A MAXMIN PROBLEM ON FINITE AUTOMATA

Jean-Marc Champarnaud and Jean-Eric Pin
L.I.T.P, Universités Paris 6 et 7, Tour 55-56,
4 Place Jussieu, 75252 Paris Cedex 05, France*

January 19, 1999

Abstract

We solve the following problem proposed by H. Straubing. Given a two letter alphabet A, what is the maximal number of states $f(n)$ of the minimal automaton of a subset of A^n, the set of all words of length n. We give an explicit formula to compute $f(n)$ and we show that $1 = \underline{\lim}_{n\to\infty} nf(n)/2^n \leq \overline{\lim}_{n\to\infty} nf(n)/2^n = 2$.

The purpose of this note is to solve the following question, raised by H. Straubing. Let $A = \{a, b\}$ be a two-letter alphabet. For each finite language L, denote by $s(L)$ the number of states of the minimal (deterministic) automaton of L, and put

$$f(n) = \max\{s(L) \mid L \subset A^n\}$$

The problem is to compute $f(n)$ and to give, if possible, an asymptotic equivalent.

We first recall some definitions (see [1] for more details). An automaton $\mathcal{A} = (Q, A, \cdot, q_0, F)$ consists of a (finite) set of states Q, a finite set of letters A, an initial state $q_0 \in Q$, a set of final states $F \subset Q$, and a partial function $Q \times A \to Q$ denoted by $(q, a) \to q\cdot a$. This function is extended to a (partial) function $Q \times A^* \to Q$, called the transition function, by the rules

(a) for every $q \in Q$, $q\cdot 1 = q$,

(b) for every $q \in Q$, $u \in A^*$ and $a \in A$, $q\cdot(ua) = (q\cdot u)\cdot a$ if $(q\cdot u)$ and $(q\cdot u)\cdot a$ are defined, and $q\cdot(ua)$ is undefined otherwise.

If the transition function is a total function, $\mathcal{A}$ is a *complete* automaton and it is *uncomplete* otherwise. The *language* accepted by $\mathcal{A}$ is the set

$$L(\mathcal{A}) = \{u \in A^* \mid q_0\cdot u \in F\}$$

A state q is *accessible* (resp. *coaccessible*) if $q_0\cdot u = q$ (respectively $q\cdot u \in F$) for some word $u \in A^*$. Two states q and q' are *equivalent* (in $\mathcal{A}$) if, for every word $u \in A^*$,

*This research was supported by the "Programme de Recherches Coordonnées - Mathématiques et Informatique".

$q\cdot u \in F$ is equivalent to $q'\cdot u \in F$. An automaton is *reduced* if, for any $q, q' \in Q$, q equivalent to q' implies $q = q'$.

Let us mention a trivial, but useful, observation. If $q, q' \notin F$ are not equivalent, then there exists a letter $a \in A$ such that either $q\cdot a \neq q'\cdot a$, or $q\cdot a$ is defined and $q'\cdot a$ is undefined, or $q\cdot a$ is undefined and $q'\cdot a$ is defined.

Finally, an automaton is *minimal* if it is reduced and if every state is both accessible and coaccessible. As is well known, every rational language is accepted by a (unique) minimal automaton.

We first establish some elementary facts about the minimal automaton

$$\mathcal{A} = (Q, A, \cdot\,, q_0, F)$$

of a non-empty language $L \subset A^n$. Set, for $i \geq 0$,

$$Q_i = \{q \in Q \mid \text{there exists } u \in A^i \text{ such that } q_0\cdot u = q\} \text{ and } k_i = \operatorname{Card} Q_i.$$

Then we can state

Proposition 1. *The following properties hold :*

(1) *the family $(Q_i)_{0\leq i\leq n}$ is a partition of Q,*

(2) *$Q_0 = \{q_0\}$ and $Q_n = \{q_f\}$, where q_f is the unique final state of $\mathcal{A}$,*

(3) *for $0 \leq i \leq n-1$, $Q_{i+1} = Q_i\cdot a \cup Q_i\cdot b$,*

(4) *for $0 \leq i \leq n-1$, $(k_i + 1) \leq (k_{i+1} + 1)^2$.*

Proof (1) Since L is non empty, it contains a word $u = a_1 \cdots a_n$. Now, for $0 \leq i \leq n$, $q_0\cdot a_1 \cdots a_i \in Q_i$, and hence Q_i is non empty. Assume that $Q_i \cap Q_j$ is not empty and let $q \in Q_i \cap Q_j$. Then there exists a word u of length i and a word v of length j such that $q_0\cdot u = q$ and $q_0\cdot v = q$. Since $\mathcal{A}$ is minimal, the state q is coaccessible, and hence there exists a word w such that $q\cdot w$ is a final state. It follows that $uw, vw \in L$ and thus $|uw| = |vw| = n$. Therefore $i = |u| = |v| = j$ and the Q_i's are pairwise disjoint.

We claim that Q_i is empty for $i > n$. Indeed, let $q \in Q_i$. Then by definition, $q = q_0\cdot u$ for some word u of length $> n$. Thus q is not coaccessible, a contradiction. Now $Q = \bigcup_{i\geq 0} Q_i$ and it follows that the family $(Q_i)_{0\leq i\leq n}$ is a partition of Q.

(2) The equality $Q_0 = \{q_0\}$ is clear. Let $q \in Q_n$. Then there exists a word u of length n such that $q_0\cdot u = q$ and a word w such that $q\cdot w \in F$ (since q is coaccessible). Thus $uw \in L$ and hence $|uw| = n$. It follows that $w = 1$, $u \in L$ and $q \in F$. Let q' be another final state. Then $q_0\cdot u' = q'$ for some $u' \in L$. Let $v \in A^*$. Then $q\cdot v \in F$ (resp. $q'\cdot v \in F$) if and only if $v = 1$. It follows that $q = q'$ since $\mathcal{A}$ is reduced.

(3) is obvious.

(4) For a given $q \in Q_i$, either $q\cdot a \in Q_{i+1}$ or $q\cdot a$ is undefined; this gives $(k_{i+1} + 1)$ possibilities. Similarly, there are $(k_{i+1} + 1)$ possibilities for $q\cdot b$. Furthermore, since q is coaccessible, either $q\cdot a$ or $q\cdot b$ is defined. Finally, this gives $(k_{i+1} + 1)^2 - 1$ possibilities for the pair $(q\cdot a, q\cdot b)$. But since $\mathcal{A}$ is reduced, two distinct states q and q' cannot have the same image under a and b. Thus $k_i \leq (k_{i+1} + 1)^2 - 1$. □

Corollary 2. *For $0 \leq i \leq n$, $k_i \leq \min(2^i, 2^{2^{n-i}} - 1)$.*

Proof We make use of Proposition 1. By (2), $k_0 = 1$ and by (3), $k_{i+1} \leq 2k_i$ for $0 \leq i \leq n-1$. Thus $k_i \leq 2^i$ by induction on i. Similarly, $k_n = 1$ by (2) and $k_i \leq (k_{i+1}+1)^2 - 1$ by (4). Thus $k_i \leq 2^{2^{n-i}} - 1$ by induction on $n-i$. □

Set $g(n) = \sum_{0 \leq i \leq n} \min(2^i, 2^{2^{n-i}} - 1)$. Since the family $(Q_i)_{0 \leq i \leq n}$ is a partition of Q, we have

$$\operatorname{Card} Q = \sum_{0 \leq i \leq n} \operatorname{Card} Q_i \leq g(n).$$

Therefore, we have proved

Proposition 3. *The minimal automaton of a language $L \subseteq A^n$ has at most $g(n)$ states. Therefore $f(n) \leq g(n)$.*

Our main result states that the opposite inequality also holds.

Theorem 4. *For every $n \geq 0$, $f(n) = g(n)$.*

Proof The result is trivial if $n = 0$. We assume now $n > 0$. By Proposition 3, it suffices to exhibit a minimal automaton with $g(n)$ states that accepts a language $L \subseteq A^n$. Let x be the unique positive real number such that $n = 2^x + x$, and let $k = \lceil 2^x \rceil$. The following lemma gives the property for which k was selected.

Lemma 5. *Let j be a positive integer.*

(1) *If $j < k$, then $2^j < 2^{2^{n-j}} - 1$.*

(2) *If $j \geq k$, then $2^j > 2^{2^{n-j}} - 1$.*

Proof (1) If $j < k$, then $j < 2^x$ and $x < n-j$ by the definition of x. Thus $j < 2^x < 2^{n-j}$ and hence $j + 1 \leq 2^{n-j}$. Now if $j > 0$, $2^{n-j} \geq j+1 \geq 2$ and if $j = 0$, $2^{n-j} = 2^n \geq 2$, since $n > 0$. Thus $2^{n-j} \geq 2$ in any case and $2^j \leq 2^{2^{n-j}-1} < 2^{2^{n-j}} - 1$.

(2) If $j \geq k$, then $j \geq 2^x$ and $x \geq n - j$ by the definition of x. Thus $2^j \geq 2^{2^x} \geq 2^{2^{n-j}} > 2^{2^{n-j}} - 1$. □

We now construct a complete automaton $\mathcal{A} = (Q, A, \cdot, q_0, \{q_f\})$ as follows. The set Q is the disjoint union of a sink state 0 and of $(n+1)$ sets Q_i $(0 \leq i \leq n)$ such that

(a) $Q_0 = \{q_0\}$ and $Q_n = \{q_f\}$,

(b) for $0 \leq i < k$, $\operatorname{Card} Q_i = 2^i$,

(c) for $k \leq i \leq n$, $\operatorname{Card} Q_i = 2^{2^{n-i}} - 1$,

and the transitions satisfy the following conditions

(d) $0 \cdot a = 0$ and $0 \cdot b = 0$,

(e) for $0 \leq i < k-1$, $\{q \cdot c \mid q \in Q_i \text{ and } c \in \{a, b\}\} = Q_{i+1}$

(since $\operatorname{Card} Q_{i+1} = 2^{\operatorname{Card} Q_i}$, this implies that all the states $q \cdot c$, where $q \in Q_i$ and $c \in \{a, b\}$, are distinct).

(f) for $k - 1 \leq i < n$ and $q, q' \in Q_i$, (f1) $(q\cdot a, q\cdot b) \in ((Q_{i+1} \cup \{0\}) \times (Q_{i+1} \cup \{0\})) \setminus \{(0,0)\}$,

(f2) $(q\cdot a, q\cdot b) = (q'\cdot a, q'\cdot b)$ implies $q = q'$,

(f3) for every $s \in Q_{i+1}$, there exists $t \in Q_i$ such that $t\cdot a = s$ or $t\cdot b = s$.

To ensure that condition (f) can be satisfied, it suffices to verify that, for $k-1 \leq i \leq n$,

$$\mathrm{Card}\, Q_i \leq (1 + \mathrm{Card}\, Q_{i+1})^2 - 1 \text{ and } \mathrm{Card}\, Q_{i+1} \leq 2\,\mathrm{Card}\, Q_i.$$

Both conditions are trivially satisfied for $i \geq k$, and follow from Lemma 5 for $i = k-1$.

We derive from $\mathcal{A}$ an uncomplete automaton $\mathcal{B}$ by removing the sink state 0 and all the transitions of the form $q\cdot a = 0$ or $q\cdot b = 0$. $\mathcal{B}$ is now an automaton with $g(n)$ states in which every state is accessible and coaccessible (by conditions (e) and (f3)). Furthermore $\mathcal{B}$ is reduced (by conditions (e) and (f2)) and hence minimal. Finally, as required, every word accepted by $\mathcal{B}$ has length n.

Example 1. Let $n = 5$. Then $g(5) = 1+2+4+8+3+1 = 19$ and $k = 4$. An automaton with 19 states recognizing a set of words of length 5 is represented in Figure 1 (next page).

The behaviour of $g(n)$ when n tends to infinity is given by the following theorem.

Theorem 6. *The following formula holds*

$$1 = \underline{\lim}_{n\to\infty} ng(n)/2^n \leq \overline{\lim}_{n\to\infty} ng(n)/2^n = 2.$$

Proof It follows from Lemma 5 that

$$g(n) = \sum_{0\leq j\leq k-1} 2^j + \sum_{k\leq j\leq n} (2^{2^{n-j}} - 1) = T_1 + T_2$$

where

$$T_1 = 2^k + 2^{2^{n-k}} \text{ and } T_2 = -2 + \sum_{k+1\leq j\leq n} (2^{2^{n-j}} - 1).$$

We first study T_2. If $j \geq k+1 \geq 2^x + 1$, then $n - j \leq x - 1$, whence $2^{n-j} \leq 2^{x-1} = 2^x/2 \leq n/2$, and therefore $2^{2^{n-j}} - 1 \leq 2^{n/2}$. Thus $-2 \leq T_2 \leq n2^{n/2}$ and it follows that $\lim_{n\to\infty} nT_2/2^n = 0$.

We now come back to T_1. Put $d = x - \lfloor x \rfloor$. By the definition of x, $n = \lfloor x \rfloor + \lceil 2^x \rceil$ and hence $n - k = x - d$ and $k = 2^x + d$. Therefore $T_1 = 2^{2^x + d} + 2^{2^{x-d}}$ and

$$nT_1/2^n = (x + 2^x)T_1/2^x 2^{2^x} = (1 + x2^{-x})(2^d + 2^{(2^{-d}-1)2^x}).$$

Since $x + 2^x = n \leq 2\cdot 2^x$, we have $x \leq \log_2 n$ and $2^{-x} \leq 2/n$. Consequently, one has $1 \leq 1 + x2^{-x} \leq 1 + 2\log_2 n/n$, and thus $\lim_{n\to\infty}(1 + x2^{-x}) = 1$. It remains to study the term $T(n) = 2^d + 2^{(2^{-d}-1)2^x}$. To start with, since $0 \leq d < 1$, we have $1 \leq 2^d \leq 2$, whence

$$\underline{\lim}_{n\to\infty} ng(n)/2^n \geq 1.$$

Let ε be a real number such that $0 < \varepsilon < 1$. We claim that the inequality $T(n) \leq 1+\varepsilon$ holds for infinitely many n. This will be a consequence of the following lemma.

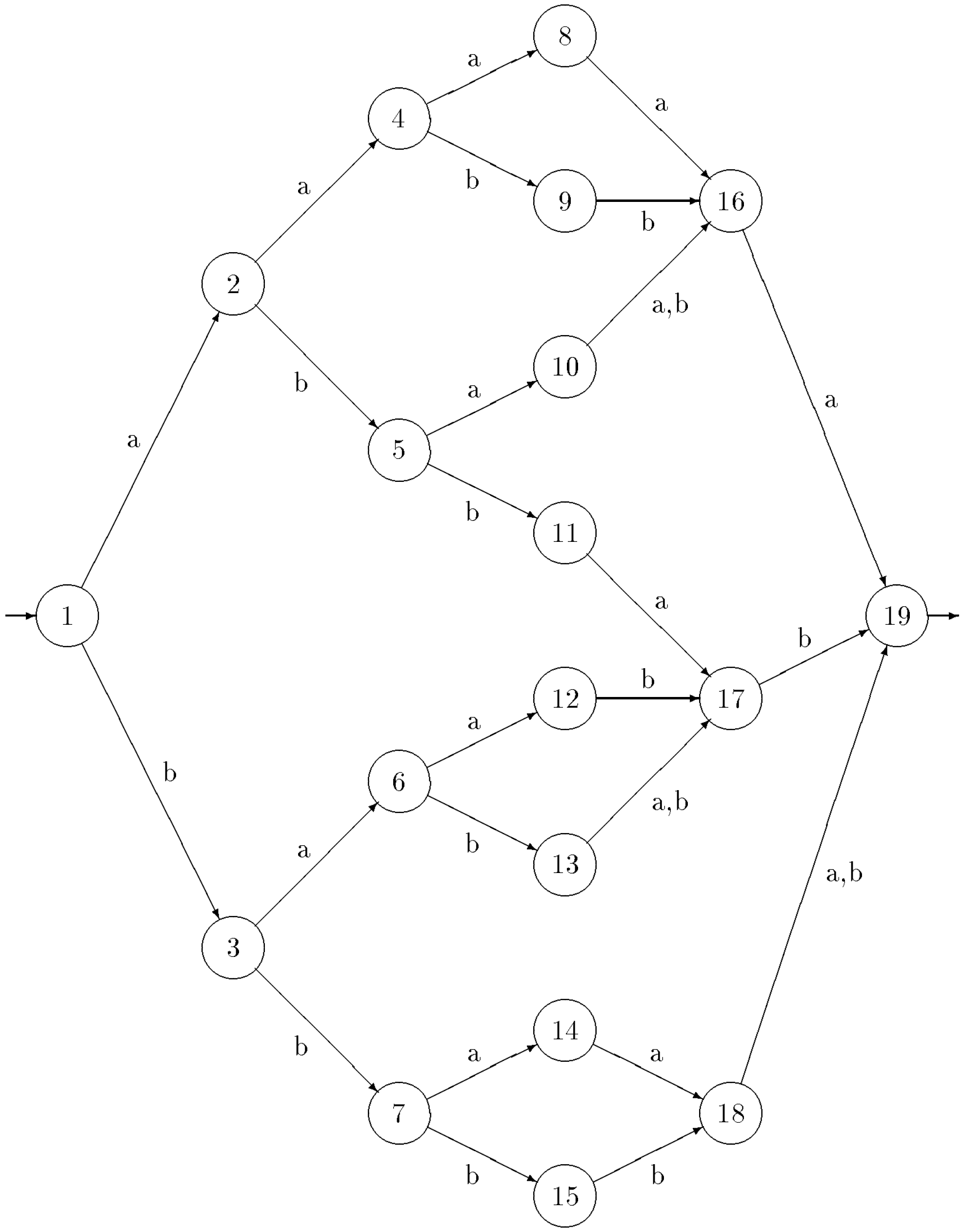

Figure 1.

Lemma 7. *Let ε_1, ε_2 be two real numbers such that $0 < \varepsilon_1 < \varepsilon_2 < 1$. Then there exists an integer r_0 such that, for every $r \geq r_0$, there exists a real number δ such that*

(a) $\varepsilon_1 < \delta < \varepsilon_2$, *and*

(b) $m = r + \delta + 2^{r+\delta}$ *is an integer.*

Proof We take $r_0 \geq \log_2[(2 - (\varepsilon_2 - \varepsilon_1))/(2^{\varepsilon_2} - 2^{\varepsilon_1})]$, so that, for every $r \geq r_0$,

$$(r + \varepsilon_2 + 2^{r+\varepsilon_2}) - (r + \varepsilon_1 + 2^{r+\varepsilon_1}) \geq 2.$$

Now, since the function $t \to r + t + 2^{r+t}$ is monotone, there exists a real δ with $\varepsilon_1 < \delta < \varepsilon_2$ such that $m = r + \delta + 2^{r+\delta}$ is an integer. $\square$

To prove the claim, we apply the lemma with $\varepsilon_1 = -\log_2(1-\varepsilon/3)$ and $\varepsilon_2 = \log_2(1+\varepsilon/2)$. One verifies easily that the condition $0 < \varepsilon_1 < \varepsilon_2 < 1$ is satisfied. Then, for any large enough r, there exists an integer $m < r$ and a real δ with $\varepsilon_1 < \delta < \varepsilon_2$ such that

$$T(m) = 2^{\delta} + 2^{(2^{-\delta}-1)2^r} \leq 2^{\varepsilon_2} + 2^{(2^{-\varepsilon_1}-1)2^r} \leq 1 + \varepsilon/2 + 2^{-(\varepsilon/3)2^k}.$$

Thus if $r \geq \log_2((3/\varepsilon)\log_2(2/\varepsilon))$, then $2^{-(\varepsilon/3)2^k} \leq \varepsilon/2$ and $T(m) \leq 1+\varepsilon$, proving the claim. It follows that $\underline{\lim}_{n\to\infty} T(n) \leq 1$, whence $\underline{\lim}_{n\to\infty} ng(n)/2^n = 1$.

On the other hand, $2^{-d} - 1 \leq -1/3d$ and thus $T(n) \leq 2^d + 2^{-(d2^x)/3}$. Let $0 < \varepsilon < 1/3$. Then for $n > -6\log_2 \varepsilon$, we have

$$-6\log_2 \varepsilon < n = x + 2^x < 2^x + 2^x$$

and hence $-2^x/3 < \log_2 \varepsilon$. Setting $y = 2^d$, we obtain $T(n) \leq y + y^{\log_2 \varepsilon}$, where $1 \leq y \leq 2$. But a short calculation shows that, on this interval, the function $t \to t + t^{\log_2 \varepsilon}$ reaches its maximum for $t = 2$. Therefore $T(n) \leq 2 + \varepsilon$ for every $\varepsilon > 0$ and

$$\overline{\lim_{n\to\infty}}\, ng(n)/2^n \leq 2.$$

Finally, let $0 < \varepsilon < 1$ and put $\varepsilon_1 = \log_2(2-\varepsilon)$ and $\varepsilon_2 = (1+\varepsilon_1)/2$. Then $0 < \varepsilon_1 < \varepsilon_2 < 1$, and by Lemma 7, there exists infinitely many integers m such that $m = r + \delta + 2^{r+\delta}$ with $\varepsilon_1 < \delta < \varepsilon_2$ and

$$T(m) = 2^{\delta} + 2^{(2^{-\delta}-1)2^r} \geq 2^{\delta} > 2^{\varepsilon_1} = 2 - \varepsilon.$$

Therefore $\overline{\lim}_{n\to\infty} T(n) \geq 2 - \varepsilon$ for every $\varepsilon > 0$ and hence $\overline{\lim}_{n\to\infty} ng(n)/2^n = 2$. $\square$

Reference.

[1] S. Eilenberg, Automata, Languages and Machines, Vol. A, (1974), Academic Press, New-York.