Journal of Automata, Languages and Combinatorics **6** (2001) 4, x–y

TIGHT BOUNDS ON THE SIMULATION OF UNARY PROBABILISTIC AUTOMATA BY DETERMINISTIC AUTOMATA [1, 2]

MASSIMILIANO MILANI
Laboratoire d'Informatique Algorithmique: Fondements et Applications,
Université Paris 7 – Denis Diderot
2, Place Jussieu F-75251 Paris Cedex 05, France
e-mail: `milani@liafa.jussieu.fr`

and

GIOVANNI PIGHIZZINI
Dipartimento di Scienze dell'Informazione, Università degli Studi di Milano
via Comelico 39, I-20135 Milano, Italy
e-mail: `pighizzini@dsi.unimi.it`

ABSTRACT

In 1963, RABIN proved that any n-state probabilistic automaton, accepting a language with a δ-isolated cut-point, can be simulated by a deterministic automaton with no more than $(1+1/\delta)^{n-1}$ states. In this paper, we improve this result in the unary case, namely in the case of automata and languages defined over a one letter input alphabet. We show that any unary n-state probabilistic automaton with an isolated cut-point can be simulated by a deterministic automaton with $O(\mathrm{e}^{\sqrt{n \ln n}})$ states in the cyclic part. Furthermore, we show that this result is tight.

We also study the number of states of the initial path of the minimum deterministic automaton which simulates the given n-state probabilistic automaton. We show that this number cannot be bounded by any function of the only parameter n.

Keywords: unary automata, probabilistic automata, descriptional complexity, state complexity.

1. Introduction

Probabilistic models were introduced at the very beginning of information and computer science. In particular, these models are the basis of information theory (see,

[1]Full version of a submission presented at the Second International Workshop on *Descriptional Complexity of Automata, Grammars and Related Structures* held in London, Ontario, Canada, July 27 – 29, 2000.

[2]Partially supported by MURST, under the project "Modelli di calcolo innovativi: metodi sintattici e combinatori".

e.g., the pioneering work of Shannon and Weaver [15]). A systematic study of *probabilistic automata*, as a generalization of deterministic ones, started in early sixties. Rabin [14] motivated this investigation by observing that devices, which are usually intended to be deterministic, in the reality can exhibit stochastic behaviors. For instance, consider sequential circuits which can have random malfunctioning of the components.

While the evolution of a deterministic automaton is governed by a transition function, which associates with every state and input symbol a new state, in the probabilistic case the evolution is stochastic, namely the transition from a state to another state on an input symbol is performed with a certain probability. In this way, probabilistic automata generalize deterministic ones. The language accepted by a probabilistic automaton can be defined as the set of words whose acceptance probability is greater than a given threshold, which is usually called *cut-point*. It is well-known that the class of languages accepted by probabilistic automata, i.e., the class of so called *stochastic languages*, strictly contains the class of regular languages.

Due to the stochastic nature of probabilistic models, the outcome produced by a probabilistic experiment can be affected by errors. Then, from a practical point of view, it is important to perform experiments with small error probabilities.

To this aim, in the field of probabilistic automata, *isolated* cut-points are very interesting (the formal definition of this notion will be recalled in Section 2 of the paper). In fact, if an automaton recognizes a language with an isolated cut-point, then the error probability can be reduced below any given value ϵ, by simply iterating the probabilistic experiment for a suitable number of times, which depends only on ϵ.

In 1963, Rabin proved that the class of languages accepted by probabilistic automata with isolated cut-points coincides with the class of regular languages [14]. In this paper, we give a new proof of this result in the unary case, i.e., in the case of languages and automata defined over a one letter input alphabet. Moreover, we study the relationships between the complexities of the descriptions of unary regular languages by probabilistic and by deterministic automata.

We briefly recall that the transition graph of a unary deterministic automaton consists of a path, which starts from the initial state, and it is followed by a simple cycle of states. Our main result, presented in Section 3 and in Section 4, is a tight upper bound on the number of states in the cycle. In particular, we show that any n-state probabilistic automaton accepting, with an isolated cut-point, a unary language, can be simulated by a deterministic automaton with at most $O(\mathrm{e}^{\sqrt{n \ln n}})$ states in the cycle. Furthermore, for $n \geq 1$, we exhibit a probabilistic automaton witnessing the optimality of this upper bound. We point out that, as proved in [3, 12], $O(\mathrm{e}^{\sqrt{n \ln n}})$ is also the cost of the optimal simulations of one-way nondeterministic, two-way deterministic, and two-way nondeterministic unary automata by one-way deterministic ones.

Subsequently, we study the number of states of the initial path. We show that this number cannot be bounded with respect to the number of states of the given probabilistic automaton. In fact, we prove that for any integer n, there exists a unary language accepted by a probabilistic automaton with two states, such that any deterministic automaton accepting it requires at least n states. This improves a similar result obtained in [11], where a language defined over a binary alphabet is used.

We conclude the paper by pointing out, in Section 5, some final remarks and open problems.

2. Preliminaries

In this section, we recall basic notions, notations and facts used in the paper.

Given a set S, $\sharp S$ denotes its cardinality, and 2^S the family of all its subsets. The absolute value of a number z is denoted by $|z|$. For any $z > 0$, $\ln z$ denotes the natural logarithm of z. The *greatest common divisor* of integers $a_1, \ldots, a_s$ writes as $\gcd(a_1, \ldots, a_s)$. Their *least common multiple* is denoted by $\mathrm{lcm}(a_1, \ldots, a_s)$.

Given an alphabet Σ, Σ^* denotes the set of strings on Σ, with the empty string ϵ. Given a string $x \in \Sigma^*$, $|x|$ denotes its length. A language $\mathcal{L}$ is said to be *unary* (or *tally*) whenever it can be built over a *single letter* alphabet. In this case, we let $\mathcal{L} \subseteq 1^*$.

Let us take a brief look on the computational model of finite automata. For a detailed exposition, we refer the reader to [7]. A *one-way nondeterministic finite automaton* (nfa) over an input alphabet Σ is a 4-tuple $A = (Q, \delta, q_0, F)$, where Q is the finite set of states, $\delta : Q \times \Sigma \to 2^Q$ is the transition function, $q_0 \in Q$ is the initial state, $F \subseteq Q$ is the set of final states. The transition function δ can be extended to strings in a standard way. The language accepted by A is the set $\mathcal{L}(A) = \{x \in \Sigma^* \mid \delta(q_0, x) \cap F \neq \emptyset\}$. The one-way automaton A is *deterministic* (dfa) if and only if $\sharp\delta(q, \sigma) = 1$, for any $q \in Q$, and $\sigma \in \Sigma$ (this implies that deterministic automata are assumed to be *complete*).

All automata we will consider in this paper are *unary*, namely, they are defined over a one letter input alphabet $\Sigma = \{1\}$. It is easy to observe that the transition graph of a unary dfa consists of a path, which starts from the initial state, and it is followed by a cycle of one or more states (see Figure 1).

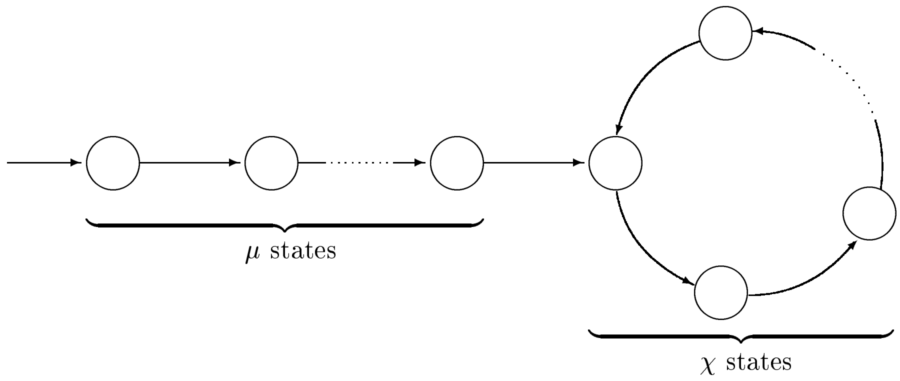

Figure 1: A unary dfa

As in [3], the *size* of a dfa A is the pair (χ, μ), where $\chi \geq 1$ and $\mu \geq 0$ denote the number of states which belong to the cycle and to the path, respectively.

Observing the form of unary dfa's, it is not difficult to conclude that unary regular languages correspond to *ultimately periodic sets* of integers:

Theorem 1 *Given a unary regular language $\mathcal{L}$ and two integers $\chi \geq 1$, $\mu \geq 0$, the following statements are equivalent:*

(i) *$\mathcal{L}$ is accepted by a dfa of size (χ, μ);*

(ii) *for any integer $m \geq \mu$, $1^m \in \mathcal{L}$ if and only if $1^{m+\chi} \in \mathcal{L}$.*

A unary dfa is said to be *cyclic* when its graph is a cycle, i. e., when it is accepted by a dfa of size $(\chi, 0)$, for some $\chi \geq 1$. To emphasize the periodicity of the accepted language $\mathcal{L}$, we say also that $\mathcal{L}$ is χ*-cyclic.* A language accepted by a unary dfa of size (χ, μ) is said to be *ultimately* χ*-cyclic.*

As observed above, the transition graph of a unary dfa consists of a path which is followed by a cycle. On the other hand, transition graphs of nondeterministic unary automata do not have any particular structure. However, with at most a quadratic increasing in the number of states, it is possible to transform any unary nfa, into an equivalent nfa, whose transition graph has a very particular form. More precisely:

Definition 2 [3, Def. 4.2] *A unary nfa $A = (Q, \delta, q_0, F)$ is in* Chrobak normal form *if Q can be partitioned into $k+1$ disjoint sets $Q_0, C_1, \ldots, C_k$ such that:*

- $Q_0 = \{q_0, q_1, \ldots, q_m\}$,
- *for* $i = 1, 2, \ldots, k$, $C_i = \{p_{i,0}, p_{i,1}, \ldots, p_{i,y_i-1}\}$,
- *for* $i = 1, 2, \ldots, k$, *and* $j = 0, 1, \ldots, y_i - 1$, $\delta(p_{i,j}, 1) = \{p_{i,(j+1) \,\mathrm{MOD}\, y_i}\}$, *i. e., C_i is a cycle of length y_i,*
- *for* $i = 0, 1, \ldots, m-1$, $\delta(q_i, 1) = \{q_{i+1}\}$, *i. e., Q_0 is an initial path of length m,*
- $\delta(q_m, 1) = \{p_{1,0}, p_{2,0}, \ldots, p_{k,0}\}$.

In Figure 2, a nfa in Chrobak normal form is depicted.

Theorem 3 [3, Lemma 4.3] *For each unary n-state nfa A there exists an equivalent nfa A' in Chrobak normal form with $O(n^2)$ states in the initial path and at most n states in the cycles.*

We now recall basic notions related to probabilistic automata. Roughly speaking, *probabilistic automata* (pfa) can be obtained from deterministic ones, by substituting the transition function with a probability distribution, i. e., with a function

$$\delta : Q \times \Sigma \times Q \to [0, 1],$$

such that, for any $q \in Q$, $\sigma \in \Sigma$, it holds that $\sum_{p \in Q} \delta(q, \sigma, p) = 1$.

The value $\delta(q, \sigma, p)$, for $q, p \in Q$, $\sigma \in \Sigma$, is the probability that the automaton, in the state q, by reading the input symbol σ, reaches the state p. As in the deterministic

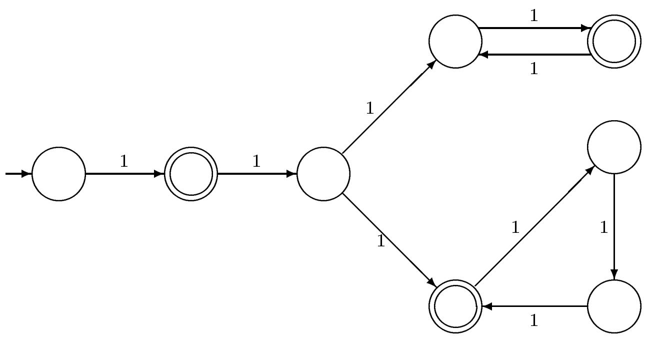

Figure 2: A nfa in Chrobak normal form

case, the computation starts in an initial state q_0. A computation is accepting if and only if it ends in a final state. Thus, a probabilistic automaton over an input alphabet Σ can be defined as a 4-tuple $A = (Q, \delta, q_0, F)$.

Usually, to denote probabilistic automata, it is better to use an equivalent matricial form. In particular, a pfa with n states over an input alphabet Σ can be denoted as a 3-tuple $A = (\pi, \Delta, \eta)$, where $\pi \in [0,1]^n$ is a stochastic row vector, the *initial distribution*, Δ is a set of stochastic matrices, which contains, for any $\sigma \in \Sigma$, a $n \times n$ square matrix Δ_σ, the *transition matrix* associated with the symbol σ, $\eta \in \{0,1\}^n$ is the characteristic (column) vector of the set of *final states*.

A string $w = \sigma_1 \ldots \sigma_m$ is accepted by a pfa A with probability

$$Pr_A(w) = \pi \Delta_{\sigma_1} \ldots \Delta_{\sigma_m} \eta.$$

The *language* accepted by the pfa A with respect to a given *cut-point* λ, $0 \leq \lambda < 1$, is the set of strings accepted by A with probability greater than λ, i. e.,

$$\mathcal{L}_{A,\lambda} = \{w \in \Sigma^* \mid Pr_A(w) > \lambda\}.$$

It is well known that the class of *stochastic languages*, i. e., of the languages accepted by pfa's, is wider than the class of regular languages. However, as proved by Rabin in [14], when the cut-point is *isolated*, the language accepted by a pfa is regular. More precisely:

Definition 4 *Given a pfa A and two real numbers λ, δ, with $0 \leq \lambda < 1$, $\delta > 0$, λ is a δ-*isolated *cut-point for A if and only if $|Pr_A(x) - \lambda| \geq \delta$, for any $x \in \Sigma^*$.*

Theorem 5 [14] *Given an n-state pfa A, if λ is a δ-isolated cut-point for A, then the language $\mathcal{L}_{A,\lambda}$ is regular. Furthermore, the number of states of the minimum dfa accepting $\mathcal{L}_{A,\lambda}$ is at most*

$$\left(1 + \frac{1}{2\delta}\right)^{n-1}.$$

Actually, the upper bound on the number of states presented in Theorem 5, was slightly improved by Paz [13] and by Gabbasov and Murtazina [5]. However, as recently shown by Ambainis [1], it is very near to the optimum.

3. Unary Probabilistic Automata and Stochastic Matrices

In this section, we study the periodicity of unary languages accepted by pfa's with isolated cut-points. Our main result is an upper bound on this periodicity. This bound is related to the number of states of a given pfa accepting the language under consideration.

First of all, we observe that, in the unary case, there is only one transition matrix, which will be denoted as Δ. Thus, the probability associated with a string of length m depends on the m-th power of Δ, and the language $\mathcal{L}_{A,\lambda}$ accepted with respect to a given cut-point λ can be expressed as:

$$\mathcal{L}_{A,\lambda} = \{1^m \mid \pi\Delta^m\eta > \lambda\}.$$

By Theorem 5, if λ is isolated, then $\mathcal{L}_{A,\lambda}$ is regular, namely, it defines an ultimately periodic set of integers. Let $\chi \geq 1$ be the period of this set. By Theorem 1, there exists a dfa accepting $\mathcal{L}_{A,\lambda}$ whose cyclic part has exactly χ states. In order to evaluate χ, we have to study the asymptotic behavior of the powers of the transition matrix Δ. To this aim, we briefly recall the following fact (for more details, we refer the reader to [6]):

Theorem 6 *For any $n \times n$ stochastic matrix Δ, there is an integer d such that the limit*

$$\lim_{q\to\infty} \left(\Delta^d\right)^q$$

exists.

Furthermore, there are $k \geq 1$ positive integers $d_1, \ldots, d_k$, which depend on Δ, such that $d_1 + \cdots + d_k \leq n$ and $d = \mathrm{lcm}(d_1, \ldots, d_k)$.

To evaluate the constant d of Theorem 6, we consider the function

$$F(n) = \max\{\mathrm{lcm}(x_1, \ldots, x_s) \mid x_1, \ldots, x_s > 0 \text{ are integers and } x_1 + \cdots + x_s = n\}. \tag{1}$$

Evaluating the growth rate of $F(n)$ is known as LANDAU's problem [8]; several approximations of $F(n)$ are given in the literature (see, e. g., [16]). To our purposes, it is enough to know that:

Lemma 7 $F(n) = \Theta(\mathrm{e}^{\sqrt{n \ln n}})$, *i. e.,* $F(n) = O(\mathrm{e}^{\sqrt{n \ln n}})$ *and* $F(n) = \Omega(\mathrm{e}^{\sqrt{n \ln n}})$.

With this tools, we are able to prove our main result:

Theorem 8 *Given an n-state unary pfa A, let $\mathcal{L}_{A,\lambda}$ be the language accepted by A with respect to a δ-isolated cut-point λ. Then, there exists an integer $d = O(\mathrm{e}^{\sqrt{n \ln n}})$, such that, for any sufficiently large m, $1^m \in \mathcal{L}_{A,\lambda}$ if and only if $1^{m+d} \in \mathcal{L}_{A,\lambda}$, i. e., the language $\mathcal{L}_{A,\lambda}$ is ultimately d-cyclic.*

Proof. The fact that $\mathcal{L}_{A,\lambda}$ is regular, i. e., ultimately d-cyclic, for some integer $d \geq 1$, can be proved as a consequence of Theorem 5. We now give a different proof, which is based on Theorem 6. Moreover, we show that $d = O(\mathrm{e}^{\sqrt{n \ln n}})$.

Given the transition matrix Δ of A, let d be the constant given in Theorem 6. Moreover, let Δ^∞ denotes the limit matrix $\lim_{q\to\infty} \left(\Delta^d\right)^q$. For any integer m, we can write

$$\Delta^m = \Delta^{m \bmod d} \cdot \left(\Delta^d\right)^{\lfloor m/d \rfloor}.$$

If m is sufficiently large, then the matrix $\left(\Delta^d\right)^{\lfloor m/d \rfloor}$ is very close to the limit matrix Δ^∞. Thus, Δ^m can be approximated with the matrix $\Delta^{m \bmod d}\Delta^\infty$, which will be denoted as $\Delta_{m \bmod d}$. In other words, there are d "limit matrices" $\Delta_0, \Delta_1, \ldots, \Delta_{d-1}$, such that, for $j = 0, \ldots, d-1$,

$$\Delta_j = \Delta^j \Delta^\infty = \Delta^j \cdot \lim_{q\to\infty} \left(\Delta^d\right)^q = \lim_{q\to\infty} \Delta^{j+dq}.$$

Hence, there exists an integer μ, such that for any $m \geq \mu$, it holds that

$$|\pi\Delta^m\eta - \pi\Delta_{m \bmod d}\eta| < \delta.$$

Since the cut-point λ is δ-isolated, i. e., $|\pi\Delta^m\eta - \lambda| \geq \delta$, this implies that

$$\pi\Delta^m\eta > \lambda \text{ if and only if } \pi\Delta_{m \bmod d}\eta > \lambda.$$

Since $(m+d) \bmod d = m \bmod d$, the reader can easily conclude that for any $m \geq \mu$:

$$\pi\Delta^m\eta > \lambda \text{ if and only if } \pi\Delta^{m+d}\eta > \lambda,$$

or, in other words, that

$$1^m \in \mathcal{L}_{A,\lambda} \text{ if and only if } 1^{m+d} \in \mathcal{L}_{A,\lambda}.$$

Thus, the language $\mathcal{L}_{A,\lambda}$ is ultimately d-cyclic. Moreover, from Lemma 7, we get the upper limit $d = O(\mathrm{e}^{\sqrt{n \ln n}})$. □

4. Probabilistic versus Deterministic Unary Automata

In this section, we study the cost, in terms of states, of the simulation of unary pfa's by dfa's. First of all, we are able to evaluate the number of the states in the cycle of a dfa which simulates a given unary pfa:

Corollary 9 *Any n-state unary pfa which accepts a language with respect to an isolated cut-point can be simulated by a dfa with $O(\mathrm{e}^{\sqrt{n \ln n}})$ states in the cycle.*

Proof. The result is an immediate consequence of Theorem 1 and of Theorem 8. □

We point out that the cost $O(\mathrm{e}^{\sqrt{n \ln n}})$, given in Corollary 9, coincides with the optimal costs of the simulations of nfa's, two-way dfa's, and two-way nfa's by dfa's presented in [3] and [12].

Now, we will prove that the result stated in Corollary 9 (and hence also the result in Theorem 8) is tight, namely, for any integer n there exists an n-state unary pfa which is simulated by a *minimum* dfa with $\Theta(e^{\sqrt{n \ln n}})$ states in the cyclic part. Subsequently, we will show that the number of states of the initial path can be unbounded with respect to n.

Theorem 10 *There exists a sequence of unary languages $\mathcal{L}_n$ ($n = 1, 2, 3, \ldots$) such that, for each $n \geq 1$:*

(i) *there exists a pfa with n states which recognizes $\mathcal{L}_n$;*

(ii) *any minimal nfa which recognizes $\mathcal{L}_n$ has $\Omega(e^{\sqrt{n \ln n}})$ states;*

(iii) *the minimum dfa which recognizes $\mathcal{L}_n$ has $\Theta(e^{\sqrt{n \ln n}})$ states.*

Proof. For $n \geq 1$, we consider the language $\mathcal{L}_n = \left(1^{F(n)}\right)^*$, where $F(n)$ is the function defined in (1). Note that $\mathcal{L}_n$ is an $F(n)$-cyclic language. Consider the tuple of integers $(d_1, \ldots, d_k)$, for which the maximum in the definition of $F(n)$ is obtained, i.e., $F(n) = \mathrm{lcm}(d_1, \ldots, d_k)$ and $d_1 + \cdots + d_k = n$.

We now informally describe an n-state pfa A_n accepting the language $\mathcal{L}_n$. The transitions of A_n are deterministic. In particular, the states are arranged in k disjoint cycles $C_1, \ldots, C_k$, of lengths $d_1, \ldots, d_k$, respectively. On each cycle C_j, $j = 1, \ldots, k$, a state i_j is chosen as initial and final state, while all the other states are nonfinal. Hence, the cycle C_j can be seen as a d_j-cyclic dfa that accepts the language $(1^{d_j})^*$. Finally, the pfa has an initial distribution which associates probability $1/k$ with the initial states of the cycles, namely the states $i_1, \ldots, i_k$, and null probabilities with the remaining states. In Figure 3 the pfa A_n is represented.

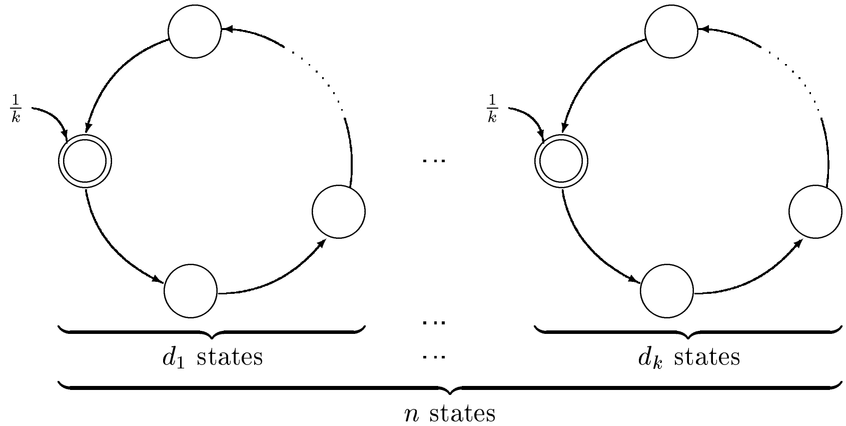

Figure 3: The pfa A_n

Consider the input string 1^m of length m. If m is a multiple of $F(n)$, i.e., a multiple of each d_j, then this string is accepted with probability 1. On the other hand, if $F(n)$ does not divide m, then there is at least one cycle C_j which does not accept 1^m. Thus, the probability associated with 1^m cannot exceed $1 - \frac{1}{k}$. Hence, given $\lambda = 1 - \frac{1}{2k}$, it is easy to conclude that λ is a $\frac{1}{2k}$-isolated cut-point for A_n, and $\mathcal{L}_n = \mathcal{L}_{\lambda,A_n}$. This proves (i).

To prove (ii), observe that the shortest nonempty word of $\mathcal{L}_n$ is $1^{F(n)}$. Hence, any nfa accepting L_n must contain a path of at least $F(n)$ pairwise different states between the initial state and a final state. Hence, the result is a consequence of the lower estimation of $F(n)$ given in Lemma 7.

Finally, to prove (iii), it is enough to observe that $\mathcal{L}_n$ is $F(n)$-cyclic, and to consider Theorem 1, Lemma 7, and statement (ii) of this theorem. □

In [11], it was proved that, for any integer $k \geq 1$, the language $L_k = \{w\sharp w \mid w \in \{0,1\}^k\}$ is accepted by a 68-state pfa, with an isolated cut-point. On the other hand, any nfa and any dfa accepting L_k must have at least 2^k states. As a consequence, the number of states of a dfa equivalent to a given n-state pfa cannot be bounded with respect to n. We now show that this is true even if we restrict our attention to *unary* automata. In particular, as already anticipated, we prove that the number of states on the initial path of a unary dfa which simulates a given unary n-state pfa can be unbounded with respect to n:

Theorem 11 *For any integer $k \geq 1$, there exists a language $\mathcal{L}_k$ such that:*

(i) *$\mathcal{L}_k$ is accepted by a 2-state pfa with an isolated cut-point;*

(ii) *the size of the minimum dfa which accepts $\mathcal{L}_k$ is $(1,k)$.*

Proof. Given a probability p, with $0 < p < 1$, consider the pfa A defined as:

$$A = \left(\pi = \begin{bmatrix} 1 & 0 \end{bmatrix},\ \Delta = \begin{bmatrix} 1-p & p \\ 0 & 1 \end{bmatrix},\ \eta = \begin{bmatrix} 0 \\ 1 \end{bmatrix}\right).$$

Note that, for any integer $m \geq 0$:

$$\Delta^m = \begin{bmatrix} (1-p)^m & 1-(1-p)^m \\ 0 & 1 \end{bmatrix},$$

and then $Pr_A(1^m) = 1-(1-p)^m$. Moreover, for $m \geq 1$, it holds that $Pr_A(1^m) > Pr_A(1^{m-1})$. Given $k \geq 1$, consider the number

$$\lambda_k = \frac{Pr_A(1^k) + Pr_A(1^{k-1})}{2}.$$

It is immediate to verify that λ_k is an isolated cut-point for A. Furthermore, $Pr_A(1^m) > \lambda_k$ if and only if $m \geq k$, namely, the language $\mathcal{L}_{A,\lambda_k}$ accepted by A with respect to λ_k coincides with the language $\mathcal{L}_k = 1^k 1^*$. The minimum dfa accepting this language has size $(1,k)$ (see Figure 4). □

5. Conclusions and Open Problems

We conclude the paper by presenting some final remarks and some open problems.

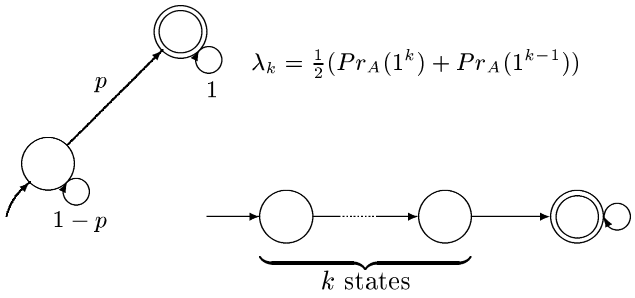

Figure 4: A 2-state pfa and the minimum dfa both recognizing $\mathcal{L}_k = 1^k 1^*$

Concerning the simulation of pfa's (with isolated cut-points) via dfa's, we point out that RABIN's Theorem (Th. 5) gives an upper limitation on the number of states of the minimum dfa that recognizes the same language which is accepted by a given n-state pfa, with respect to a δ-isolated cut-point λ. This limitation is a function of both δ and λ.

In Corollary 9, we proved that, in the unary case, the number of states in the cyclic part of the minimum dfa which simulates the given pfa is bounded by a function that does not depend on δ. As shown in Theorem 11, the same result does not hold for the initial path. Hence, at the moment, the only limitation we know on the number of states of the initial path derives from the result of RABIN. It is an open problem whether or not this limitation can be improved in the unary case.

Now, we consider simulations between probabilistic and nondeterministic automata.

In Theorem 10, we have shown that for every integer n there exists a unary language $\mathcal{L}_n$ which is accepted by an n-state pfa, with an isolated cut-point, but which requires a superpolynomial number of states to be accepted by nfa's.

Now, we examine the converse problem, namely the problem of simulating a given nfa with a pfa. This problem can be trivially solved by considering the same set of states and by substituting nondeterministic with probabilistic choices, i. e., by associating nonnull probabilities exactly with all transitions of the given nfa. By taking $\lambda = 0$ as cut-point, we are able to accept the same language of the original automaton. However, this cut-point could be nonisolated. Another trivial solution, can be obtained by applying the subset construction to the given nfa, in order to get an equivalent dfa. This dfa can be seen as a pfa which accepts the same language with the $\frac{1}{2}$-isolated cut-point $\lambda = \frac{1}{2}$. In this case, the number of states of the resulting pfa can be exponential.

We now prove that, in the unary case, any nfa can be transformed, with at most a quadratic increment in the number of states, in a pfa which recognizes the same language, with an isolated cut-point:

Theorem 12 *Let A be a unary nfa with n states accepting a language $\mathcal{L}$. There exists a pfa with $O(n^2)$ states recognizing, with an isolated cut-point, the same language.*

Sketch of proof. By reducing the given nfa A to Chrobak normal form (see Definition 2 and Theorem 3), we get an equivalent nfa A' with $O(n^2)$ states in the initial path and at most n states in the cycles. This automaton performs only one nondeterministic choice, in the last state q_m of the initial path, among k different possibilities. The pfa B simulating A is obtained from A', by substituting the nondeterministic move from q_m, with k transitions which one of them having probability $\frac{1}{k}$. To recognize the same language, it is enough to choose as cut-point the value $\lambda = \frac{1}{2k}$. The reader can verify that λ is $\frac{1}{2k}$-isolated. □

Finally, we present some remarks concerning *two-way automata*, i. e., automata which are able to move the input head in two directions.

It is well-known that in the case of deterministic and nondeterministic automata, this possibility does not increase the recognizing power, i. e., two-way deterministic and nondeterministic finite automata exactly characterize the class of regular languages.

In the probabilistic case the situation is quite different. Two-way probabilistic automata (2pfa) accept, as their one-way counterpart, exactly stochastic languages [9]. However, considering only languages accepted *with isolated cut-points*, 2pfa's are strictly powerful than pfa's. In fact, as shown in 1981 by FREIVALDS [4], the language $\{0^n 1^n \mid n \geq 0\}$ is accepted by a 2pfa with an isolated cut-point, while, as shown by RABIN (see Theorem 5), in this case one-way pfa's accept only regular languages. On the other hand, in the unary case, 2pfa's and pfa's with isolated cut-points are equivalent. In fact, in 1991 KAŅEPS [10] proved that unary 2pfa's with isolated cut-points are not more powerful than dfa's, i. e., they characterize unary regular languages.

An open problem is the tight evaluation of the costs, in terms of states, of the simulation of unary 2pfa's by dfa's or some other kinds of automata. Some results in this direction are presented in [2].

References

[1] A. AMBAINIS, The complexity of probabilistic versus deterministic finite automata. In: *Algorithms and Computation, 7th International Symposium* (*ISAAC '96*). LNCS **1178**, Springer, 1996, 233–238.

[2] A. AMBAINIS, R. FREIVALDS, M. KARPINSKI, Weak and strong recognition by 2-way randomized automata. In: *Randomization and approximation techniques in Computer Science, International Workshop* (*RANDOM '97*). LNCS **1269**, Springer, 1997, 175–185.

[3] M. CHROBAK, Finite automata and unary languages. *Theoretical Computer Science* **47** (1986), 149–158.

[4] R. FREIVALDS, Probabilistic two-way machines. In: *Mathematical Foundations of Computer Science*, LNCS **118**, Springer, 1981, 33-45.

[5] N. Z. GABBASOV, T. A. MURTAZINA, Improving the estimate of Rabin's reduction theorem. *Algorithms and automata*, Kazan University, 1979, 7–19 (in Russian).

[6] F. R. GANTMACHER, *Application of theory of matrices.* Interscience Publisher, inc., New York, 1959.

[7] J. HOPCROFT, J. ULLMAN, *Introduction to automata theory, languages, and computation.* Addison-Wesley, Reading, MA, 1979.

[8] E. LANDAU, Über die Maximalordung der Permutationen gegebenen Grades. *Archiv. der Math. und Phys.* **3** (1903), 92–103.

[9] J. KAŅEPS, Stochasticity of the languages recognizable by 2-way finite probabilistic automata. *Diskretnaya Matematika* **1** (1989) 4, 63–74 (in Russian).

[10] J. KAŅEPS, Regularity of one-letter languages acceptable by 2-way finite probabilistic automata. In: *Fundamentals of Computation Theory* (*FCT '91*), LNCS **529**, Springer, 1991, 287–296.

[11] C. KINTALA, K. PUN, D. WOTSCHKE, Concise representations of regular languages by degree and probabilistic finite automata. *Mathematical Systems Theory* **26** (1993), 379–395.

[12] C. MEREGHETTI, G. PIGHIZZINI, Optimal simulations between unary automata. *SIAM J. Comp.* **30** (2001), 1976–1992.

[13] A. PAZ, Some aspects of probabilistic automata. *Information and Control* **9** (1966) 1, 26–60.

[14] M. O. RABIN, Probabilistic automata. *Information and Control* **6** (1963) 3, 230–245.

[15] C. E. SHANNON, W. WEAVER, *The mathematical theory of communication.* Univ. of Illinois Press, Urbana, Illinois, 1948.

[16] M. SZALAY, On the maximal order in S_n and S_n^*. *Acta Arithmetica* **37** (1980), 321–331.

(Received: November 16, 2000)