Asymptotic size of search trees for Fibonacci matchings

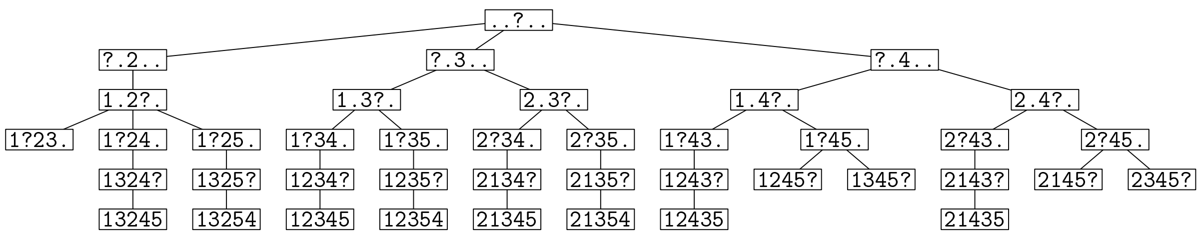
(Don Knuth, Stanford Computer Science Department)
(04 February 2020; revised 05 February 2020)

I like to collect toy problems that serve as "training wheels," because they help me to acquire mathematical tools for later use on real-world challenges. This note is a case in point, based on three problems suggested by a recent preprint of Persi Diaconis [1].

A *Fibonacci matching* is a way to match $\{1', \ldots, n'\}$ to $\{1, \ldots, n\}$ in such a way that each k' has been matched only with $k{-}1$ or k or $k{+}1$. It's well known, and easy to discover, that there are exactly F_{n+1} such matchings — a Fibonacci number, hence the name. Furthermore, those matchings are in bijection with "Morse code sequences" of length n; there's an inversion for every dash. For example, the matching that takes $1'2'3'4'5'6'7'8'9'$ to 132546798 corresponds to the Morse code sequence dot-dash-dash-dot-dot-dash of length 9.

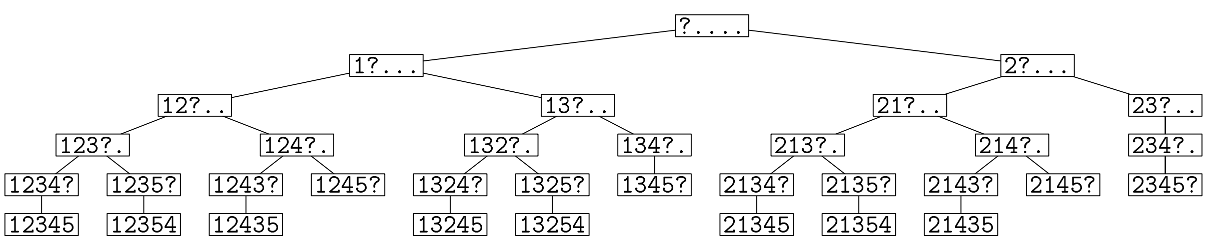
One way to find such matchings is to search exhaustively through all possibilities, assigning a mate first to p'_1, next to p'_2, …, and finally to p'_n, where $p_1p_2\ldots p_n$ is a permutation of $\{1, 2, \ldots, n\}$. During this process we're allowed to assign p'_k to either $p_k - 1$ or p_k or $p_k + 1$, unless one of those numbers is 0, or $n+1$, or already assigned to p'_1, …, or already assigned to p'_{k-1}. This leads to a tree of partial matchings, with at most three branches at every node.

For example, here's how that search tree looks when $n = 5$ and $p_1p_2p_3p_4p_5 = 31425$:



Level k of this tree, for $k \geq 0$, constains the nodes that represent partial matchings of size k — one for each way to assign mates to p'_1 through p'_{k-1}. Those mates are named explicitly, with dots in the unassigned positions. However, if $k < n$ there's a question mark in position p_{k+1}, instead of a dot. (Notice that there may not be any available mate remaining for p_{k+1}, as in the case '`1?23.`'; such a node has no descendants.)

Toy problem 1. What is the search tree size when $p_1p_2\ldots p_n = 12\ldots n$? The tree for $n = 5$ makes it clear that left-to-right assignment is easy to analyze quantitatively:

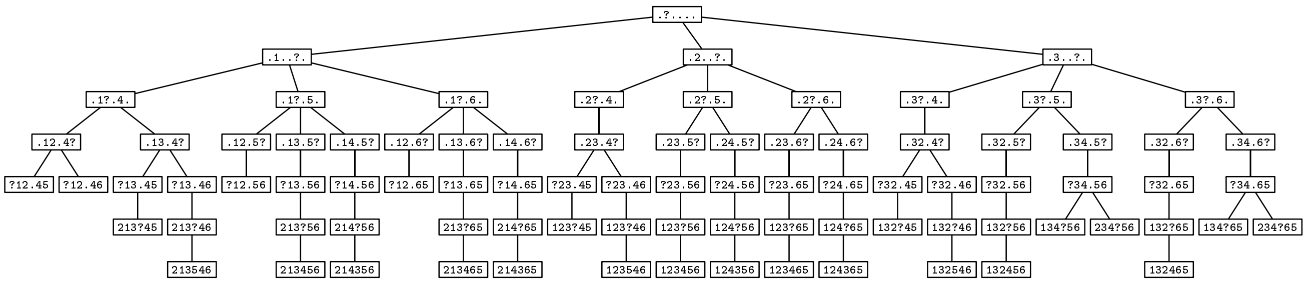


Level k now contains F_{k+1} nodes with all assignments $\leq k$. There also are $F_{k+2} - 1$ additional nodes, if $k < n$, of which $F_{k+1} - 1$ end with '$k(k{+}1)$?'. (Classify the nodes as type A, 'fresh'; type B, 'critical'; type C, 'doomed'. The root is type A. The children of type A are types A and B. The children of type B are types A and C, except only A at the bottom. Nodes of type C have a single child, of type C, except at the bottom.)

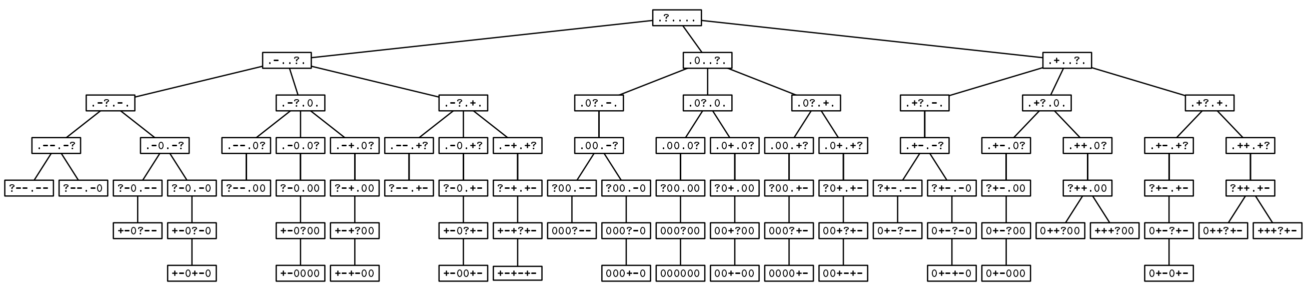
Thus there are $F_{n+4} - n - 3$ partial matchings altogether, on levels less than n. The bottom level adds F_{n+1} further nodes, which represent the *total* matchings.

Toy problem 2. What is the search tree size when $n = 3m$ and $p_1p_2\ldots p_n$ is the "skip-by-three" permutation $25\ldots(3m-1)36\ldots(3m)14\ldots(3m-2)$? (This permutation starts with m three-way branches; the next m branches will be ternary, binary, or unary.)

Here, for example, is the case $m = 2$, $n = 6$:



The pattern is clearer, however, if we replace $(k-1, k, k+1)$ in position k respectively by $(\texttt{-}, \texttt{0}, \texttt{+})$:



(Zoom if you can't read it.) Level k clearly contains 3^k nodes, for $0 \le k \le m$.

Levels $m + 1$ through $2m$ are more interesting. For these we define auxiliary sequences

$$A_0 = B_0 = 1; \qquad A_{n+1} = 2A_n + 2B_n; \qquad B_{n+1} = 3A_n + 4B_n.$$

Since the generating functions $A(z) = \sum_n A_n z^n$ and $B(z) = \sum_n B_n z^n$ satisfy $A(z) = 1 + z(2A(z) + 2B(z))$ and $B(z) = 1 + z(3A(z) + 4B(z))$, we find $A(z) = (1 - 2z)/(1 - 6z + 2z^2)$ and $B(z) = (1 + z)/(1 - 6z + 2z^2)$. Hence the growth rate is the largest zero of $x^2 - 6x + 2 = 0$, namely $3 + \sqrt{7} \approx 5.6457$. Let $U(z) = \sum_n U_n z^n = 1/(1 - 6z + 2z^2)$. The first few values are

$n =$	0	1	2	3	4	5	6	7	8	9	10	11	12
$A_n =$	1	4	22	124	700	3952	22312	125968	711184	4015168	22668640	127981504	722551744
$B_n =$	1	7	40	226	1276	7204	40672	229624	1296400	7319152	41322112	233294368	1317121984
$U_n =$	1	6	34	192	1084	6120	34552	195072	1101328	6217824	35104288	198190080	1118931904

[Incidentally, the OEIS currently has $\langle U_n \rangle$ in A154244, $\langle B_n \rangle$ in A180034. Its definition of A180034 is somewhat strange. I guess I'll submit $\langle A_n \rangle$, and contribute some brief amendments to the others.]

If $m > 1$, level $m + 1$ contains A_1 nodes below nodes that have '$\texttt{-}$' in position 5, but B_1 nodes below the others; total $(A_1 + 2B_1)3^{m-2} = U_1 3^{m-1}$.

If $m > 2$, level $m + 2$ contains A_2 nodes below nodes that have '$\texttt{-}$' in position 8, but B_2 nodes below the others; total $(A_2 + 2B_2)3^{m-3} = U_2 3^{m-2}$.

And so on: If $m > k$, level $m + k$ contains A_k nodes below nodes that have '$\texttt{-}$' in position $3k + 2$, but B_k nodes below the others; total $(A_k + 2B_k)3^{m-k-1} = U_k 3^{m-k}$.

That takes us up through level $2m - 1$. Level $2m$ is almost like its predecessor; but the nodes ending with '$\texttt{-}$' in position $n - 1$ have two children, while other nodes have just one. There are A_{m-1} nodes of the former kind. Hence the total number of nodes on level $2m$ is $3U_{m-1} + A_{m-1} = A_m = U_m - 2U_{m-1}$.

Finally, what about levels $2m + 1$ through $3m$? That's where "bad" choices finally cause lines to die out, until eventually only F_{n+1} perfect matchings remain. On the other hand, some nodes still do branch (temporarily) into two lines in this third phase.

One can show by induction that the A_m nodes on level $2m$ include exactly U_{m-1} that begin with '0', as well as exactly U_{m-1} that begin with '+'.

Using that fact, we can show that, for $0 < k \le m$, the number of nodes on level $2m+k$ that put '-' in position $3k-2$ is $2F_{3k-3}U_{m-k}$. The number that put '0' there is $2F_{3k-2}U_{m-k}$. And the number that put '+' there is $F_{3k-2}A_{m-k}$ if followed by '-0', $2F_{3k-2}U_{m-k-1}$ if followed by '-+', and $2(F_{3k}-1)U_{m-k-1}$ if followed by '++'. Summing these gives a total of $F_{3k+1}U_{m-k} + 2(F_{3k}-1)U_{m-k-1}$. (Notice that this nicely gives F_{n+1} when $k = m$, because $U_{-1} = 0$.)

As the level increases from m to $2m$, the number of nodes per level rises exponentially, by a factor of roughly $(3+\sqrt{7})/3$. Then, between levels $2m$ and $3m$, it falls exponentially, by a factor of roughly $\phi^3/(3+\sqrt{7})$. Hence the total number of nodes is asymptotically proportional to the size of level $2m$, namely $\Theta((3+\sqrt{7})^m) = \Theta(\alpha^n)$, where $\alpha = (3+\sqrt{7})^{1/3} \approx 1.78063$.

Toy problem 3. What is the average search tree size, averaged over all $n!$ permutations $p_1p_2\ldots p_n$?

This problem is significantly more difficult, so I eventually asked for help. First, however, I found it reasonably easy to count the number $a_{n,k}$ of partial matchings of size k: Exactly $a_{n-1,k}$ of them leave n' and n unmatched. Exactly $a_{n-1,k-1} + a_{n-2,k-2}$ of them leave n' and n matched (either to each other or to their other neighbors). Then several cases arise if n' is matched but not n: Exactly $a_{n-2,k-1}$ of them end with '.-'. Exactly $a_{n-3,k-2}$ of them end with '.--'. And so on. (Note that $a_{n-k,0} = 1$; it counts the partial k-matching that ends with k minus signs.) Finally, by symmetry, there are just as many cases where n is matched but not n'. Hence the generating function $A(q,z) = \sum_{k,n} a_{n,k}\, q^k z^n$ satisfies

$$A(q,z) = 1 + (z + qz + q^2z^2 + 2qz^2/(1-qz))A(q,z).$$

We have therefore

$$A(q,z) = \frac{1-qz}{1-z-2qz-qz^2+q^3z^3} = 1 + (1{+}q)z + (1{+}4q{+}2q)z^2 + (1{+}7q{+}11q{+}3q)z^3 + \cdots.$$

After finding these formulas and computing the sequence of sums $\sum_k a_{n,k}$, namely (1, 2, 7, 22, 71, 228, 733, ...), I naturally decided to look it up in the OEIS. That sequence turns out to be A030186, which is in fact a gold mine of information — because the problem of partial Fibonacci matchings happens to be the same as the problem of placing k dominoes on a $2 \times n$ chessboard (matching black squares to white neighbors)! Thus I learned that this recurrence for $a_{n,k}$ was first found by McQuistan and Lichtman in 1970 [2], who gave a table of $a_{n,k}$ for $0 \le k \le n \le 10$ and showed that $a_{n,k}$ is maximized when $k \approx .606n$.

But these numbers $a_{n,k}$, interesting as they are, aren't the answer to the problem. Each of the $a_{n,k}$ partial matchings occurs exactly $k!\,(n-k)!$ times among the $n!$ search trees, because it occurs in the trees for precisely those permutations $p_1p_2\ldots p_n$ whose first k elements are matched.

Thus the average number of nodes on level k is $a_{n,k}/\binom{n}{k}$. And the answer to toy problem 3, call it a_n, is the sum of those numbers for $0 \le k \le n$. For example, we have $(a_0, a_1, a_2, a_3, a_4, a_5) = (1, 2, 5, 10, \frac{119}{6}, \frac{189}{5})$.

At this point I wrote to Ira Gessel, asking for suggestions about what to do. And he replied immediately [3] by reminding me about the Beta function, namely

$$\mathrm{B}(k+1, n-k+1) \;=\; \int_0^1 t^k(1-t)^{n-k}\,dt \;=\; \frac{\Gamma(k+1)\,\Gamma(n-k+1)}{\Gamma(n+2)} = \frac{k!\,(n-k)!}{(n+1)!}.$$

Aha! It follows that

$$\sum_{k,n} \frac{k!\,(n-k)!}{(n+1)!} a_{n,k}\, q^k z^n \;=\; \int_0^1 A\Big(\frac{tz}{(1-t)}, (1-t)z\Big)\,dt \;=\; \int_0^1 \frac{(1-tqz)\,dt}{1-(1-t)z-2tqz-(1-t)tqz^2+t^3q^3z^3}.$$

Setting $q = 1$ gives us a decent generating function for the answer:

$$\frac{a_0}{1} + \frac{a_1}{2}z + \frac{a_2}{3}z^2 + \cdots = G(z) = \int_0^1 \frac{1-t}{1-(1+t)z-(1-t)tz^2+t^3z^3}\,dt.$$

But now what? We want to know the asymptotic behavior of the coefficients of $G(z)$. Ira observed that the discriminant of the denominator polynomial $p(t,z)$ with respect to t is $4z^6(z^3+z^2+18z-11)$; the nonzero points where this vanishes are where $p(t,z)=(t-r_1(z))(t-r_2(z))(t-r_3(z))$ has a double root. Such points are probably singularities of $G(z)$, so they may well be key to an asymptotic analysis.

Indeed, the roots of $z^3+z^2+18z-11=0$ are the reciprocals of the roots of $11z^3=18z^2+z+1$. That equation has one real root $r\approx 1.7199502092911808681$; and it also has complex roots $\approx -0.04179\pm 0.22607i$ of negligible magnitude ≈ 0.23. The values of a_n for $n\le 100$ are consistent with an asymptotic growth rate of roughly r^n. So the answer we seek almost certainly comes from a singularity at $z=1/r$.

How can it be proved rigorously? This time I wrote to Philippe Jacquet for help. Sure enough, he soon came through with an explanation [4] of how to deal with generating functions of this type. By studying the behavior of $p(t,z)$ for $0\le t\le 1$ when z is near $1/r$, using a bivariate Taylor expansion, he proved that $G(z)$ has a quadratic singularity there. More precisely, there's an asymptotic expansion

$$G(z)=\frac{\gamma_{-1}}{\sqrt{1/r-z}}+\gamma_0+\gamma_1\sqrt{1/r-z}+\gamma_2(1/r-z)+\gamma_3(1/r-z)^{3/2}+\gamma_4(1/r-z)^2+\cdots,$$

where the coefficients γ_m are expressible as complicated functions of r.

Now we're essentially done, because $[z^n]\,(1/r-z)^\alpha=r^{n-\alpha}[z^n]\,(1-z)^\alpha$; and

$$[z^n]\,(1-z)^\alpha\sim\frac{n^{-\alpha-1}}{\Gamma(\alpha)}\Big(1+\frac{\alpha(\alpha+1)}{2n}+\frac{\alpha(\alpha+1)(\alpha+2)(3\alpha+1)}{24n^2}+\cdots\Big)$$

by Eq. (2.2) in [5].

Consequently $a_n=c\sqrt{n}\,r^n(1+O(1/n))$, where c is a constant.

I could find an exact expression for c if I had time; but I've got other commitments at the moment. Empirically, $c\approx 1.14$. (In the neighborhood of $n=1000$, $a_n\approx 1.1401\sqrt{n}\cdot r^n(1+.55/n+O(1/n^2))$. The value of a_{1000}, to about seventeen decimal places, is $1.1830684781516635\times 10^{237}$.)

Extremes. Does the permutation in problem 1 minimize the search tree size? Does the permutation in problem 2 maximize it? I leave those questions to the reader.

[1] `http://www.statweb.stanford.edu/~cgates/PERSI/papers/sequential-importance-sampling.pdf`: Persi Diaconis, "Sequential importance sampling for estimating the number of perfect matchings in bipartite graphs: An ongoing conversation with Laci," (2018).

[2] R. B. McQuistan and S. J. Lichtman, "Exact recursion relation for $2\times N$ arrays of dumbbells," *Journal of Mathematical Physics* **10** (1970), 3095–3099.

[3] Ira Gessel, personal communications (27 and 28 December 2019).

[4] Philippe Jacquet, "Knuth problem 2020," unpublished notes (received 24 January 2020).

[5] Philippe Flajolet and Andrew Odlyzko, "Singularity analysis of generating functions," *SIAM Journal on Discrete Mathematics* **3** (1990), 216–240.