Theoretical Informatics and Applications
Theoret. Informatics Appl. **36** (2002) 277–291
DOI: 10.1051/ita:2002014

ON THE SIZE OF ONE-WAY QUANTUM FINITE AUTOMATA WITH PERIODIC BEHAVIORS*

Carlo Mereghetti[1] and Beatrice Palano[2]

Abstract. We show that, for any stochastic event p of period n, there exists a *measure-once one-way quantum finite automaton (1qfa)* with at most $2\sqrt{6n} + 25$ states inducing the event $ap + b$, for constants $a > 0$, $b \geq 0$, satisfying $a + b \leq 1$. This fact is proved by designing an algorithm which constructs the desired 1qfa in polynomial time. As a consequence, we get that any periodic language of period n can be accepted with isolated cut point by a 1qfa with no more than $2\sqrt{6n}+26$ states. Our results give added evidence of the strength of measure-once 1qfa's with respect to classical automata.

Mathematics Subject Classification. 68Q10, 68Q19, 68Q45.

Introduction

One of the main investigations in the field of quantum computing certainly deals with the study of the computational power of quantum devices with respect to their classical counterparts. In this sense, the results obtained by, *e.g.*, Shor [21, 22] and Grover [7] give evidences that the quantum paradigm might lead to faster algorithms. Nevertheless, it is reasonable to think that the first implementations of quantum machines will not be fully quantum mechanical. Instead, we can expect that they will consist of "expensive" quantum components embedded in classical

Keywords and phrases: Quantum finite automata, periodic events and languages.

* *Partially supported by M.I.U.R. COFIN, under the project "Linguaggi formali e automi: teoria e applicazioni". Some results in this paper were presented in a preliminary form* [16] *at the* 7th Italian Conference on Theoretical Computer Science*, Torino, Italy, Oct. 4–6, 2001.*

[1] Dipartimento di Informatica, Sist. e Com., Università degli Studi di Milano – Bicocca, Via Bicocca degli Arcimboldi 8, 20126 Milano, Italy; e-mail: mereghetti@disco.unimib.it

[2] Dipartimento di Informatica, Università degli Studi di Torino, Corso Svizzera 185, 10149 Torino, Italy; e-mail: beatrice@di.unito.it

devices (see, *e.g.* [3]). This motivates the study of the computational power of "small" quantum devices such as *quantum finite automata (qfa's)*.

The simplest version of qfa's are the *one-way qfa's (1qfa's)* which, roughly speaking, are defined by imposing the quantum paradigm – unitary evolution plus observation – to the classical model of one-way deterministic or probabilistic automata (1dfa's and 1pfa's, resp.).

Two variants of 1qfa's are considered in the literature: in the first one, called *measure-once* [5, 19], the probability of accepting strings is evaluated by "observing" 1qfa's just once, at the end of input scanning. In the *measure-many* model [4, 5, 13], instead, such an observation is performed after each move. In this work, we will be concerned *only* with *measure-once 1qfa's*. Thus, the attribute measure-once will always be understood.

The question *1qfa's vs. classical automata* is usually tackled from two points of view: the *recognizability* of languages, and the *size* – number of states – of automata when they perform certain tasks. It is well known from [5] that, quite surprisingly, the class of languages accepted by 1qfa's with isolated cut point is a proper subclass (group languages [20]) of regular languages. On the other hand, it is also well known that, in some cases, 1qfa's turn out to be more succinct than 1dfa's and 1pfa's. For instance, fix a prime p, and define the unary language $L_p = \{1^{kp} \mid k \in \mathbf{N}\}$. In [4], it is proved that accepting L_p with isolated cut point requires exactly p states on 1pfa's, while a Monte-Carlo 1qfa (a more "reliable" version of isolated cut point 1qfa, see [8,9]) with $\mathcal{O}(\log p)$ states for L_p is exhibited.

Several other results are given in the literature, that witness strength and weakness of 1qfa's (see, *e.g.* [4,8,9,11]). Almost all of them are obtained by constructing 1qfa's accepting *ad hoc* languages or solving suitably defined problems.

Here, we aim to give a *general method* for building succinct 1qfa's that have a *periodic behavior*. More precisely: the stochastic event induced by a unary (*i.e.*, with a single letter input alphabet) 1qfa $\mathcal{A}$ is the function $p : \mathbf{N} \to [0, 1]$ defined, for any $k \in \mathbf{N}$, as $p(k) =$ probability that $\mathcal{A}$ accepts the string 1^k. We are interested in designing unary 1qfa's inducing given *periodic events*, *i.e.*, events satisfying $p(k) = p(k + n)$, for a fixed period $n > 0$ and any $k \in \mathbf{N}$. Actually, we will be content with obtaining a "linear approximation" of p, that is, an event of the form $ap + b$, for some constants $a > 0$, $b \geq 0$, with $a + b \leq 1$. It is not hard to verify that, from a language acceptance point of view, the events p and $ap + b$ are fully equivalent (as explained in Sect. 2).

We prove that:

> *For any stochastic event p of period n taken as input, there exists a unary 1qfa $\mathcal{A}$ with at most $2\sqrt{6n} + 25$ states which induces $ap + b$, for some constants $a > 0$, $b \geq 0$, with $a + b \leq 1$.*

We provide an algorithm which actually constructs $\mathcal{A}$ in polynomial time. To this purpose, we first show that any event induced by a unary 1qfa has a sort of *normal form*. We then display an algorithm which, in a first phase, computes some parameters in this normal form so to reproduce the *harmonic structure* of p. In a second phase, the algorithm turns the computed parameters into a well formed

unary 1qfa $\mathcal{A}$ inducing $ap+b$ with at most $2\sqrt{6n}+25$ states. It is interesting to observe that our construction enables us to show that the size of $\mathcal{A}$ is bounded by the size of *difference covers* for $\mathbf{Z}_n$, *i.e.*, sets $\Delta \subseteq \mathbf{Z}_n$ such that each element in $\mathbf{Z}_n$ can be obtained as the difference modulo n of two elements in Δ.

This result allows us to give an upper bound on the size of 1qfa's accepting *periodic languages*. A unary languages L is said to be periodic if it can be written as $L = \{1^k \mid k \in \mathbf{N} \text{ and } (k \bmod n) \in S\}$, for a fixed $S \subseteq \mathbf{Z}_n$. The reader is referred to, *e.g.* [10,18] where the relevance of periodic languages is emphasized. We show that:

> *Any periodic language of period n can be accepted with isolated cut point by a unary 1qfa with no more than $2\sqrt{6n}+26$ states.*

Our results once more witness the strength, by a quadratic state decreasing, of 1qfa's with respect to classical automata. It is well known, for instance, that accepting n-periodic languages on 1dfa's requires at least n states. Furthermore, when n is prime, we cannot hope to save states even by using 1pfa's [4, 17] or two-way nondeterminism [18]. For a more extensive discussion on these and other topics related to the question quantum *vs.* classical devices, we refer the reader to Section 4. Here, we just notice that a quadratic saving of computational resources when using quantum instead of classical paradigm often shows up in the literature (see, *e.g.* [7,11]). It might be interesting to investigate the nature of this recurrent phenomenon.

The paper is organized as follows: in Section 1, we give basics on linear algebra, quantum finite automata, and difference cover. In Section 2, we present the polynomial time algorithm to construct a $\mathcal{O}(\sqrt{n})$-state 1qfa inducing a linear approximation of a given n-periodic stochastic event. In Section 3, we show how to recognize n-periodic languages on 1qfa's with isolated cut point and $\mathcal{O}(\sqrt{n})$ states. Finally, in Section 4, we discuss our results in the light of quantum *vs.* classical question, and we point out some possible directions for future researches.

1. Preliminaries

1.1. Linear algebra

Here, we recall some basic notions on vector spaces and linear algebra. For more details, we refer the reader to any of the standard books on the subject, such as [14,15]. Given a complex number $z \in \mathbf{C}$, its *complex conjugate* is denoted by z^*, and its *modulus* is $|z| = \sqrt{zz^*}$. Let $\mathcal{V}$ be a vector space of finite dimension n on $\mathbf{C}$. The *inner product* of vectors $x, y \in \mathcal{V}$, with $x = (x_1, x_2, \ldots, x_n)$ and $y = (y_1, y_2, \ldots, y_n)$, is defined as $\langle x, y \rangle = \sum_{i=1}^{n} x_i y_i^*$. The *norm* of x is defined as $\| x \| = \sqrt{\langle x, x \rangle}$. If $\langle x, y \rangle = 0$ (and $\| x \| = \| y \| = 1$) then x and y are *orthogonal* (*orthonormal*). A *decomposition of* $\mathcal{V}$ is a set $\{\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_k\}$ $(k \leq n)$ of mutually orthogonal subspaces of $\mathcal{V}$ such that each $x \in \mathcal{V}$ can be written as the sum of the projections of x onto each $\mathcal{S}_i$. $\mathcal{V} = \mathcal{S}_1 \oplus \mathcal{S}_2 \oplus \cdots \oplus \mathcal{S}_k$.

We denote by $\mathbf{C}^{m\times n}$ the set of complex matrices having m rows and n columns. Given two matrices $M \in \mathbf{C}^{m\times m}$ and $N \in \mathbf{C}^{n\times n}$, their *direct sum* is the block diagonal matrix $M \oplus N \in \mathbf{C}^{(m+n)\times(m+n)}$ having M and N along its main diagonal and 0 elsewhere.

Let us introduce some properties of normal matrices that will turn out to be useful in what follows. We denote by $M^\dagger \in \mathbf{C}^{m\times m}$ the *conjugate transpose* of the matrix M. If $MM^\dagger = M^\dagger M$ then M is said to be *normal*. Two important subclasses of normal matrices are the unitary and the Hermitian matrices. A matrix M is said to be *unitary* whenever $MM^\dagger = I = M^\dagger M$, where I is the identity matrix. The eigenvalues of unitary matrices are complex numbers of modulus 1, *i.e.*, they are in the form $\mathrm{e}^{i\vartheta}$, for some real ϑ. This fact characterizes the class of unitary matrices if we restrict to normal matrices. Alternative characterizations of normal and unitary matrices are contained, respectively, in

Proposition 1.1. ([15], Th. 4.10.3) *A matrix $M \in \mathbf{C}^{m\times m}$ is normal if and only if there exists a unitary matrix $X \in \mathbf{C}^{m\times m}$ such that $M = XDX^\dagger$, where $D = \mathrm{diag}(\nu_1, \nu_2, \dots, \nu_m)$ is the diagonal matrix of the eigenvalues of M.*

Proposition 1.2. ([15], Ths. 4.7.24, 4.7.14) *A matrix $M \in \mathbf{C}^{m\times m}$ is unitary if and only if:*

(i) its rows are mutually orthonormal vectors;
(ii) $\| xM \| = \| x \|$, for each vector $x \in \mathbf{C}^{1\times m}$.

A matrix M is said to be *Hermitian* whenever $M = M^\dagger$. All the eigenvalues of an Hermitian matrix are real. An Hermitian matrix is *positive semidefinite* if and only if all its eigenvalues are non negative. Alternative characterizations are contained in

Proposition 1.3. ([14], Ths. 4.12, 4.8) *An Hermitian matrix $M \in \mathbf{C}^{m\times m}$ is positive semidefinite if and only if:*

(i) $xMx^\dagger \geq 0$, for each vector $x \in \mathbf{C}^{1\times m}$;
(ii) $M = YY^\dagger$, for some matrix $Y \in \mathbf{C}^{m\times m}$.

Let $\omega = \mathrm{e}^{i\frac{2\pi}{n}}$ be the n-th root of the unity ($\omega^n = 1$), and define the matrix $W \in \mathbf{C}^{n\times n}$ whose (r, c)-th component is ω^{rc}, for $0 \leq r, c < n$. The *discrete Fourier transform* of a vector $x \in \mathbf{C}^{1\times n}$ is the vector $Wx^T \in \mathbf{C}^{n\times 1}$, where we denote with $x^T \in \mathbf{C}^{n\times 1}$ the transpose of vector x. The *inverse discrete Fourier transform* of x is the vector $(1/n)W^\dagger x$. Notice that $(1/n)W^\dagger W = I = W(1/n)W^\dagger$.

Let $f : \mathbf{N} \to \mathbf{C}$ be a periodic function of period n, *i.e.*, for any $k \in \mathbf{N}$, $f(k) = f(k + n)$ holds true. We say that f is n-periodic, for short, and it can be represented by the vector $(f(0), f(1), \dots, f(n-1))$. It is well know that f can be expressed as a linear combination of trigonometric functions by using the discrete Fourier transform and its inverse. More precisely:

$$f(k) = \frac{1}{n} \sum_{j=0}^{n-1} F(j)\, \omega^{-kj}, \tag{1}$$

where $(F(0), F(1), \ldots, F(n-1))^T = W(f(0), f(1), \ldots, f(n-1))^T$. By defining the support set $\mathrm{Supp}(F) = \{j \in \mathbf{Z}_n \mid F(j) \neq 0\}$, we can also write equation (1) as $f(k) = 1/n \sum_{j \in \mathrm{Supp}(F)} F(j)\, \omega^{-kj}$.

The reader is referred to, *e.g.* [1] (Chap. 7) for more details on the discrete Fourier transform and its relevance from a computational point of view. Here, we just recall that computing the discrete Fourier transform of n-dimensional vectors requires $\mathcal{O}(n \log n)$ sequential time.

1.2. Difference cover

The set $\Delta \subseteq \mathbf{Z}_n$ is a *difference cover* for $\mathbf{Z}_n$ if, for each $\kappa \in \mathbf{Z}_n$, there exist two elements $x, \tilde{x} \in \Delta$ such that $\kappa \equiv x - \tilde{x} \,(\mathrm{mod}\, n)$. The problem of covering $\mathbf{Z}_n$ by differences is well studied in the literature. Its relevance is also due to connections with some mutual exclusion issues in distributed systems, especially concerning *quorums* [6].

In [23], Wichmann proposes the following sequence of integers, for any $r \geq 0$ (x^r here means $xx \cdots x$ repeated r times):

$$\sigma = 1^r\, (r+1)^1\, (2r+1)^r\, (4r+3)^{2r+1}\, (2r+2)^{r+1}\, 1^r.$$

From σ, construct the set D of $6r+4$ integers by setting $a_1 = 0$, and $a_{i+1} = a_i + b_i$ for $1 \leq i \leq 6r+3$, where b_i is the i-th element of σ. It is easy to see that $a_{6r+4} = 12r^2 + 18r + 6$. The set D has the remarkable property that, for any $1 \leq d \leq 12r^2 + 18r + 6$, there exist $a, b \in D$ such that $d = a - b$.

In [6], Colbourn uses this fact to show that, *for any* $n \leq 24r^2 + 36r + 13$, D *is a difference cover for* $\mathbf{Z}_n$. This is basically due to the fact that, given $d \in \mathbf{Z}_n$, d or $-d$ can be represented by a positive integer less than or equal to $12r^2 + 18r + 6$. Hence, to find a difference cover for any $\mathbf{Z}_n$, it is enough to choose the smallest r satisfying $24r^2 + 36r + 13 \geq n$, and then to construct the corresponding set D with $6r+4$ elements. Simple arithmetics shows that $6r + 4 \leq \sqrt{1.5n} + 6$, and hence:

Theorem 1.4. ([6], Th. 2.4) *For any* $n \geq 0$, *there exists a difference cover for* $\mathbf{Z}_n$ *of cardinality at most* $\sqrt{1.5n} + 6$.

1.3. Quantum finite automata

Here, we are interested only in *measure-once* quantum finite automata [4, 5, 19]. Roughly speaking, in this kind of automata, the probability of acceptance is evaluated only at the end of the computation. In the literature, *measure-many* automata are also considered [2, 4, 5, 13], where such an evaluation is taken after each move. Hereafter, the attribute measure-once will always be understood.

The "hardware" of a *one-way quantum finite automaton* is that of a classical finite automaton. Thus, we have an input tape which is scanned by an input head moving one position right at each move[3], plus a finite state control. Formally:

[3]This kind of automata are sometimes referred to as *real time* automata [9, 19], stressing the fact that they can never present stationary moves.

Definition 1.5. A one-way quantum finite automaton (1qfa, for short) is a quintuple $\mathcal{A} = (Q, \Sigma, \pi(0), \delta, F)$, where

- $Q = \{s_1, s_2, \dots, s_q\}$ is the finite set of states;
- Σ is the finite input alphabet;
- $\pi(0) \in \mathbf{C}^{1\times n}$, with $\| \pi(0) \|^2 = 1$, is the vector of the initial amplitudes of the states;
- $F \subseteq Q$ is the set of accepting states;
- $\delta : Q \times \Sigma \times Q \to \mathbf{C}$ is the transition function mapping into the set of complex numbers having square modulus not exceeding 1; $\delta(s_i, \sigma, s_j)$ is the amplitude of reaching the state s_j from the state s_i, upon reading σ. The transition function must satisfy the following condition of well-formedness: for any $\sigma \in \Sigma$ and $1 \leq i, j \leq q$,

$$\sum_{k=1}^{q} \delta(s_i, \sigma, s_k)\, \delta^*(s_j, \sigma, s_k) = \begin{cases} 1 \text{ if } i = j \\ 0 \text{ otherwise.} \end{cases}$$

It is often useful to express the transition function on every $\sigma \in \Sigma$ as the *transition matrix* $U(\sigma) \in \mathbf{C}^{q\times q}$ whose (i, j)-th entry is the amplitude $\delta(s_i, \sigma, s_j)$. Since δ satisfies the condition of well-formedness above displayed, the rows of each $U(\sigma)$ are easily seen to be mutually orthonormal vectors and hence, by Proposition 1.2(i), *$U(\sigma)$'s are unitary matrices*. The 1qfa $\mathcal{A}$ can thus be represented as a triple $\mathcal{A} = (\pi(0), \{U(\sigma)\}_{\sigma\in\Sigma}, \eta_F)$, where $\eta_F \in \{0, 1\}^{n\times 1}$ is the characteristic vector of the accepting states.

Let us briefly discuss how our 1qfa $\mathcal{A}$ works. At any given time t, the *state* of $\mathcal{A}$ is a *superposition* of the states in Q and is represented by a vector $\pi(t)$ of norm 1 in the Hilbert space $l^2(Q)$. The i-th component of $\pi(t)$ is the amplitude of the state s_i. The computation on input $x = x_1 x_2 \dots x_n \in \Sigma^*$ starts in the superposition $\pi(0)$. After k steps, *i.e.*, after reading the first k input symbols, the state of $\mathcal{A}$ is the superposition

$$\pi(k) = \pi(0)U(x_1)U(x_2)\cdots U(x_k).$$

Since $\| \pi(0) \| = 1$ and $U(x_i)$'s are unitary matrices, Proposition 1.2(ii) ensures that $\| \pi(k) \| = 1$. When $\mathcal{A}$ enters the final superposition $\pi(n) = \pi(0) \prod_{i=1}^{n} U(x_i)$, we *observe* $\mathcal{A}$ by the *standard observable* $\mathcal{O} = \{l^2(F), l^2(Q \setminus F)\}$. $\mathcal{O}$ is the decomposition of $l^2(Q)$ into the two subspaces spanned by the accepting and nonaccepting states, respectively. The probability of accepting x is given by the square norm of the projection of $\pi(n)$ onto $l^2(F)$. Formally:

$$p_{acc}(x) = \sum_{\{j \mid (\eta_F)_j = 1\}} \left| \left(\pi(0) \prod_{i=1}^{n} U(x_i) \right)_j \right|^2,$$

where the subscript j denotes the j-th vector component.

A *stochastic event* is a function $p : \Sigma^* \to [0,1]$. The stochastic event *induced or defined by the 1qfa* $\mathcal{A}$ is the function $p_{\mathcal{A}} : \Sigma^* \to [0,1]$ defined, for any $x \in \Sigma^*$, as $p_{\mathcal{A}}(x) = p_{acc}(x)$. The *language accepted by* $\mathcal{A}$ *with cut point* $\lambda \geq 1/2$ is the set

$$L_{\mathcal{A},\lambda} = \{x \in \Sigma^* \mid p_{\mathcal{A}}(x) > \lambda\}.$$

A language L is said to be accepted by $\mathcal{A}$ *with isolated cut point* λ, if there exists $\varepsilon > 0$ such that, for any $x \in L$ ($x \notin L$), we have $p_{\mathcal{A}}(x) \geq \lambda + \varepsilon$ ($\leq \lambda - \varepsilon$).

A 1qfa $\mathcal{A}$ is *unary* if $|\Sigma| = 1$. In this case, we let $\Sigma = \{1\}$, and we can write $\mathcal{A} = (\pi(0), U, \eta_F)$ since we have a unique transition matrix U. With a slight abuse of notation, we will be writing k for the input string 1^k. The probability of accepting k now writes as

$$p_{acc}(k) = \sum_{\{j \mid (\eta_F)_j = 1\}} |(\pi(0)U^k)_j|^2. \tag{2}$$

The stochastic event *induced or defined by the unary automaton* $\mathcal{A}$ is the function $p_{\mathcal{A}} : \mathbf{N} \to [0,1]$, with $p_{\mathcal{A}}(k) = p_{acc}(k)$. A stochastic event $p : \mathbf{N} \to [0,1]$ is said to be *n-periodic* if it is an n-periodic function.

A unary language is a set $L \subseteq 1^*$. L is *n-periodic* if there exists a set $S \subseteq \mathbf{Z}_n$ such that $L = \{k \in \mathbf{N} \mid (k \bmod n) \in S\}$.

2. Synthesis of 1qfa's from periodic events

The first problem we shall be dealing with is the synthesis of 1qfa's inducing given periodic stochastic events. As a matter of fact, we will consider a relaxed version of this problem where, given a periodic event p, we aim to obtain a 1qfa inducing $ap + b$, for some reals $a > 0,\ b \geq 0$ satisfying $a + b \leq 1$.

If p is taken to be n-periodic, then it can be specified as input for the problem by providing the vector $(p(0), p(1), \ldots, p(n-1))$. Thus, formally we state:

Synthesis from events (SynE)

- ⋆ Input: An n-periodic stochastic event $(p(0), p(1), \ldots, p(n-1))$.
- ⋆ Output: A 1qfa $\mathcal{A}$ inducing the event $ap+b$, for some reals $a > 0$, $b \geq 0$, with $a + b \leq 1$.

We begin by preparing some tools to approach the problem. First of all, we point out some closure properties on the stochastic events induced by 1qfa's. We prove such properties for unary 1qfa's, but their extension to 1qfa's working on general input alphabets is straightforward.

Proposition 2.1. *Let $\mathcal{A} = (\pi_{\mathcal{A}}, U_{\mathcal{A}}, \eta_{\mathcal{A}})$ and $\mathcal{B} = (\pi_{\mathcal{B}}, U_{\mathcal{B}}, \eta_{\mathcal{B}})$ be two 1qfa's.*

(i) There exists a 1qfa $\overline{\mathcal{A}}$ with the same number of states as $\mathcal{A}$ such that $p_{\overline{\mathcal{A}}} = 1 - p_{\mathcal{A}}$.

(ii) For any nonnegative reals α, β satisfying $\alpha + \beta = 1$, there exists a 1qfa $\alpha\mathcal{A} + \beta\mathcal{B}$ such that $p_{\alpha\mathcal{A}+\beta\mathcal{B}} = \alpha p_{\mathcal{A}} + \beta p_{\mathcal{B}}$, and whose number of states is the sum of the number of states of $\mathcal{A}$ plus the number of states of $\mathcal{B}$.

Proof.

(i) Define $\overline{\mathcal{A}} = (\pi_{\mathcal{A}}, U_{\mathcal{A}}, \neg\eta_{\mathcal{A}})$, where $\neg\eta$ is the bitwise negation of η.
(ii) Define $\alpha\mathcal{A} + \beta\mathcal{B} = ((\sqrt{\alpha}\pi_{\mathcal{A}}, \sqrt{\beta}\pi_{\mathcal{B}}), U_{\mathcal{A}} \oplus U_{\mathcal{B}}, (\eta_{\mathcal{A}}, \eta_{\mathcal{B}}))$.

In both cases, it is easy to verify that we construct a well-defined 1qfa inducing the desired event. □

At this point, a quick comment on the relevance of SynE is in order. *From a language recognition point of view, the events p and $ap + b$ are equivalent* in the following sense: suppose we have a unary 1qfa $\mathcal{A}$ accepting the language $L_{\mathcal{A},\lambda}$ and suppose we are able to construct a unary 1qfa $\mathcal{A}_1$ inducing the event $ap_{\mathcal{A}} + b$. By setting $\lambda_1 = a\lambda + b$, it is easy to see that $L_{\mathcal{A}_1,\lambda_1} = L_{\mathcal{A},\lambda}$.

Here, a technical detail should be considered. As stated in Section 1.3, we must require that $\lambda_1 \geq 1/2$. If the opposite is true, by Proposition 2.1(ii), we construct the 1qfa $\mathcal{A}_2 = \frac{1}{2}\mathcal{A}_1 + \frac{1}{2}\mathcal{U}$, where $\mathcal{U}$ is a single state 1qfa realizing the event $p_{\mathcal{U}}(k) = 1$, for any $k \in \mathbf{N}$. We have $p_{\mathcal{A}_2} = 1/2(ap_{\mathcal{A}} + b) + 1/2$ and, by setting $\lambda_2 = (1/2)\lambda_1 + 1/2$, one easily gets $\lambda_2 \geq 1/2$ and $L_{\mathcal{A}_2,\lambda_2} = L_{\mathcal{A},\lambda}$.

In other words, solving SynE enables us to obtain unary 1qfa's accepting unary languages defined by a precise stochastic event.

Let us now show that the stochastic events induced by unary 1qfa's have a sort of *normal form*. In what follows, we denote by M_{ij} the (i,j)-th entry of the matrix M and by x_i the i-th component of the vector x.

Proposition 2.2. *Let p be a stochastic event induced by a unary 1qfa $\mathcal{A} = (\pi, U, \eta)$ with q states. Then, for any $k \in \mathbf{N}$,*

$$p(k) = \sum_{1 \leq s,t \leq q} \mathrm{e}^{ik(\vartheta_s - \vartheta_t)} B_{st},$$

where B is an Hermitian positive semidefinite matrix.

Proof. According to equation (2) in Section 1.3, the stochastic event induced by $\mathcal{A}$ writes as $p(k) = \sum_{\{j \mid \eta_j = 1\}} |(\pi U^k)_j|^2$, for any $k \in \mathbf{N}$. Since $U \in \mathbf{C}^{q \times q}$ is a unitary matrix, by Proposition 1.1, we can write $U = X \operatorname{diag}(\mathrm{e}^{i\vartheta_1}, \mathrm{e}^{i\vartheta_2}, \ldots, \mathrm{e}^{i\vartheta_q})\, X^\dagger$, where X is a unitary matrix and $\mathrm{e}^{i\vartheta}$'s are the norm 1 eigenvalues of U. Thus,

$$U^k = X \operatorname{diag}\left(\mathrm{e}^{ik\vartheta_1}, \mathrm{e}^{ik\vartheta_2}, \ldots, \mathrm{e}^{ik\vartheta_q}\right) X^\dagger,$$

and hence

$$p(k) = \sum_{\{j \mid \eta_j = 1\}} \left| \left(\pi X \operatorname{diag}\left(\mathrm{e}^{ik\vartheta_1}, \mathrm{e}^{ik\vartheta_2}, \ldots, \mathrm{e}^{ik\vartheta_q}\right) X^\dagger\right)_j \right|^2. \tag{3}$$

By letting $\xi = \pi X$ and substituting in (3), we get

$$\begin{aligned} p(k) &= \sum_{\{j \mid \eta_j = 1\}} \left(\left(\xi_1 \mathrm{e}^{ik\vartheta_1}, \ldots, \xi_q \mathrm{e}^{ik\vartheta_q}\right) X^\dagger\right)_j \left(\left(\xi_1 \mathrm{e}^{ik\vartheta_1}, \ldots, \xi_q \mathrm{e}^{ik\vartheta_q}\right) X^\dagger\right)_j^* \\ &= \sum_{\{j \mid \eta_j = 1\}} \left(\sum_{s=1}^{q} \xi_s \mathrm{e}^{ik\vartheta_s} X_{sj}^\dagger\right) \left(\sum_{t=1}^{q} \xi_t^* \mathrm{e}^{-ik\vartheta_t} (X_{tj}^\dagger)^*\right) \\ &= \sum_{1 \le s,t \le q} \mathrm{e}^{ik(\vartheta_s - \vartheta_t)} \sum_{\{j \mid \eta_j = 1\}} \xi_s X_{sj}^\dagger \left(\xi_t X_{tj}^\dagger\right)^*. \end{aligned}$$

Now, define the matrix B as

$$B_{st} = \sum_{\{j \mid \eta_j = 1\}} \xi_s X_{sj}^\dagger \left(\xi_t X_{tj}^\dagger\right)^*,$$

for $1 \le s, t \le q$. It is easy to verify that $B = B^\dagger$, and hence B is Hermitian. To prove that B is positive semidefinite, by Proposition 1.3(i), it is enough to show that $xBx^\dagger \ge 0$, for any $x \in \mathbf{C}^{1\times q}$:

$$\begin{aligned} xBx^\dagger &= \sum_{1 \le s,t \le q} x_s \left(\sum_{\{j \mid \eta_j = 1\}} \xi_s X_{sj}^\dagger (\xi_t X_{tj}^\dagger)^*\right) x_t^* \\ &= \sum_{\{j \mid \eta_j = 1\}} \left(\sum_{s=1}^{q} x_s \xi_s X_{sj}^\dagger\right) \left(\sum_{t=1}^{q} x_t \xi_t X_{tj}^\dagger\right)^* \\ &= \sum_{\{j \mid \eta_j = 1\}} \left|\sum_{s=1}^{q} x_s \xi_s X_{sj}^\dagger\right|^2 \ge 0. \end{aligned}$$

□

We are now ready to concentrate on SynE. Recall that our aim is to build a unary 1qfa $\mathcal{A}$ which induces $ap + b$, for some reals $a > 0$, $b \ge 0$, $a + b \le 1$, and an n-periodic stochastic event $p : \mathbf{N} \to [0, 1]$ given as input.

We start by observing that the event p is an n-periodic function and hence, according to equation (1) in Section 1.1, it expands as

$$p(k) = \frac{1}{n} \sum_{j=0}^{n-1} P(j)\, \omega^{-kj}, \tag{4}$$

for $(P(0), P(1), \ldots, P(n-1))^T = W(p(0), p(1), \ldots, p(n-1))^T$ being the discrete Fourier transform of p. On the other hand, in the light of Proposition 2.2, to be induced by a unary 1qfa, the event p must have the form

$$p(k) = \sum_{1 \le s,t \le q} \mathrm{e}^{ik(\vartheta_s - \vartheta_t)} B_{st}, \tag{5}$$

for some real ϑ's and an Hermitian positive semidefinite matrix B. These observations lead us to design an algorithm consisting of TWO PARTS. In the FIRST PART, we compute ϑ's and B so that equation (5) exactly reproduces equation (4). In the SECOND PART, we construct from such ϑ's and B a well formed 1qfa inducing the event $ap+b$.

FIRST PART OF THE ALGORITHM

★ INPUT: $(p(0), p(1), \dots, p(n-1))$

STEP 1: Compute $(P(0), P(1), \dots, P(n-1))^T = W(p(0), p(1), \dots, p(n-1))^T$, the discrete Fourier transform, and let $\mathrm{Supp}(P) = \{j \in \mathbf{Z}_n \mid P(j) \neq 0\}$.

STEP 2: Find a difference cover $\Delta = \{a_1, a_2, \dots, a_q\}$ for $\mathbf{Z}_n$.

STEP 3: For each $1 \leq t \leq q$, let $\vartheta_t = -\frac{2\pi}{n} a_t$.

STEP 4: For each $j \in \mathrm{Supp}(P)$, let

$$N(j) = |\{(a_s, a_t) \mid a_s, a_t \in \Delta \text{ and } j \equiv a_s - a_t \,(\mathrm{mod}\, n)\}|,$$

and, for $1 \leq s, t \leq q$, compute

$$B_{st} = \begin{cases} \dfrac{1}{n}\dfrac{P(j)}{N(j)} & \text{if } j \in \mathrm{Supp}(P) \text{ and } j \equiv a_s - a_t \,(\mathrm{mod}\, n) \\ 0 & \text{otherwise.} \end{cases}$$

It is easy to verify that $B \in \mathbf{C}^{q\times q}$ *is an Hermitian matrix:* to see that $B_{st} = {B_{ts}}^*$, it is enough to notice that $P(j) = P^*(-j \bmod n)$ and $N(j) = N(-j \bmod n)$, for each $j \in \mathbf{Z}_n$. By plugging ϑ's obtained at STEP 3 and B obtained at STEP 4 into equation (5), we get exactly $p(k)$ as in equation (4).

Now comes the SECOND PART of the algorithm which yields a 1qfa $\mathcal{A} = (\pi, U, \eta)$ inducing $ap+b$ from ϑ's and B computed in the FIRST PART. There are two ways of reconstructing $\mathcal{A}$, depending on whether B is positive semidefinite or not.

SECOND PART OF THE ALGORITHM

STEP 5.A: IF B *is positive semidefinite* THEN

- Find a matrix $Y \in \mathbf{C}^{q\times q}$ satisfying $YY^\dagger = B$. Such Y exists by Proposition 1.3(ii).
- Construct the $2q \times 2q$ matrix $M = \left(\begin{array}{c|c} Y & E \\ \hline F & \end{array}\right)$, where the row vectors $M_i \in \mathbf{C}^{1\times 2q}$ are mutually orthogonal. To get this, the first q rows of M can be computed by setting a lower triangular matrix $E \in \mathbf{C}^{q\times q}$ as

$$E_{ij} = \begin{cases} 1 & \text{if } i = j \\ -\langle\, Y_i, Y_j \,\rangle - \displaystyle\sum_{k=1}^{j-1} E_{ik}{E_{jk}}^* & \text{if } i > j \\ 0 & \text{otherwise,} \end{cases}$$

where Y_i is the i-th row of Y. At this point, we can take the q rows of F as an orthogonal basis of the subspace which is orthogonal to that

spanned by the vectors $M_1, M_2, \ldots, M_q$. Such a task can be performed by using standard tools in linear algebra (see, *e.g.* [14]).

- Define $X_i^\dagger = M_i / \parallel M_i \parallel$ to be the i-th row of the $2q \times 2q$ unitary matrix $X^\dagger$. Unitarity of $X^\dagger$ comes from the fact that we are constructing its rows as mutually orthonormal vectors, and hence Proposition 1.2(i) applies.
  Define also the vectors $\tilde{\xi} \in \mathbf{C}^{1\times 2q}$ and $\eta \in \mathbf{C}^{2q\times 1}$ as

$$\tilde{\xi}_i = \begin{cases} \parallel M_i \parallel & \text{for } i \leq q \\ 0 & \text{otherwise} \end{cases} \quad \text{and} \quad \eta_i = \begin{cases} 1 & \text{for } i \leq q \\ 0 & \text{otherwise.} \end{cases}$$

- Compute the $\mathbf{C}^{1\times 2q}$ vector $\pi = \xi X^\dagger$, where $\xi = \tilde{\xi} / \parallel \tilde{\xi} \parallel$.

★ OUTPUT: $\mathcal{A} = (\pi, XDX^\dagger, \eta)$, where $D = \mathrm{diag}(\mathrm{e}^{i\vartheta_1}, \ldots, \mathrm{e}^{i\vartheta_q}, \overbrace{1, \ldots, 1}^{q-\text{times}})$ (with ϑ's coming from STEP 3 in the FIRST PART of the algorithm).
Output also $a = \frac{1}{||\tilde{\xi}||^2}$.

Fact: It is not hard to see that $\mathcal{A}$ is a well formed 1qfa: first, notice that π is obtained by multiplying the norm 1 vector ξ by the unitary matrix $X^\dagger$. Hence, by Proposition 1.2(ii), $\parallel \pi \parallel = 1$. Next, notice that the transition matrix $XDX^\dagger$ is unitary, being the product of unitary matrices.

The 1qfa $\mathcal{A}$ has $2q$ states and it is easily seen to induce the event ap, with $0 < a = \frac{1}{||\tilde{\xi}||^2} \leq 1$.

STEP 5.B: IF *B is not positive semidefinite* THEN

- Find two Hermitian positive semidefinite matrices $G, H \in \mathbf{C}^{q\times q}$ such that $B = G - H$. These two matrices can be constructed as follows. Since B is an Hermitian matrix, by Proposition 1.1, we can write $B = X\mathrm{diag}(\nu_1, \nu_2, \ldots, \nu_q)X^\dagger$, where ν's are the real eigenvalues of B and X is a unitary matrix.
  Define $D^+ = \mathrm{diag}(\upsilon_1, \upsilon_2, \ldots, \upsilon_q)$, where $\upsilon_i = \nu_i$ if $\nu_i > 0$, and 0 otherwise. Set $D^- = D^+ - D$. Let $G = XD^+X^\dagger$ and $H = XD^-X^\dagger$. It is easy to see that $B = G - H$, and that both G and H are Hermitian. Moreover, one can easily verify that, for each $x \in \mathbf{C}^{1\times q}$, both $xGx^\dagger \geq 0$ and $xHx^\dagger \geq 0$ hold true. Hence, by Proposition 1.3(i), G and H are positive semidefinite.
- Perform STEP 5.A by having as input G and H. This yields two $2q$-state 1qfa's $\mathcal{A}_1$ and $\mathcal{A}_2$ inducing, respectively, the events a_1p_1 and a_2p_2, with $0 < a_1, a_2 \leq 1$ and $p_1 - p_2 = p$.
- Let $\mathcal{U}$ be the 1-state 1qfa inducing the event $p_{\mathcal{U}}(k) = 1$, for any $k \in \mathbf{N}$. Use Proposition 2.1 to construct the following 1qfa's:

  IF $a_1 \leq a_2$ THEN
  - construct $\mathcal{A}_3 = \frac{a_1}{a_2}\mathcal{A}_2 + (1 - \frac{a_1}{a_2})\mathcal{U}$. $\mathcal{A}_3$ has $2q+1$ states and induces the event $a_1p_2 + (1 - \frac{a_1}{a_2})$.
  - construct $\overline{\mathcal{A}_3}$, *i.e.*, the $(2q+1)$-state 1qfa that induces the event $1 - p_{\mathcal{A}_3} = \frac{a_1}{a_2} - a_1p_2$.

★ OUTPUT: $\mathcal{A}_4 = \frac{1}{2}\mathcal{A}_1 + \frac{1}{2}\overline{\mathcal{A}_3}$. Output also $a = \frac{a_1}{2}$ and $b = \frac{a_1}{2a_2}$.

IF $a_1 > a_2$ THEN

- construct $\mathcal{A}_3 = \frac{a_2}{a_1}\mathcal{A}_1 + (1 - \frac{a_2}{a_1})\mathcal{U}$. $\mathcal{A}_3$ has $2q+1$ states and induces the event $a_2 p_1 + (1 - \frac{a_2}{a_1})$;
- construct $\overline{\mathcal{A}_2}$, *i.e.*, the $2q$-state 1qfa inducing the event $1 - a_2 p_2$.

★ OUTPUT: $\mathcal{A}_4 = \frac{1}{2}\mathcal{A}_3 + \frac{1}{2}\overline{\mathcal{A}_2}$. Output also $a = \frac{a_2}{2}$ and $b = 1 - \frac{a_2}{2a_1}$.

Fact: It is easy to see that, in both cases, $\mathcal{A}_4$ is a $(4q+1)$-state 1qfa inducing the event $ap + b$, for $a, b > 0$, with $a + b \leq 1$.

In conclusion, the above algorithm provides a constructive proof of the following:

Theorem 2.3. *For any n-periodic event p, there exists a unary 1qfa with at most $2\sqrt{6n}+25$ states inducing the event $ap+b$, for some reals $a > 0$, $b \geq 0$, $a+b \leq 1$.*

Proof. We use our algorithm to construct a 1qfa $\mathcal{A}$ for $ap + b$. As one may easily verify, $\mathcal{A}$ turns out to have at most $4q + 1$ states, where q is the cardinality of a difference cover for $\mathbf{Z}_n$. By Theorem 1.4, q is bounded above by $\sqrt{1.5n}+6$, whence the result follows. □

We end with a quick evaluation of the complexity of our algorithm.

FIRST PART OF THE ALGORITHM: computing the discrete Fourier transform at STEP 1 requires $\mathcal{O}(n \log n)$ time, as observed in Section 1.1. The operations at STEPS 3 and 4 are easily seen to require polynomial time. Finally, at STEP 2, we can construct a difference cover for $\mathbf{Z}_n$ in polynomial time by using Wichmann's sequence, as addressed in Section 1.2.

SECOND PART OF THE ALGORITHM: the hardest tasks at STEPS 5.A and 5.B are basically to solve some problems from linear algebra, such as: $YY^\dagger$ factorization of Hermitian positive semidefinite matrices, computation of basis for orthogonal subspaces, decomposition of Hermitian matrices. For all these tasks, polynomial time algorithms can be obtained from the literature (see, *e.g.* [12]).

This enables us to conclude that a $(2\sqrt{6n} + 25)$-state 1qfa for the event $ap + b$ can be constructed in polynomial time.

3. Synthesis of 1qfa's from periodic languages

We now focus on accepting periodic languages, *i.e.*, unary languages in the form $L = \{k \in \mathbf{N} \mid (k \bmod n) \in S\}$, for a fixed $S \subseteq \mathbf{Z}_n$. As recalled in the introduction, recognizing n-periodic languages by 1dfa's takes at least n states. Moreover, in some cases, *e.g.* when n is a prime, even using 1pfa's (or also two-way nondeterminism) does not help in saving states.

By using the results in the previous section, we are always able to design 1qfa's with $\mathcal{O}(\sqrt{n})$ states and isolated cut point for n-periodic languages, as proved in the following:

Theorem 3.1. *Any n-periodic language can be accepted with isolated cut point on a 1qfa having no more than $2\sqrt{6n} + 26$ states.*

Proof. With each n-periodic language $L = \{k \in \mathbf{N} \mid (k \bmod n) \in S\}$, with $S \subseteq \mathbf{Z}_n$, we can associate the n-periodic event p defined, for each $k \geq 0$, as

$$p(k) = \begin{cases} 1 \text{ if } (k \bmod n) \in S \\ 0 \text{ otherwise.} \end{cases}$$

By Theorem 2.3, there exists a 1qfa $\mathcal{A}$, with no more than $2\sqrt{6n} + 25$ states, that induces $ap + b$, for some reals $a > 0$, $b \geq 0$, $a + b \leq 1$.

If $b + a/2 \geq 1/2$, we let $\lambda = b + a/2$ and $\varepsilon = a/2$. Otherwise, $\mathcal{U}$ being the 1-state 1qfa inducing the event $p(k) = 1$ for any $k \in \mathbf{N}$, we construct the automaton $\mathcal{A}_1 = \frac{1}{2}\mathcal{A} + \frac{1}{2}\mathcal{U}$ by adding one more state to the states of $\mathcal{A}$, and we let $\lambda = b/2 + a/4 + 1/2$ and $\varepsilon = a/4$.

It is easy to see that $\mathcal{A}$ or $\mathcal{A}_1$ accepts L with cut point $\lambda \geq 1/2$ isolated by ε. □

4. Some concluding remarks and open problems

In this work, we have provided a polynomial time algorithm for constructing small 1qfa's that induce periodic stochastic events or accept periodic languages. More precisely, we have shown that a linear approximation of any n-periodic event can be induced by a 1qfa with at most $2\sqrt{6n} + 25$ states, while any n-periodic language can be accepted with no more than $2\sqrt{6n} + 26$ states.

These results point out that, on a wide class of problems, 1qfa's are quadratically more succinct than classical automata[4]. In fact, it is well known that any 1dfa recognizing an n-periodic language must have at least n states. Yet, even by using two-way nondeterministic automata to accept p-periodic languages, for prime p's, we must employ at least p states [18].

More can be said even on the question quantum *vs.* probabilistic automata. As an immediate consequence of results in [17], we get that, for prime p's, any 1pfa accepting a p-periodic language with isolated cut point must have at least p states. This clearly implies the same state lower bound to induce p-periodic events by 1pfa's. Our results show that 1qfa's can be built that induce p-periodic events using only $\mathcal{O}(\sqrt{p})$ states. Moreover, we have used this fact to accept p-periodic languages with isolated cut point on $\mathcal{O}(\sqrt{p})$-state 1qfa's.

It should be noticed that, by using *ad hoc* techniques on *ad hoc* problems, we can sometimes obtain even more succinct 1qfa's. For instance, in [4], a $\mathcal{O}(\log p)$-state Monte-Carlo[5] 1qfa for $L_p = \{k \in \mathbf{N} \mid (k \bmod p = 0)\}$ is exhibited. However, due to its generality, we cannot expect our method to be so "state-saving". Nevertheless, it can be used as a tool to generate small quantum machines that can eventually serve as starting points for further refinements. Yet, we feel that our method could be of help in approaching open questions on quantum finite automata, some

[4] A similar quadratic decrease is proved in [11] for a particular binary language.

[5] A language L is accepted by a 1qfa $\mathcal{A}$ in Monte-Carlo mode if there exists $\varepsilon > 0$ such that $\mathcal{A}$ accepts with certainty every string in L, and rejects with probability $1 - \varepsilon$ every string not in L.

of which are quickly suggested hereafter:

- how to construct 1qfa's exactly inducing given periodic stochastic events?
- how to obtain Monte-Carlo 1qfa's accepting periodic languages?
- what about the size of 1qfa's for the previous two points?
- what about the size of minimal 1qfa's inducing periodic stochastic events or accepting periodic languages?

Acknowledgements. The authors wish to thank Alberto Bertoni for stimulating discussions.

References

[1] A. Aho, J. Hopcroft and J. Ullman, *The Design and Analysis of Computer Algorithms*. Addison-Wesley (1974).
[2] A. Ambainis, A. Ķikusts and M. Valdats, On the Class of Languages Recognizable by 1-way Quantum Finite Automata, in *Proc. 18th Annual Symposium on Theoretical Aspects of Computer Science*. Springer, *Lecture Notes in Comput. Sci.* **2010** (2001) 305-316.
[3] A. Ambainis and J. Watrous, *Two-way Finite Automata with Quantum and Classical States*. Technical Report (1999) `quant-ph/9911009`.
[4] A. Ambainis and R. Freivalds, 1-way Quantum Finite Automata: Strengths, Weaknesses and Generalizations, in *Proc. 39th Annual Symposium on Foundations of Computer Science*. IEEE Computer Society Press (1998) 332-342.
[5] A. Brodsky and N. Pippenger, *Characterizations of 1-Way Quantum Finite Automata*, Technical Report. Department of Computer Science, University of British Columbia, TR-99-03 (revised).
[6] C. Colbourn and A. Ling, Quorums from Difference Covers. *Inform. Process. Lett.* **75** (2000) 9-12.
[7] L. Grover, A Fast Quantum Mechanical Algorithm for Database Search, in *Proc. 28th ACM Symposium on Theory of Computing* (1996) 212-219.
[8] J. Gruska, *Quantum Computing*. McGraw-Hill (1999).
[9] J. Gruska, Descriptional complexity issues in quantum computing. *J. Autom. Lang. Comb.* **5** (2000) 191-218.
[10] T. Jiang, E. McDowell and B. Ravikumar, The Structure and Complexity of Minimal nfa's over a Unary Alphabet. *Int. J. Found. Comput. Sci.* **2** (1991) 163-182.
[11] A. Ķikusts, *A Small 1-way Quantum Finite Automaton*. Technical Report (1998) `quant-ph/9810065`.
[12] M. Kohn, *Practical Numerical Methods: Algorithms and Programs*. The Macmillan Company (1987).
[13] A. Kondacs and J. Watrous, On the Power of Quantum Finite State Automata, in *Proc. 38th Annual Symposium on Foundations of Computer Science*. IEEE Computer Society Press (1997) 66-75.
[14] M. Marcus and H. Minc, *Introduction to Linear Algebra*. The Macmillan Company (1965). Reprinted by Dover (1988).
[15] M. Marcus and H. Minc, *A Survey of Matrix Theory and Matrix Inequalities*. Prindle, Weber & Schmidt (1964). Reprinted by Dover (1992).
[16] C. Mereghetti and B. Palano, Upper Bounds on the Size of One-way Quantum Finite Automata, in *Proc. 7th Italian Conference on Theoretical Computer Science*. Springer, *Lecture Notes in Comput. Sci.* **2202** (2001) 123-135.

[17] C. Mereghetti, B. Palano and G. Pighizzini, On the Succinctness of Deterministic, Nondeterministic, Probabilistic and Quantum Finite Automata, in *Pre-Proc. Descriptional Complexity of Automata, Grammars and Related Structures*. Univ. Otto Von Guericke, Magdeburg, Germany (2001) 141-148. *RAIRO: Theoret. Informatics Appl.* (to appear).
[18] C. Mereghetti and G. Pighizzini, Two-Way Automata Simulations and Unary Languages. *J. Autom. Lang. Comb.* **5** (2000) 287-300.
[19] C. Moore and J. Crutchfield, Quantum automata and quantum grammars. *Theoret. Comput. Sci.* **237** (2000) 275-306.
[20] J.-E. Pin, On Languages Accepted by finite reversible automata, in *Proc. 14th International Colloquium on Automata, Languages and Programming*. Springer-Verlag, *Lecture Notes in Comput. Sci.* **267** (1987) 237-249.
[21] P. Shor, Algorithms for Quantum Computation: Discrete Logarithms and Factoring, in *Proc. 35th Annual Symposium on Foundations of Computer Science*. IEEE Computer Science Press (1994) 124-134.
[22] P. Shor, Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Comput.* **26** (1997) 1484-1509.
[23] B. Wichmann, A note on restricted difference bases. *J. London Math. Soc.* **38** (1963) 465-466.

Communicated by J. Gruska.
Received February 19, 2002. Accepted May 28, 2002.