_1(k_1);\ldots;O_n(k_n)$ we associate a history which is a pair $h = (S,V)$ where $S = O_1;O_2;\ldots;O_n$ is the schema of L and h, and $V = \text{rank}(k_1,F_0);\ldots;\text{rank}(k_i,F_{i-1});\ldots;\text{rank}(k_n,F_{n-1})$ is the valuation of L and h ; the integer n is the length of k. By convention there exists one (empty) history of length 0.

The history associated with the legal sequence given above has schema A A Q^- A Q^+ S Q^+ S S and valuation 0 1 1 0 2 1 0 1 0.

We define the height h_i of a schema $O_1;\ldots;O_n$ at time (stage) i, $0 \le i \le n$ by $h_i = |F_i|$; the valuation $V_i;\ldots;V_n$ associated to schema $O_1;\ldots;O_n$ is a sequence of integers such that $0 \le V_i < \text{pos}(O_i,h_{i-1})$ where $\text{pos}(A,h) = \text{pos}(Q^-,h) = h+1$ and $\text{pos}(S,h) = \text{pos}(Q^+,h) = h$ are the number of possible outcomes of each operation on a file of size h.

Histories are thus combinatorial objects, and there are finitely many histories of a given length.

Lemma 1 : *The number of histories with a given schema* $O_1;O_2;\ldots;O_n$ *is equal to* $\prod_{1 \le i \le n} \text{pos}(O_i,h_{i-1})$.

For example, the set of histories of lenth 2 contains 2 schemas Q^-Q^-, Q^-A, AS, AQ^-, AQ^+, AA corresponding to the 8 histories

Q^-Q^-	Q^-A	AS	AQ^-	AQ^-	AQ^+	AA	AA.
0 0	0 0	00	00	01	00	00	01

We denote by : 1) H_n the set of histories of length n, and of initial and final height $h_0 = h_n = 0$; our main interest lies in these histories.

2) $H_{k,\ell,n}$ the set of histories of length n, of initial height $h_0 = k$ and final height $h_n = \ell$.

Of course, $H_n = H_{0,0,n}$ and we denote the cardinalities of these sets by $H_n = |H_n|$ and $H_{k,\ell,n} = |H_{k,\ell,n}|$. By inspection, we check that $H_1 = 1$, $H_2 = 2$ and $H_3 = 6$. By convention $H_{k,\ell,0} = \delta_{k,\ell}$.

The level crossing numbers $N_{k,n}$ represent, among all histories $g = \begin{matrix} O_1\ldots O_n \\ V_1\ldots V_n \end{matrix}$ in H_n, the total number of indices i such that $0 \le i \le n$ and $h_i = k$.

For example, $N_{0,3} = 16$, $N_{1,3} = 8$, $N_{2,3} = 0$ by inspection and we check the obvious relation $\sum_k N_{k,n} = (n+1)\, H_{0,0,n}$. It is not difficult to prove the property $N_{k,n} = \sum_i H_{0,k,i} \cdot H_{k,0,n-i}$.

We also introduce the other level crossing numbers $NO_{k,n}$ for each operation $O \in \{A,S,Q^+,Q^-\}$ as the number of indices i such that $0 \le i \le n$, $h_i = k$ and $O_i = O$; we thus have four types of numbers : $NA_{k,n}$, $NS_{k,n}$, $NQ^+_{n,n}$ and $NQ^-_{k,n}$. By inspection $NQ^-_{0,3} = 5$, $NA_{0,3} = 5$, $NQ^-_{1,3} = 2$, $NQ^+_{1,3} = 1$, $NS_{1,3} = 5$. Again we check that

$$\sum_{O,k} NO_{k,n} = n.H_{O,O,n}.$$

II.2) Integrated costs over H_n.

To each legal sequence of operations $O_1(k_1);\ldots;O_n(k_n)$ and specific data organisation we associate a cost which is the sum of the costs (supposed previously defined) of the specified sequence of algorithm on the keys.

In this paper, we restrict attention to dictionaries for which this cost depends solely upon the underlying history. If we call equivalent two legal sequences of operations having the same history, we require that the cost of executing two equivalent legal sequences of operations in the same (see "equivalent request sequences" in [Sny. 77]).

This is the case in a large class of dictionary representations, including our three structures ULL, SLL, BST and all balanced tree representations known to the authors. Such a result can be established in a rather general setting under the following hypothesis :

1) Decision Tree Hypothesis (H_1) : The only way in which the data organisation can access the keys is by performing comparisons between keys.
2) Oblivion Hypothesis (H_2) : Only keys which are present in the structure (i.e. have not been deleted) can be compared.

Hypothesis 1) excludes from consideration dictionaries based on h-code or digital search that perform arithmetics on the keys.

Hypothesis 2) excludes data organisation with some kind of garbage collection that may mark deleted elements instead of actually removing them from the data structure.

Under these hypothesis, executions of equivalent legal sequences of operations will in fact be isomorphic, in the strong sense that the same sequence of machine instructions will be executed (possibly manipulating different but order-isomorphic data).

For dictionary representations that meet these requirements, our goal is to compute the integrated cost K_n over H_n defined by $K_n = \sum_{h \in H_n} \text{cost}(h)$, where cost(h) is defined as the cost of executing any of the legal operation sequences having h for history.

Comparing dictionary algorithms over the set of canonical histories is the analogue of what we do when we reduce the analysis of a sorting algorithm over an infinite set of keys to an analysis over the n! permutations of $\{1,2,\ldots,n\}$.

II.3) Expression of the integrated cost for stationary structures

Computing the integrated cost K_n terms out to be possible if one can replace in K_n the cost of any operation $O \in \{A,S,Q^+,Q^-\}$ operating on a file of size k by a function CO_k depending on the integer k only. Thus we have the formula :

$$K_n = \sum_{k,O} NO_{k,n}\ CO_k.$$

We say that a data organisation is stationnary if this formula holds. It can be proved ([Fra. 78]) under randomness preservation hypotheses (essentially the (I_o,D_o) hypothesis in [Knu. 77]); in this case, the CO_k are the usual mean costs of operation O for the size k.

The three structures ULL, SLL and BST are stationnary ; the only non trivial part in proving these facts resides in Hibbard's theorem for deletion in BST ([Knu. 73], [Hib. 62], [Kno. 75]). Binomial lists ([Knu. 73] p. 169 in "On line Merge Set") are stationnary, but none of the known balanced-tree algorithms has this property.

Using Knuth [73] as a source, we have collected in Tables 1 and 2 the average costs of operations in our three structures, measured in number of comparisons and MIX time.

	ULL	SLL	BST
Q^+	$\frac{k+1}{2}$	$\frac{k+1}{2}$	$2(1+\frac{1}{k})\,H_k - 3$ $(k\geq 1)$
Q^-	k	$\frac{k+2}{2}$	$2(H_{k+1}-1)$
A	0	$\frac{k+2}{2}$	$2(H_{k+1}-1)$
S	k	$\frac{k+1}{2}$	$2(1+\frac{1}{k})\,H_k - 3$ $(k\geq 1)$

Table 1
Average number of comparisons per operation on a file of size k.

	ULL	SLL	BST
Q^+	$3k+4$	$3k+5$	$15\,H_k - 21 + \frac{15}{k}\,H_k$ $(k\geq 1)$
Q^-	$6k+4$	$3k+8$	$15\,H_{k+1} - 11$
A	11	$3k+19$	$15\,H_{k+1} + 5$
S	$9k+17$	$4.5k+16.5$	$19\,H_k - 13.5 + \frac{13}{k}\,H_k + \frac{3}{k}$ $(k\geq 1)$

Table 2
Average number of MIX time units per operation on a file of size k.

(In these tables H_k denotes the k-th harmonic number : $H_k = 1 + \frac{1}{2} + \ldots + \frac{1}{k}$).

In section 5, we shall compute the integrated costs K_n of these three structures.

III. CONTINUED FRACTIONS AND DICTIONARIES

In the last section, we have reduced the computation of integrated costs to the evaluation of certain combinatorial sums in which appear the quantities H_n, $H_{k,\ell,n}$, $N_{k,n}$, etc... . In this section we express the generating functions relative to these quantities in terms of continued fractions. The result is not specific to dictionaries and applies to a wide class of data organizations including priority queues, linear lists and stacks. It leads in each case to a particular class of orthogonal polynomials : the Laguerre, Hermite, Mittag-Leffler and Fibonacci polynomials ([Fla. 78]).

III.1) Histories and Continued fractions

We consider schemas as defined in section 2 and we introduce an arbitrary possibility set Π : $pos(A;k) = \alpha_k$; $pos(Q^+;k) = \kappa_k^+$; $pos(Q^-;k) = \kappa_k^-$; $pos(S;k) = \sigma_k$. We also let κ_k denote $\kappa_k^+ + \kappa_k^-$. Dictionaries thus correspond to the particular case : $\alpha_k = k+1$; $K_k = 2k+1$; $\sigma_k = k$. The following result is from [Fla. 78] :

Theorem F : (The Continued Fraction expansion theorem) :

Let H_n be the number of histories ending at zero, relative to the possibility set Π, and let $H(z) = \sum_{n\geq 0} H_n z^n$ be the corresponding generating function. Then $H(z)$ has the following continued fraction expansion :

$$H(z) = \cfrac{1}{1 - \kappa_0 z - \cfrac{\alpha_0\sigma_1 z^2}{1 - \kappa_1 z - \cfrac{\alpha_1\sigma_2 z^2}{\ddots}}}$$

Proof (sketch). Define the alphabet $X = \{a_0, a_1, \ldots, q_0, q_1, \ldots, s_1, s_2, \ldots\}$ where ω_j (ω = a,q or s) denotes operation ω on a file of size j. Let $S^{[h]}$ denote the set of schemas represented by words over X having height $\leq h$. The $S^{[h]}$ have the following regular expression descriptions :

$$S^{[0]} = (q_0)^* \quad ; \quad S^{[1]} = (q_0 + a_0\,(q_1)^*\,s_1)^* \quad ; \quad S^{[2]} = (q_0 + a_0(q_1 + a_1(q_2)^*\,s_2)^*\,s_1)^* \ldots,$$

and in general $S^{[h+1]}$ is obtained by substituting $(q_h + a_h(q_{h+1})^*\,s_{h+1})$ for q_h in the expression of $S^{[h]}$. It follows from classical results in the theory of formal series in non-commuting variables, due to Schützenberger, that characteristic series can be obtained from set descriptions by replacing : Union

by sum ; catenation by product ; Kleene star by quasi-inverse. This is valid provided certain non-ambiguity conditions are satisfied. These conditions do hold here and using the notation X(E) for the characteristic series of set E, we have :

$$\chi(S^{[h]}) = \cfrac{1}{1 - q_0 - \cfrac{a_0 \mid}{1 - q_1 - \cfrac{a_1 \mid s_2}{\cdots \cfrac{}{1 - q_h}}}} \qquad \text{, with } \frac{u \mid w}{v} = u(1-v)^{-1}w.$$

Similarly, with $S = \bigsqcup_h S^{[h]}$, we have the continued fraction expansion :

$$\chi(S) = \cfrac{1}{1 - q_0 - \cfrac{a_0 \mid s_1}{1 - q_1 - \cfrac{a_1 \mid s_2}{\cdots}}}$$

Histories relative to the possibility set Π can be similarly represented by words over the alphabet $Y = \{\omega_j^{(i)}\}$ with $\omega \in \{a,q,s\}$ and $i,j \in N$. Here $\omega_j^{(i)}$ denotes the i-th possibility of operation ω on a file of size j. With $H^{[h]}$ the set of histories of height $\leq h$, the expression of $\chi(H^{[h]})$ follows from that of $\chi(S^{[h]})$ by the formal substitution $\omega_j \to \sum_i \omega_j^{(i)}$ where $0 \leq i \leq pos(\omega,j)$.

Thus with $H = \bigsqcup_h H^{[h]}$, we have the identity : $\chi(H) = \cfrac{1}{1 - (\sum q_0^{(i)} - \cfrac{\sum a_0^{(i)} \mid \sum s_1^{(i)}}{1 - \sum q_1^{(i)} - \cfrac{\sum a_1^{(i)} \mid \sum a_2^{(i)}}{\cdots}})}$

where index ranges are defined by the possibility set Π. The theorem follows by replacing all variables of Y by the single variable z. □

These fractions are known as Jacobi-type fractions or J-fractions ; when all the κ_j's are zero, they are called Stieltjes-type continued fraction or S-fractions. ([Stie ; 1894])

With $H_n^{[h]}$ the number of histories of height $\leq h$ and $H^{[h]}(z)$ the corresponding generating function, we have :

Proposition 2 : *Histories of height* $\leq h$ *have a rational generating function given by* $H^{[h]}(z) = \frac{P_h(z)}{Q_h(z)}$, *where* P_h *and* Q_h *are polynomials which satisfy the recurrences*

$$P_{-1}(z) = 0; P_0(z) = 1 \quad ; \quad P_h(z) = (1-q_h z)P_{h-1}(z) - a_{h-1}\, s_h\, z^2\, P_{h-2}(z)$$

$$Q_{-1}(z) = 1; Q_0(z) = 1-q_0 z \; ; \; Q_h(z) = (1-q_h z)Q_{h-1}(z) - a_{h-1}\, s_h\, z^2\, Q_{h-2}(z)$$

Hence $\deg P_h = \deg Q_{h-1} = h$ *for all* h.

The recurrence satisfied by the P and Q polynomials is the classical linear recurrence of denominators and numerators of convergents of continued fractions. The polynomials P and Q that appear here also play a role in the expression of generating functions of histories starting at level k and finishing at level l.

Proposition 3 : *Let* $H_{k,\ell}(z) = \sum_{n \geq 0} H_{k,\ell,n} z^n$; *we have :*

$$H_{k,\ell}(z) = \frac{Q_{\mu-1}(z)}{\alpha_0\alpha_1\cdots\alpha_{\kappa-1}\sigma_1\sigma_2\cdots\sigma_\kappa z^{\kappa+1}} \left(Q_{\lambda-1}(z)\, H(z) - P_{\lambda-1}(z)\right)$$

where $\mu = \min(k,\ell)$ *and* $\lambda = \max(k,\ell)$.

In particular this gives expressions for $H_{0,k}(z)$ *and* $H_{k,0}(z)$, *namely :*

$$H_{0,k}(z) = \frac{1}{\sigma_1\sigma_2\cdots\sigma_k z^k}\left(Q_{k-1}(z)\, H(z) - P_{k-1}(z)\right), \text{ and } H_{k,0}(z) = \frac{1}{\alpha_0\alpha_1\cdots\alpha_{k-1} z^k}\left(Q_{k-1}(z)\, H(z) - P_{k-1}(z)\right).$$

From this we can determine the generating functions of level crossings. For instance we have :

$$N_k(z) = \frac{1}{\alpha_0\alpha_1\cdots\alpha_{k-1}\sigma_1\sigma_2\cdots\sigma_k z^{2k}} \left(Q_{k-1}(z)\, H(z) - P_{k-1}(z)\right)^2.$$

III.3) Orthogonality

An alternative way of looking at the relations between the formal series H(z) and the polynomials $Q_h(z)$ which appear in the convergents is by means of orthogonality relations. Starting from the sequence of numbers H_n we define a scalar product over the set of polynomials by $\langle x^p | x^q \rangle = H_{p+q}$, for $p,q \geq 0$, and extend it by bilinearity to all polynomials. A classical result states (cf. for instance [Wall. 48]) :

Proposition 4 : *Let* $\overline{Q}_k(z)$ *be the reciprocal polynomial of* $Q_k(z)$, i.e. $\overline{Q}_k(z) = z^{k+1} Q_k(\frac{1}{z})$; *the following orthogonality relations hold :* $\langle x^l | \overline{Q}_{k-1}(x) \rangle = 0$ for $0 \leq l < k$, *and*

$$\langle x^k | \overline{Q}_{k-1}(x) \rangle = \alpha_0\alpha_1\cdots\alpha_{k-1}\sigma_1\sigma_2\cdots\sigma_k .$$

In other word the $\overline{Q}_k$ form an orthogonal basis w.r.t. the scalar product canonically associated to the sequence $\{H_n\}$. The relation with histories is expressed by :

Proposition 5 : *The number of histories finishing at level* k *and with length* n *is given by :*

$$H_{0,k,n} = \frac{1}{\sigma_1\sigma_2\cdots\sigma_k} \langle x^n | \overline{Q}_{k-1}(x) \rangle .$$

At this stage this only appears as the result of a purely algebraic manipulation. However once the measure with respect to which the $\overline{Q}_k$ are orthogonal has been determined (this is the celebrated moment problem), it provides one possible way of computing the $H_{0,k,n}$ and related quantities.

III.4) Dictionaries

We now specialise these results to the case where the possibility set Π corresponds to dictionaries : pos(A;k) = k+1 ; pos(Q;k) = 2k+1 ; pos(S;k) = k. Thus the generating series of dictionary histories is given in this case by :

$$H(z) = \cfrac{1}{1 - 1z - \cfrac{1^2 z^2}{1 - 3z - \cfrac{2^2 z^2}{1 - 5z - \cfrac{3^2 z^2}{\cdots}}}}$$

This continued fraction was studied by Euler in connection with formal solutions to the differential equation $z^2 H' + (z-1)H + 1 = 0$. It can also be put under the equivalent form :

$$H(z) = \cfrac{1}{1 - \cfrac{1z}{1 - \cfrac{2z}{1 - \cfrac{2z}{\cdots}}}}$$

with the sequence of coefficients (1,2,2,3,3,4,4,...) and as such appears as a limiting case of the continued fraction of Gauss. From this can be derived the remarkably simple expression $H_n = n!$. This result has been first proved by Françon and Viennot ([F.& V. 78] ; [Fra. 78]) who constructed an explicit bijection between histories and permutations.

The convergents of Euler's fraction involve, as denominators, polynomials Q_h whose generating function can be explicitely determined. Let

$$Q(s,t) = \sum_{j \geq 0} Q_{j-1}(z) \frac{t^j}{j!} ,$$

the linear recurrence on the Q_j's translates into a partial defferential equation on Q from which we derive

$$Q(z,t) = \frac{1}{1+tz} \exp \frac{t}{1+tz} .$$

The $\bar{Q}$ polynomials thus have a generating function $\bar{Q}(z,t) = \sum_{k\geq 0} \bar{Q}_{k-1}(z)\,\frac{t^k}{k!}$ equal to $\frac{1}{1+t}\exp\frac{zt}{1+t}$, from which follows the explicit expression :

$$\bar{Q}_{k-1}(z) = \sum_j (-1)^{k-j}\binom{k}{j}\frac{k!}{j!}\,x^j.$$

The $\bar{Q}$ polynomials are related to the classical Laguerre polynomials.

IV. INTEGRATED COSTS : THE INTEGRAL FORMULA

Section 3 provides algebraic tools which we now use for computing explicit expressions of the numbers $H_{0,k,n}$ of dictionary histories, and level crossing $N_{k,n}$.

IV.1) Counting histories and level crossings

Proposition 5 provides the expression $H_{0,k,n} = H_{k,0,n} = \frac{1}{k!}\,\langle x^n | \bar{Q}_{k-1}(x)\rangle$ where the polynomials $\bar{Q}$ have the generating function :

$$\bar{Q}(x,t) = \sum_{k\geq 0} \bar{Q}_{k-1}(x)\,\frac{t^k}{k!} = \frac{1}{1+t}\,\exp\frac{tx}{1+t}.$$

Let us compute the scalar product $S = \langle \bar{Q}(x,t) | \bar{Q}(x,v)\rangle_x$ in two different ways :

$$S = \sum_{k,p\geq 0} \langle \bar{Q}_{k-1}(x) | \bar{Q}_{p-1}(x)\rangle \frac{t^k}{k!}\cdot\frac{v^p}{p!}$$
$$= \sum_{k\geq 0} \langle \bar{Q}_{k-1}(x) | \bar{Q}_{k-1}(x)\rangle \frac{(tv)^k}{k!k!} = \frac{1}{1-tv}$$

by the orthogonality relations (proposition 4) $\langle \bar{Q}_{k-1} | \bar{Q}_{p-1}\rangle = \delta_{k,p}\cdot(k!)^2$; on the other hand,

$$S = \sum_{k,n\geq 0} \langle \bar{Q}_{k-1}(x) | x^n\rangle \frac{t^k}{k!}\cdot\frac{v^n}{(1+v)^{n+1}}\cdot\frac{1}{n!}$$
$$= \sum_{k,n\geq 0} H_{0,k,n}\,t^k\,\frac{v^n}{(1+v)^{n+1}}\cdot\frac{1}{n!}\ ,\ \text{by proposition 5.}$$

Identifying these expressions, and making the change of variable $\frac{v}{1+v} = z$ yields :

Proposition 6 : *The generating function* $\hat{H}(z,t) = \sum_{n,k\geq 0} H_{0,k,n}\,\frac{z^n}{n!}\,t^k$ *of histories has the expression* $\hat{H}(z,t) = \frac{1}{1-z-zt}$ *and its coeficients are given by* $H_{0,k,n} = n!\binom{n}{k}$.

From this expression of $H_{0,k,n}$, we can compute level crossing numbers $N_{k,n}$ by the formula (c) $N_{k,n} = \sum_i H_{0,k,i}\cdot H_{k,0,n-i}$, thus obtaining : $N_{k,n} = n!\sum_i \binom{i}{k}\binom{n-i}{k}/\binom{n}{i}$, a formula of Françon [78]. In order to derive an explicit expression of the generating function $N(z,t) = \sum_{k,n\geq 0} N_{k,n}\,t^k\,\frac{z^{n+1}}{(n+1)!}$, we remind the reader of a result, expressing the convolution theorem of Laplace transforms on formal power series :

Lemma C : *Let* $A(x) = \sum_{n\geq 0} a_n\,\frac{x^n}{n!}$ and $B(x) = \sum_{n\geq 0} b_n\,\frac{x^n}{n!}$ *be two exponential series ; the convolution* $C(x) = (A*B)(x) = \sum_{n\geq 0}\left(\sum_i a_i.b_{n-i}\right)\frac{x^{n+1}}{(n+1)!}$ *has the expression* $C(x) = \int_0^x A(x-\tau)B(\tau)d\tau$.

Proof : We merely check term to term equality :

$$C = \int_0^x A(x-\tau)B(x)dx = \sum_{n,m\geq 0}\int_0^t \frac{a_n b_m}{n!m!}(x-\tau)^n\,\tau^m\,d_\tau$$
$$= \sum_{n,m,k\geq 0}\int_0^t (-1)^k\,\frac{a_n b_m}{n!m!}\binom{n}{k}\,x^{n-k}\tau^{m+k}\,d_\tau$$
$$= \sum_{n,m\geq 0}\frac{a_n b_m}{n!m!}\,t^{n+m+1}\cdot\sum_k\frac{(-1)^k}{m+k+1}\binom{n}{k}\ ;$$

using the well known inversion formula for binomial coefficients

$$\frac{1}{(m+n)\binom{n+m}{n}} = \sum_k \frac{(-1)^k}{m+k+1} \binom{n}{k},$$

we get

$$C = \sum_{n,m\geq 0} a_n b_m \frac{t^{n+m+1}}{(n+m+1)!} = (A*B)(x) \qquad \square$$

Using formula (C) and this lemma, we can express the exponential generating function $\hat{N}(z,t) = \sum_{k,n\geq 0} N_{k,n}\, t^k \cdot \frac{z^{n+1}}{(n+1)!}$ to $\hat{H}$ by : $\hat{N}(z,t) = \sum_{k\geq 0} t^k\, \hat{H}_{0,k}(z) * \hat{H}_{k,0}(z)$ where, by proposition 6, $\hat{H}_{0,k}(z) = \hat{H}_{k,0}(z) = \frac{z^k}{(1-z)^{k+1}}$; substituting in the above expression yields $\hat{N}(z,t) = \int_0^z \frac{d\tau}{(1-x+\tau)(1-\tau)-t\tau(x-\tau)}$.

Explicit integration is obtained by decomposing the rational function of τ in the integrand into simple elements :

Proposition 7 : *The exponential generating function for level crossings* $N(z,t) = \sum_{k,n\geq 0} N_{k,n}\, t^k \frac{z^{n+1}}{n+1!}$ *has the expression* $N(z,t) = \frac{1}{\lambda(1-t)} \ln \frac{2\lambda+z}{2\lambda-z}$ *where* $\lambda = \left(\frac{z^2}{4} + \frac{1-z}{1-t}\right)^{\frac{1}{2}}$.

IV.2) The integral cost theorem

Let $c_0, c_1, \ldots, c_k, \ldots$ be an arbitrary sequence of non negative real numbers, where c_k represents the (average) unitary cost of operations on a stationary data structure of size $k\geq 0$. We let K_n represent the integrated cost of that structure, over the set H_n of dictionary histories of length n, starting and finishing with the empty set.

Theorem I : (The integral cost theorem).

For dictionaries, the generating function of unitary costs $C(u) = \sum_{k\geq 0} c_k\, u^k$ *and the exponential generating function of integrated cost* $K(z) = \sum_{n\geq 0} K_n \frac{z^{n+1}}{(n+1)!}$ *are related by the integral formula* :

$$K(z) = \frac{2}{2-z} \int_0^t \frac{C(u)du}{\sqrt{(1-u)(t-u)}} \quad , \quad T = \left(\frac{z}{2-z}\right)^2 .$$

Proof : Starting from $K(z) = \sum_{k,n\geq 0} c_k . N_{k,n} \frac{z^{n+1}}{(n+1)!}$ and using Lemma C yields $K(z) = \sum_{k\geq 0} c_k\, \hat{H}_{0,k}(z) * \hat{H}_{k,0}(z)$. By proposition 6, we get $K(z) = \sum_{k\geq 0} \int_0^z c_k \left(\frac{\tau(z-\tau)}{(1-\tau)(1-z+\tau)}\right)^k \frac{d\tau}{(1-\tau)(1-z-n)}$ thus

$K(z) = \int_0^z C\left(\frac{\tau(z-\tau)}{(1-\tau)(1-z+\tau)}\right) \frac{d\tau}{(1-\tau)(1-z+\tau)}$. The theorem then follows from two quadratic changes of variable. $\square$

This theorem solves the theoretical problem of computing integrated costs. With some additional effort, we compute integrated costs for the special cases relevant to our analyses, namely :

- polynomial costs : for $m\geq 0$, define $C_k^{(m)} = \binom{k}{m} = \frac{k(k-1)\ldots(k-m+1)}{m!}$;
- inverse costs : define $C_k^{(inv)} = \frac{1}{k+1}$;
- harmonic costs : let $C_k^{(H)} = H_k = 1 + \frac{1}{2} + \ldots + \frac{1}{k}$.

The corresponding generating functions are

$$C^{(m)}(u) = \sum_{k\geq 0} c_k^{(m)}\, u^k = \frac{u^m}{(1-u)^{m+1}} \text{ , and}$$

$$C^{(pol)}(u,v) = \sum_{k,m\geq 0} c_k^{(m)} u^k v^m = \frac{1}{1-u-uv} \quad ;$$

$$C^{(inv)}(u) = \sum_{k\geq 0} \frac{1}{k+1} u^k = \frac{1}{u} \ln \frac{1}{1-u} \quad ;$$

$$C^{(H)}(u) = \sum_{k\geq 0} H_k u^k = \frac{1}{1-u} \ln \frac{1}{1-u} \quad .$$

Plugging these expressions into Theorem I, and using (fairly) elementary calculus leads to :

Proposition 8 : *Let* $K^{(m)}$, $K^{(pol)}$, $K^{(inv)}$ *and* $K^{(H)}$ *be the exponential generating functions (in the sense of Theorem I) of integrated costs, corresponding to unitary costs* $C^{(m)}$, $C^{(pol)}$, $C^{(inv)}$ *and* $C^{(H)}$. *We have :*

- $K^{(m)}(z) = \frac{1}{(2m+1)\binom{2m}{m}} \cdot \frac{z^{m+1}}{(1-z)^{2m+1}}$;
- $K^{(pol)}(z,v) = \frac{2}{\sqrt{v\alpha}} \operatorname{arctg} \frac{vz^2}{\alpha}$ where $\alpha = 1 - z - \frac{v^2}{4}$, or, equivalently,

  $K^{(pol)}(z,v) = \frac{z}{(1-x)x\sqrt{1-x^2}} \arcsin x$ where $x = \frac{vz^2}{4(1-z)}$;
- $K^{(inv)}(z) = \frac{1}{z}.(\ln \frac{1}{1-z})^2$;
- $K^{(H)}(z) = \frac{2-z}{1-z} \ln \frac{1}{1-z} - \frac{2z}{1-z}$.

These expressions lead to remarkably simple forms for the integrated costs of sequences of n operations. Let us introduce the notation $\overline{K}_n = \frac{K_n}{n!}$; since the number of histories H_n is precisely n!, $\overline{K}_n$ is the average cost of a sequence of n operations.

Proposition 9 : *The integrated costs are given by the expressions :*

$$\overline{K}_n^{(m)} = \frac{m!}{(2m+1)!} (n+1)(n-m)(n-m-1)\ldots(n-2m+1)$$

$$\overline{K}_n^{(inv)} = 2\frac{n+1}{n+2} H_{n+1}$$

$$\overline{K}_n^{(H)} = (n+1) H_{n+1} - 2n-1 .$$

V - INTEGRATED COSTS FOR ULL, SLL AND BST.

In order to explicitly compute integrated costs for our three structures, we use the relations : $NQ^-_{k,n} = (k+1)N_{k,n-1}$, $NQ^+_{k,n} = k.N_{k,n-1}$, $NS_{k,n} = \frac{n+1-2k}{2}.N_{k-1,n-1} = NA_{k+1,n}$ and define $C_k = \frac{n-1-2k}{2}(CA_k + CS_{k+1}) + k\,CQ^+_k + (k+1)\,CQ^-_k$, the level crossing costs, where the CO_k are given by Tables 1 and 2.

	Number of Comparisons	MIX time
ULL	$\frac{k(k+1)}{2} + \frac{n-1}{2}(k+1)$	$18.5n - 14.5 + (4.5n - 27.5)k$
SLL	$1 + \frac{n-1}{2}(k+2)$	$\frac{n-1}{2}(7.5k+40) - 1.5k^2 - 24k + 8$
BST	$2nH_{k+1} - \frac{5n+3}{2} + \frac{n+1}{k+1} \cdot H_{k+1}$	$\frac{n+1}{2}(34H_{k+1} - 8.5 + \frac{13}{k+1} H_{k+1} + \frac{3}{k+1})$ $- 4k H_{k+1} + 12H_{k+1} - 33.5k - 24.5$

Table 3. Level crossing costs

Computing integrated costs then follows from Proposition 9 ; we have the additional problem of computing integrated costs relative to unitary costs of the form $c_k = kH_k$; in this case, we can show that $\overline{K}_n = \frac{n-1}{6} H_{n+1} - \frac{5n^2-n-3}{3n(n+1)}$. Putting all this together yields Table 4, below.

	ULL	SLL	BST
number of comparison	$\frac{1}{10} n^2 + \frac{3n}{10} - \frac{7}{15}$	$\frac{n}{12} + \frac{3n}{4} + \frac{1}{6}$	$2nH_n - \frac{5}{2} n + O(\log^2 n)$
MIX time	$\frac{3}{4}n^2 + \frac{167}{12}n + \frac{13}{4} + O(\frac{1}{n})$	$\frac{23}{40}n^2 + \frac{613}{40}n + \frac{293}{40} + O(\frac{1}{n})$	$\frac{49}{3} n H_{n+1} - \frac{739}{18} n + O(\log^2 n)$

Table 4. Integrated costs $\overline{K}_n$

This table can be made complete so as to provide explicit expressions for the terms $O(\log^2 n)$, but computations get to be tedious (it might be time to let a computer do the work !).

Looking at Table 4 tells us that sorted lists are only marginally better than unsorted lists ; binary search trees become more efficient as soon as n is of the order of 20.

VI - DIRECTIONS FOR FUTURE WORK

1) Continued fractions provide us with a powerful tool for computing integrated costs work similar to that done here for dictionaries should be carried out for other important data-organisations : queues, stacks, priority queues, linear lists,... .

2) Integrated costs should be computed for other relevant sets of histories : histories of length n, of bounded height, histories located in a plane rectangle ; a priori weighting of each operations can also be achieved (our history choice yields a posteriori weighting of $\frac{1}{3} - \frac{1}{6n}$ for A and S, $\frac{1}{6} - \frac{1}{3n}$ for Q^+ and $\frac{1}{6} + \frac{2}{3n}$ for Q^-).

3) A tantalizing open question is computing the integrated costs for structures (say balanced trees) which are not stationary ; asymptotic analysis might be carried out, but it certainly appears to be a challenging problem.

4) There are many interesting "sequence of operation analysis" problems for which the continued fraction approach does not immediatly apply : keys drawn from a bounded set, under hypothesis H1 alone or under various a priori probabilistic assumptions. Natural types of histories should also be found for hashing schemes, digital search trees, and, in a more general context for operations (union, attribute selection,...) other than A,S and Q.

Much methodological work remains to be done here, but we are hopeful that continued fractions will prove to be relevant in many of the questions raised.

ACKNOWLEDGEMENTS :

Many thanks are due to J. Giraud and G. Viennot for interesting suggestions and discussions relative to this work. J. Giraud much helped in our understanding of orthogonality relations. We are grateful to Louis Monier for his help in filling in Table 4.

BIBLIOGRAPHIE

[A.H.U. 74] Alfred V. Aho, John E. Hopcroft, and Jeffrey D. Ullman, The Design and Analysis of Computer Algorithms (Reading, Mass. : Addison-Wesley, 1974).

[D.&R. 76] Jon Doyle and Ronald L. Rivest, "Linear expected time of a simple Union-Find algorithm", Information Processing Letters 5 (1976), 146-148.

[Fla. 78] Philippe Flajolet, "Analyse d'Algorithmes de Manipulation de Fichiers", IRIA-Laboria Report N° 321 (August 1978), 28pp.

[F.&F.&V.&V. 79] Philippe Flajolet, Jean Françon, Gérard Viennot, Jean Vuillemin, Algorithmique et Combinatoire des Arbres et des Permutations , to appear (1979).

[Fra. 78] Jean Françon, "Histoires de Fichiers", R.A.I.R.O., Inf. Th. vol 12, pp 49-67, (1978).

[F.&V. 78] Jean Françon and Gérard Viennot, Permutations selon leur Pics, Creux, Doubles Montées et Doubles Descentes, Nombres d'Euler et de Genocchi . To appear in Discrete Math.

[Hib. 62] Thomas N. Hibbard, "Some combinatorial properties of certain trees with applications to searching and sorting", J.ACM 9 (1962), 13-28.

[J.D. 75] Arne Jonassen and Ole-Johan Dahl, "Analysis of an algorithm for priority queue administration", Math. Inst., University of Olso 1975.

[J.,K. 78] Arne Jonassen and Donald E. Knuth, "A trivial algorithm whose analysis isn't", J. of Computer and System Sc. 16 (1978) 301-322.

[Kno. 75] Gary D. Knott, "Deletion in binary storage trees", Ph.D. Thesis, Computer Science Department, Stanford University (May 1975), 93pp.

[Knu. 68] Donald E. Knuth, The Art of Computer Programming, vol. 1, Fundamental Algorithms (Reading, Mass. : Addison-Wesley, 1968).

[Knu. 73] Donald E. Knuth, The Art of Computer Programming, vol. 3, Sorting and Searching, (Reading, Mass. : Addison-Wesley, 1973).

[Knu. 77] Donald E. Knuth, Deletions that preserve randomness IEEE Trans. Soft. Engrg. SE 3 (1977) 351-359.

[K.&S. 77] Donald E. Knuth and Arnold Schönhage, "The expected linearity of a simple equivalence algorithm", Stanford University Report STAN-CS-77-599, (March 1977), 56pp.

[Sny. 77] Lawrence Snyder, "On uniquely represented data structures" Proceeding of the 18th Annual Symposium on Foundations of Computer Science, pp. 142-146.

[Tar. 75] Robert E. Tarjan, Efficiency of a good but not linear set union algorithm J.ACM, 22 (1975), 215-225.

[Wall. 48] H.S. Wall, Analytic Theory of Continued Fractions, Chelsea Publishing Company, N.Y. (1948)

[Yao. 76] Andrew Chi-Chih Yao, On the average behavior of set merging algorithms , (extended abstract), Proc. ACM Symp. Theory of Computation 8 (1976), 192-195.

[Stie. 94] T.J. Stieltjes,"Recherches sur les fractions continues", Annales Faculté des Sciences de Toulouse, vol. 8 (1894) pp. 1-122.