

Distributional Analysis of Recursive Algorithms by the Contraction Method

Ralph Neininger

University of Freiburg

November 22, 1999

Summary by Elchanan Mossel

1. Basic Algorithms

We consider *records* which belong to a k -dimensional region $D = D_1 \times \dots \times D_k \subset \mathbb{R}^k$. A *file* is a finite subset F of D . Given a *query* $q \in (D_1 \cup \{*\}) \times \dots \times (D_k \cup \{*\})$, the objective is to find all the records $r \in F$ such that $r_i = q_i$ when $q_i \neq *$. The probabilistic assumption is that all the coordinates of the records and the queries (which are not $*$) are independent uniform random variables. For the discussion below, it is easy to see that this assumption may be replaced by a weaker assumption that all variables are independent with the same continuous distribution. The *specification pattern* consists of the configuration in $\{*, S\}^k$ of specified and unspecified variables.

There exist several comparison-based trees:

- *Quadtrees*. Each record x has 2^k subtrees which correspond to all possible elements of $\{<, >\}^k$. Thus (y_0, y_1, y_2) will belong to the $(<, >, <)$ subtree of (x_0, x_1, x_2) if $y_0 < x_0, y_1 > x_1, y_2 < x_2$. See Figure 1.
- *kD trees*. Each record x at level l has two subtrees corresponding to $x_{l \bmod k} > y_{l \bmod k}$ and $x_{l \bmod k} < y_{l \bmod k}$, respectively.
- *Randomised kD trees*. Each record x at level l has two children corresponding to $x_{l(x)} > y_{l(x)}$ and $x_{l(x)} < y_{l(x)}$ where $l(x)$ are i.i.d. uniform variables in the range $0, \dots, k - 1$.

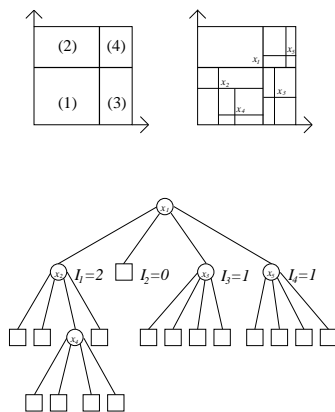


FIGURE 1. A quadtree: the data partition the unit-cube recursively into quadrants; the quadtree corresponds to this partitioning.

– *Squarish kD trees.* This is another version in which every node has two subtrees, but the coordinate with respect to which we split depends more strongly on the tree structure.

The basic quantity we are after is the limit law of C_n where C_n is the random number of nodes we traverse when finding all records which match the query. Here n is the size of F .

2. Quadrees in Two Dimensions

We let $W = (U, V)$ be the first key to be inserted and $q = (Y, *)$ be the query. So U, V , and Y are uniform i.i.d. variables. We also let I^n be the vector of cardinalities for the subtrees of the root. We thus derive the following recursive distributional equation:

$$C_n = 1_{Y < U} \left(C_{I_1^n}^1 + C_{I_2^n}^2 \right) + 1_{Y > U} \left(C_{I_3^n}^3 + C_{I_4^n}^4 \right) + 1,$$

wherein the variables Y, U, V , and C_i^j are independent and all the C_i^j have the distribution of C_i . Given (U, V) the variable I^n is multi-monomial with parameters (U, V) and n .

By previous works [1, 2, 6] there are known constants α, β , and γ for which

$$\mathbf{E}[C_n] \sim \gamma n^{\alpha-1}, \quad \mathbf{Var}[C_n] \sim \beta n^{2\alpha-2}.$$

Looking for a limit, we consider the variable: $X_n = (C_n - \mathbf{E}[C_n])/n^{\alpha-1}$.

In this way we obtain the equation:

$$(1) \quad X_n = 1_{Y < U} \left(\left(\frac{I_1^n}{n} \right)^{\alpha-1} \left(X_{I_1^n}^1 + \gamma \right) \right) + 1_{Y < U} \left(\left(\frac{I_2^n}{n} \right)^{\alpha-1} \left(X_{I_2^n}^2 + \gamma \right) \right) \\ + 1_{Y > U} \left(\left(\frac{I_3^n}{n} \right)^{\alpha-1} \left(X_{I_3^n}^3 + \gamma \right) \right) + 1_{Y > U} \left(\left(\frac{I_4^n}{n} \right)^{\alpha-1} \left(X_{I_4^n}^4 + \gamma \right) \right) - \gamma + o(1).$$

By the law of large numbers we have

$$I^n/n \rightarrow W = (UV, U(1-V), (1-U)V, (1-U)(1-V))$$

in probability. We thus obtain the following limiting equation:

$$(2) \quad X = 1_{Y < U} W_1^{\alpha-1} (X^1 + \gamma) + 1_{Y < U} W_2^{\alpha-1} (X^2 + \gamma) \\ + 1_{Y > U} W_3^{\alpha-1} (X^3 + \gamma) + 1_{Y > U} W_4^{\alpha-1} (X^4 + \gamma) - \gamma,$$

where the X^i are independent copies of X .

This suggests that we should consider the following operator on random variables Z :

$$T(Z) = 1_{Y < U} W_1^{\alpha-1} (Z^1 + \gamma) + 1_{Y < U} W_2^{\alpha-1} (Z^2 + \gamma) \\ + 1_{Y > U} W_3^{\alpha-1} (Z^3 + \gamma) + 1_{Y > U} W_4^{\alpha-1} (Z^4 + \gamma) - \gamma,$$

where the Z^i 's are independent copies of Z .

We now work with the following metric (on the space of variables with zero mean and finite variance): $l_2(Z, Z') = \inf(\mathbf{E}[Z - Z']^2)^{1/2}$ where the infimum is taken over all couplings of Z and Z' . It turns out that this space equipped with this metric is a Banach space. Moreover, using the representation of T one can see that T is a contraction on this space. It therefore follows that there exists a unique random variable Z which satisfies $T(Z) = Z$.

The main technical part of the proof is showing that we obtain the same limit if we work with the exact equations (1) instead of the approximate equations (2). This essentially uses the known estimates that $\mathbf{E}[C_n] = \gamma n^{\alpha-1}(1 + o(1))$. In this way we obtain the following theorem.

Theorem 1. Let X_n be the normalised number of traversed nodes and X the variable such that $T(X) = X$, then $l_2(X_n, X) \rightarrow 0$.

3. Other Trees

3.1. Multidimensional quadtree. In a similar manner one can prove the same kind of result for multidimensional quadtree. One of the differences is that in this case the variance $\mathbf{Var}[C_n]$ is not known beforehand. Instead, we guess that the right normalisation should be

$$X_n = \frac{C_n - \mathbf{E}[C_n]}{n^{\alpha-1}}.$$

In this way we obtain again a limit law similar to the above: the limit X depends only on the number of $*$'s in the query. Given this limit law we can now compute a constant which depends only on the number of $*$'s in the query such that $\mathbf{Var}[C_n] = \beta n^{2\alpha-2}$.

3.2. k D Trees. Vaguely speaking, the difference between quadtrees and k D trees, is that for k D trees different levels behave differently. Thus, in order to obtain a theorem similar to the above, a single recursion step should go k levels forward instead of just one. Doing that, we obtain a result similar to the above.

3.3. Randomised k D tree. The randomisation allows one to use one-level recursion, therefore obtaining a theorem and a proof similar to the case of quadtrees.

3.4. Squarish k D tree. It seems like the above methods do not work in this case. This is because the coordinate with respect to which we split depends on the structure of the tree and on the data stored in it.

4. Internal Path Length in Random Trees

In the previous sections we studied the cost of a query. In this section we consider the cost of building the tree which is nothing but the sum of depths of nodes in the tree. For the quadtrees we obtain the following recursive equation:

$$Y_n = \sum_{k=0}^{2^d-1} Y_{I_k^n}^k + n.$$

The article [3] gives the expectation $\mathbf{E}[Y_n] = (2/d)n \ln n + u_d n + o(n)$, but the variance was not derived there. We guess the normalisation: $X_n = (Y_n - \mathbf{E}[Y_n])/n$. We therefore obtain the equation:

$$X_n = \sum_{i=0}^{2^d-1} \frac{I_k^n}{n} X_{I_k^n}^k + C_n(I^n)$$

where

$$C_n(i_0, \dots, i_{2^d-1}) = 1 + \frac{1}{n} \sum_{i=0}^{2^d-1} \mathbf{E}[Y_{i_k}] - \mathbf{E}[Y_n].$$

Using the expectation formula we obtain:

$$C_n(i) = 1 + \frac{2}{d} \sum_{i=0}^{2^d-1} \frac{i_k}{n} \ln \frac{i_k}{n} + o(1).$$

We now continue in the same route as before to obtain the l_2 limit and the asymptotic variance.

5. Find Algorithm

We consider the following version of quicksort. We want to sort the values $\{1, \dots, n\}$ which are given in a random uniform permutation. In order to perform the sort we pick a pivotal element p and continue sorting the elements larger than this element, and the elements smaller than this element. The way to pick p is by taking three independent uniform keys k_1, k_2, k_3 and taking p to be their median. We thus obtain the following recursion equation:

$$C_n = 1_{Z_n > M_n} C'_{Z_n-1} + 1_{Z_n < M_n} C''_{n-Z_n} + n - 1$$

where M_n is uniform in $\{1, \dots, n\}$ and Z_n is a median of three uniform variables in $\{1, \dots, n\}$. We now continue in a similar way: it is known [4, 5] that $\mathbf{E}[C_n] = 5n/2 + O(\ln n)$, we guess that the normalisation is: $Y_n = (C_n - \mathbf{E}[C_n])/n$ to obtain a limit law. This limit law enables us to give asymptotic form for all the moments: $\mathbf{E}[C_n^k] \sim m_k n^k$ where we have a closed formula for m_k .

Bibliography

- [1] Flajolet (Philippe), Gonnet (Gaston), Puech (Claude), and Robson (J. M.). – The analysis of multidimensional searching in quad-trees. In *Proceedings of the Second Annual ACM-SIAM Symposium on Discrete Algorithms (San Francisco, CA, 1991)*. pp. 100–109. – ACM, New York, 1991.
- [2] Flajolet (Philippe), Gonnet (Gaston), Puech (Claude), and Robson (J. M.). – Analytic variations on quadtrees. *Algorithmica*, vol. 10, n° 6, 1993, pp. 473–500.
- [3] Flajolet (Philippe), Labelle (Gilbert), Laforest (Louise), and Salvy (Bruno). – Hypergeometrics and the cost structure of quadtrees. *Random Structures & Algorithms*, vol. 7, n° 2, 1995, pp. 117–144.
- [4] Kirschenhofer (P.), Prodinger (H.), and Martínez (C.). – Analysis of Hoare's FIND algorithm with median-of-three partition. *Random Structures & Algorithms*, vol. 10, n° 1-2, 1997, pp. 143–156. – Average-case analysis of algorithms (Dagstuhl, 1995).
- [5] Kirschenhofer (Peter), Martínez (Conrado), and Prodinger (Helmut). – Analysis of an optimized search algorithm for skip lists. *Theoretical Computer Science*, vol. 144, n° 1-2, 1995, pp. 199–220. – Special volume on mathematical analysis of algorithms.
- [6] Martínez (Conrado), Panholzer (Alois), and Prodinger (Helmut). – On the number of descendants and ascendants in random search trees. *Electronic Journal of Combinatorics*, vol. 5, n° 1, 1998, pp. Research Paper 20, 36 pp.
- [7] Neininger (Ralph). – Asymptotic distributions for partial match queries in kD trees. – 1999. Preprint.
- [8] Neininger (Ralph) and Rüschemdorf (Ludger). – On the internal path length of d -dimensional quad trees. *Random Structures & Algorithms*, vol. 15, n° 1, 1999, pp. 25–41.