Fortschr. Phys. **52**, No. 11–12, 1056 – 1063 (2004) / **DOI** 10.1002/prop.200410175

Probabilistic programmable quantum processors

V. Bužek[1,2,*], **M. Ziman**[1,2], and **M. Hillery**[3]

[1] Research Center for Quantum Information, Slovak Academy of Sciences, Dúbravská cesta 9, 845 11 Bratislava, Slovakia

[2] Faculty of Informatics, Masaryk University, Botanická 68a, 602 00 Brno, Czech Republic

[3] Department of Physics and Astronomy, Hunter College of CUNY, 695 Park Avenue, New York, NY 10021, USA

Received 31 July 2004, accepted 6 August 2004
Published online 14 October 2004

We analyze how to improve performance of probabilistic programmable quantum processors. We show how the probability of success of the probabilistic processor can be enhanced by using the processor in loops. In addition, we show that an arbitrary SU(2) transformations of qubits can be encoded in program state of a universal programmable probabilistic quantum processor. The probability of success of this processor can be enhanced by a systematic correction of errors via conditional loops. Finally, we show that all our results can be generalized also for qudits.

1 Introduction

The development of programmable quantum circuits is an area that has attracted attention only recently. The basic model for these circuits consists of two parts, a data register and a program register. There are two inputs, a data state, which is sent into the data register, and on which an operation is to be performed, and a program state, which is sent into the program register, that specifies the operation. The first result was due to Nielsen and Chuang, who showed that a deterministic *universal* quantum processor does not exist [1]. The problem is that a new dimension must be added to the program space for each unitary operator that one wants to be able to perform on the data. A similar situation holds if one studies quantum circuits that implement completely-positive, trace-preserving maps rather than just unitary operators [2, 3]. Some families of maps can be implemented with a finite program space, for example, the phase damping channel, but others, such as the amplitude damping channel, require and infinite program space. If one drops the requirement that the processor be deterministic, then universal processors become possible [1, 4–6]. These processors are probabilistic: they sometimes fail, but we know when this happens.

A number of examples of programmable quantum circuits have been proposed. One is a quantum "multimeter" that performs unambiguous state discrimination on a set of two states, the set being specified by the program [7]. There are also devices that evaluate the expectation value of an arbitrary operator, the data representing the state in which the expectation value is to be evaluated and the program state specifying the operator [8, 9].

In a probabilistic processor, one measures the output program state. If the proper result is obtained, the desired operation has been performed on the data state, and if not, then the output of the data register is discarded. In this kind of a scenario, one wants the probability of successfully performing the operation to be as close to one as possible. In fact, what one would like, is, given a set of operations that one

* Corresponding author E-mail: buzek@savba.sk

wishes to perform, a procedure for systematically increasing the probability of successfully performing these operations.

In the case of one-parameter unitary groups acting qubits this was done Preskill [4] and Vidal, Masanes and Cirac (VMC) [5]. Vidal, Masanes and Cirac considered the one-parameter group of operations given by $U(\alpha) = \exp(i\alpha\sigma_z)$, for $0 \leq \alpha < 2\pi$, and discussed two equivalent methods of making the probability of performing $U(\alpha)$ arbitrarily close to one. A circuit consisting of a single Controlled-NOT (CNOT) gate, with the control qubit as the data and the target qubit as the program, can successfully perform $U(\alpha)$ with a probability of 1/2. If the procedure fails, however, the data qubit, which was initially in the state $|\psi\rangle$, is left in the state $U(-\alpha)|\psi\rangle$. What we can now do, is to send this qubit back into the same circuit, but with the program state that encodes the operation $U(2\alpha)$. This also has a probability of 1/2 of succeeding, and increases the total success probability for the two-step procedure to 3/4. Note that our program state has increased to two qubits, one for the first step and one for the second. We can continue in this way simultaneously increasing the success probability and the size of the program state. It is also possible to design more complicated circuits that perform the entire procedure at once, i.e. they have a one-qubit data state, an N-qubit program state, and a success probability of $1 - (1/2)^N$ [5].

In this paper we show how to increase the success probability of programmable processors that are designed to realize various rotations on qubits and qudits.

2 Operations on qubits

We shall begin by describing the methods developed in [4] and [5] in terms of the formalism presented in [6]. There, the input data state is in the Hilbert space $\mathcal{H}_d$, the program state in the space $\mathcal{H}_p$, and G is the unitary operator, acting on the space $\mathcal{H}_d \otimes \mathcal{H}_p$, that describes the action of the circuit. This operator can be expressed as

$$G = \sum_{j,k=0}^{N} A_{jk} \otimes |j\rangle_p \, {}_p\langle k| \,, \tag{1}$$

where N is the dimension of $\mathcal{H}_p$, A_{jk} is an operator on $\mathcal{H}_d$, and $\{|j\rangle | j = 1, \ldots N\}$ is an orthonormal basis for the program space. The operators A_{jk} satisfy [6]

$$\sum_{j=1}^{N} A^{\dagger}_{jk_1} A_{jk_2} = \sum_{j=1}^{N} A_{k_1 j} A^{\dagger}_{k_2 j} = I_d \delta_{k_1 k_2} \,, \tag{2}$$

where I_d is the identity operator on $\mathcal{H}_d$. If the circuit acts on the input state $|\psi\rangle_d \otimes |\Xi\rangle_p$, we find that

$$G(|\psi\rangle_d \otimes |\Xi\rangle_p) = \sum_{j=1}^{N} A_j(\Xi)|\psi\rangle_d \otimes |j\rangle_p \,, \tag{3}$$

where

$$A_j(\Xi) = \sum_{k=1}^{N} {}_p\langle k|\Xi\rangle_p A_{jk} \,. \tag{4}$$

Let us begin by using this formalism, let us look at a CNOT gate and the simplest of the circuits discussed in [5]. Both the data and program space are two-dimensional, and the data space is the control qubit and the program space is the target qubit. Expressing the operator for the CNOT gate in the form given in eq. (1), and choosing the basis $\{|0\rangle, |1\rangle\}$ for the program space, we find that

$$\begin{aligned} &A_{00} = |0\rangle\langle 0|; \; A_{01} = |1\rangle\langle 1| \,; \\ &A_{10} = |1\rangle\langle 1|; \; A_{11} = |0\rangle\langle 0|. \end{aligned} \tag{5}$$

We want to use this circuit to perform the operation $U(\alpha)$ and this can be done with the program state

$$|\Xi(\alpha)\rangle = \frac{1}{\sqrt{2}}(\mathrm{e}^{i\alpha}|0\rangle + \mathrm{e}^{-i\alpha}|1\rangle)\,. \tag{6}$$

This gives us the output state

$$G(|\psi\rangle_d \otimes |\Xi(\alpha)\rangle_p) = \sum_{j=0}^{1} A_j(\alpha)|\psi\rangle_d \otimes |j\rangle_p \tag{7}$$

where the program operators are

$$\begin{aligned} A_0(\alpha) &= \frac{\mathrm{e}^{i\alpha}}{\sqrt{2}}|0\rangle\langle 0| + \frac{\mathrm{e}^{-i\alpha}}{\sqrt{2}}|1\rangle\langle 1| = \frac{1}{\sqrt{2}}U(\alpha)\,, \\ A_1(\alpha) &= \frac{\mathrm{e}^{i\alpha}}{\sqrt{2}}|1\rangle\langle 1| + \frac{\mathrm{e}^{-i\alpha}}{\sqrt{2}}|0\rangle\langle 0| = \frac{1}{\sqrt{2}}U(-\alpha)\,. \end{aligned} \tag{8}$$

Therefore, if we measure the output of the program register in the computational basis and obtain $|0\rangle$, then $U(\alpha)$ has been carried out on the data state. This occurs with a probability of 1/2.

If we obtain $|1\rangle$ instead of $|0\rangle$ when we measure the program register output, then the operation $U(-\alpha)$ has been performed on the data state. We can try to correct this by sending the state $U(-\alpha)|\psi\rangle_d$ back into the same circuit, but with the program state $|\Xi(2\alpha)\rangle_p$. If we measure the program output and obtain $|0\rangle$, then the output of the data register is

$$U(2\alpha)U(-\alpha)|\psi\rangle_d = U(\alpha)|\psi\rangle_d\,, \tag{9}$$

and this happens with a probability of 1/2. This will correct the previous error.

3 Realization of SU(2) rotations

In the Vidal-Masanes-Cirac model the angle of the U(1) rotation that is supposed to be performed on a qubit is encoded in a quantum state of the program. The rotation itself is then applied on the data qubit via the CNOT gate that plays the role of a programmable processor. As we have discussed above the probability of success of the rotation can be enhanced, providing the data qubit is processed conditionally in loops. The dynamics of each "run" of the processor is conditioned by the result of the measurement performed on the program register.

In what follows we will show that an analogous strategy can be applied in the case of the SU(2) rotations of a qubit, when the parameters (angles) of the SU(2) rotations are encoded in the state of the program. In our earlier work [6] we have shown an arbitrary single-qubit unitary transformation can be implemented with the probability $p = 1/4$ by using a quantum information distributor machine (QID) as the processor. The QID is a quantum processor with a single data qubit and two program qubits. The quantum information distribution is realized via a sequence of four CNOT gates, such that firstly the data qubit controls the NOT operation on the first and the second program qubits and then the first and the second program qubits act as the control with the data qubit as the target. At the end of this process a projective measurement on the two program qubits is performed. The measurement is performed in the basis: $\{|0\rangle|+\rangle; |0\rangle|-\rangle; |1\rangle|+\rangle; |1\rangle|-\rangle\}$ (where $|\pm\rangle = (|0\rangle \pm |1\rangle)/\sqrt{2}$). The realization of the desired transformation is associated with the projection onto the vector $|0\rangle|+\rangle$. In what follows we will explicitly show how to correct the cases of wrong results, i.e. of projections onto one of the vectors $|0\rangle|-\rangle, |1\rangle|+\rangle, |1\rangle|-\rangle$.

The action of the QID processor is given by relation [6, 10]

$$G = \sum_{j=0}^{3} \sigma_j \otimes |\Xi_j\rangle\langle\Xi_j|\,, \tag{10}$$

where σ_j are standard σ-matrices with $\sigma_0 = I$. The basis program vectors $|\Xi_j\rangle$ form the standard Bell basis, i.e.

$$|\Xi_0\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)\,; \qquad |\Xi_x\rangle = \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle)\,;$$
$$|\Xi_z\rangle = \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle)\,; \qquad |\Xi_y\rangle = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle)\,.$$

The general program state $|\Xi(\vec{\mu})\rangle_p$ encoding the unitary transformation $U_{\vec{\mu}} = \exp(i\vec{\mu}.\vec{\sigma}) = \cos\mu \boldsymbol{I} + i\sin\mu\,\frac{\vec{\mu}}{\mu}.\vec{\sigma}$ ($\mu = |\vec{\mu}|$) is given by the expression

$$|\Xi(\vec{\mu})\rangle_p = \cos\mu|\Xi_0\rangle + i\frac{\sin\mu}{\mu}(\mu_x|\Xi_x\rangle + \mu_y|\Xi_y\rangle + \mu_z|\Xi_z\rangle)\,. \tag{11}$$

Performing the previously mentioned measurement in the program basis $|0\rangle|+\rangle, |0\rangle|-\rangle, |1\rangle|+\rangle, |1\rangle|-\rangle$ we obtain the following unitary transformations

$$\begin{aligned}
|0\rangle|+\rangle &: |\psi\rangle_d \to U_{\vec{\mu}}|\psi\rangle_d\,;\\
|0\rangle|-\rangle &: |\psi\rangle_d \to \sigma_z U_{\vec{\mu}}\sigma_z|\psi\rangle_d\,;\\
|1\rangle|+\rangle &: |\psi\rangle_d \to \sigma_x U_{\vec{\mu}}\sigma_x|\psi\rangle_d\,;\\
|1\rangle|-\rangle &: |\psi\rangle_d \to \sigma_y U_{\vec{\mu}}\sigma_y|\psi\rangle_d\,,
\end{aligned}$$

where

$$U_{\vec{\mu}} = \cos\mu \boldsymbol{I} + \frac{i\sin\mu}{\mu}(\mu_x\sigma_x + \mu_y\sigma_y + \mu_z\sigma_z)\,. \tag{12}$$

To obtain this simple expression we have used the identity $\sigma_j\sigma_k\sigma_j = -\sigma_k$ if $k \neq j$. All observed outcomes occur with the same probability, $p = 1/4$. Using the above notation the action of the QID can be expressed in the form

$$|\psi\rangle_d \otimes |\Xi(\vec{\mu})\rangle_p \to \tfrac{1}{2}\left(\sum_{j=0}^{3}\sigma_j U_{\vec{\mu}}\sigma_j|\psi\rangle_d \otimes |\tilde{j}\rangle_p\right) \tag{13}$$

where vectors $\{|\tilde{j}\rangle_p\}$ form the basis of $\mathcal{H}_p$ associated with the realized measurement. The explicit form of the vectors is presented in following Section where we discuss a general solution of SU(N) rotations of qudits.

We see that each outcome of the measurement indicates a different unitary transformation has been applied to the data. Once we have obtained a specific result we can use the same processor again to correct an incorrectly transformed data register and consequently improve the success probability. In particular, in the case of the result j, the new program register needs to encode the correcting transformation $U_j^{(1)} = U_{\vec{\mu}}\sigma_j U_{\vec{\mu}}^{\dagger}\sigma_j$. The probability of implementing the unitary transformation using one conditioned loop is given as $p(1) = \frac{1}{4} + 3\frac{1}{16} = \frac{7}{16}$. Using more and more conditioned loops the success probability is given by $p(n) = \sum_{j=1}^{n}\frac{1}{4^j}3^{j-1} = \frac{1}{4}\sum_j(\frac{3}{4})^j = \frac{1}{4}\frac{1-(3/4)^n}{1/4} = 1-(3/4)^n$ converges to unity, i.e. $p(n) \to 1$ as the number of conditioned loops n goes to infinity. For instance, thirty conditioned loops result in the negligible probability of failure, $p \simeq 10^{-4}$.

The example of Vidal, Masanes and Cirac shows us that we are able to replace the feedback scenario with a probabilistic scenario by using different processors. An open problem is whether the same replacement can be done in general, or at least for the case of the QID.

4 SU (N) rotations of qudits

In what follow we will show that one can utilize the QID for a probabilistic implementation of SU(N) rotations of qudits. We start our discussion with a brief description of the QID in the case of qudits. First, we introduce a generalization of the two-qubit CNOT gate [10] for qudits. This is a conditional shift operator defined with a control qudit "a" and the target qudit "b"

$$D_{ab} = \sum_{k,m=0}^{N-1} |k\rangle_a\langle k| \otimes |(m+k) \bmod N\rangle_b\langle m| \,, \tag{14}$$

which implies that

$$D_{ab}^{\dagger} = \sum_{k,m=0}^{N-1} |k\rangle_a\langle k| \otimes |(m-k) \bmod N\rangle_b\langle m| \,. \tag{15}$$

From this definition it follows that the operator D_{ab} acts on the basis vectors of a qudit as

$$D_{ab}|k\rangle|m\rangle = |k\rangle|(k+m) \bmod N\rangle \,, \tag{16}$$

which means that this operator has the same action as the conditional adder and can be performed with the help of the simple quantum network discussed in [11]. Note that for $N > 2$ the two operators D and $D^{\dagger}$ differ; they describe conditional shifts in opposite directions. Therefore the generalizations of the CNOT operator to higher dimensions are just *conditional shifts*.

Following our earlier work [6, 10] we can assume the network for the probabilistic universal quantum processor to be

$$P_{123} = D_{31}D_{21}^{\dagger}D_{13}D_{12} \,. \tag{17}$$

The data register consists of system 1 and the program register of systems 2 and 3. The state $|\Xi_V\rangle_{23}$ acts as the "software" that caries the information about the operation V to be implemented on the qudit data state $|\Psi\rangle_1$. The output state of the three qudit system, after the four controlled shifts are applied, reads

$$|\Omega\rangle_{123} = D_{31}D_{21}^{\dagger}D_{13}D_{12}|\Psi\rangle_1|\Xi_V\rangle_{23} \,. \tag{18}$$

The sequence of four operators acting on the basis vectors gives $|n\rangle_1|m\rangle_2|k\rangle_3$ as

$$D_{31}D_{21}^{\dagger}D_{13}D_{12}|n\rangle_1|m\rangle_2|k\rangle_3 = |(n-m+k) \bmod N\rangle_1\,|(m+n) \bmod N\rangle_2\,|(k+n) \bmod N\rangle_3 \,. \tag{19}$$

We now turn to the fundamental program states. A basis consisting of maximally entangled two-particle states (the analogue of the Bell basis for spin-$\frac{1}{2}$ particles) is given by

$$|\Xi_{mn}\rangle = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \exp\Big(i\frac{2\pi}{N}mk\Big)|k\rangle|(k-n) \bmod N\rangle \,, \tag{20}$$

where $m, n = 0, \ldots, N-1$. If $|\Xi_{mn}\rangle_p$ is the initial state of the program register, and $|\Psi\rangle = \sum_j \alpha_j|j\rangle_d$ (here, as usual, $\sum_j |\alpha_j|^2 = 1$) is the initial state of the data register, then follows that

$$\begin{aligned} P_{123}|\Psi\rangle_1|\Xi_{mn}\rangle_{23} &= \sum_{jk} \frac{\alpha_j}{\sqrt{N}} \exp\left[\frac{2\pi ikm}{N}\right] P_{123}|j\rangle|k\rangle|k-n\rangle \\ &= \sum_{jk} \frac{\alpha_j}{\sqrt{N}} \exp\frac{2\pi ikm}{N}|j-n\rangle|k+j\rangle|k+j-n\rangle \end{aligned}$$

$$= \sum_{jk} \alpha_j \exp \frac{-2\pi i j m}{N} |j-n\rangle |\Xi_{mn}\rangle$$
$$= (U^{(m,n)}|\Psi\rangle)|\Xi_{mn}\rangle\,, \tag{21}$$

where we have introduced the notation

$$U^{(m,n)} = \sum_{s=0}^{N-1} \exp \frac{-2i\pi s m}{N} |s-n\rangle\langle s|\,. \tag{22}$$

This result is similar to the one we found in the case of a single qubit (see previous section). The operators $U^{(m,n)}$ satisfy the orthogonality relation

$$\mathrm{Tr}\left[(U^{(m',n')})^\dagger U^{(m,n)}\right] = N\delta_{m,m'}\delta_{n,n'}\,. \tag{23}$$

The space of linear operators $\mathcal{T}(\mathcal{H})$ defined on some Hilbert space $\mathcal{H}$ with the scalar product given by (23) we know as *Hilbert-Schmidt space*. Thus the unitary operators $U^{(m,n)}$ form an orthogonal basis in it and any operator $V \in \mathcal{T}(\mathcal{H})$ can be expressed in terms of them

$$V = \sum_{m,n=0}^{N-1} d_{mn} U^{(m,n)}\,. \tag{24}$$

The orthogonality relation allows us to find the expansion coefficients in terms of the operators

$$d_{mn} = \frac{1}{N}\,\mathrm{Tr}\left[\left(U^{(m,n)}\right)^\dagger V\right]\,. \tag{25}$$

Therefore, the program vector that implements the operator V is given by

$$|\Xi_V\rangle_{23} = \sum_{m,n=0}^{N-1} d_{mn}|\Xi_{mn}\rangle_{23}\,. \tag{26}$$

Application of the processor to the input state $|\Psi\rangle_1|\Xi_V\rangle_{23}$ yields the output state

$$|\Omega\rangle_{123} = \sum_{mn} d_{mn} U^{(m,n)}|\Psi\rangle_1 \otimes |\Xi_{mn}\rangle_{23}\,. \tag{27}$$

Now let us perform a measurement of the program output in the basis

$$|\Phi_{rs}\rangle = \frac{1}{N}\sum_{m,n=0}^{N-1} \exp\left[2\pi i \frac{(mr-ns)}{N}\right]|\Xi_{mn}\rangle\,. \tag{28}$$

The orthogonality of this measurement basis directly follows from the orthogonality of the entangled basis $|\Xi_{mn}\rangle$. We should also note, that the vectors $|\Phi_{rs}\rangle$ itself can be rewritten in a factorized form, i.e.

$$|\Phi_{rs}\rangle = |-r\rangle \otimes \frac{1}{\sqrt{N}}\sum_{n=0}^{N} \exp\left[2\pi i \frac{ns}{N}\right]|n-r\rangle\,, \tag{29}$$

which means that the measurement can be performed independently on two program qudits.

In order to clarify the role of the measurement we will rewrite the output state of the QID using the basis $|\Phi_{rs}\rangle$ for program qudits:

$$
\begin{aligned}
P_{123}|\Psi\rangle_1|\Xi_V\rangle_{23} &= \sum_{m,n=0}^{N-1} d_{m,n}U^{(m,n)}|\Psi\rangle_1|\Xi_{mn}\rangle_{23} \\
&= \sum_{m,n=0}^{N-1} d_{m,n}U^{(m,n)}|\Psi\rangle_1 \left[\frac{1}{N}\sum_{r,s=0}^{N-1}\exp\left[-2\pi i\frac{(mr-ns)}{N}\right]|\Phi_{rs}\rangle_{23}\right] \\
&= \frac{1}{N}\sum_{r,s=0}^{N-1}\sum_{m,n=0}^{N-1}\left\{\exp\left[-2\pi i\frac{(mr-ns)}{N}\right]d_{m,n}U^{(m,n)}\right\}|\Psi\rangle_1|\Phi_{rs}\rangle_{23}\,. \quad (30)
\end{aligned}
$$

Taking into account that

$$
\left[U^{(p,q)}\right]^{\dagger}U^{(m,n)}U^{(p,q)} = \exp\left[2\pi i\frac{(mq-np)}{N}\right]U^{(m,n)} \quad (31)
$$

and choosing $p=s$ and $q=r$ we find

$$
\frac{1}{N}\,\mathrm{Tr}\left[\left(U^{(s,r)}\right)^{\dagger}\left(U^{(m,n)}\right)^{\dagger}U^{(s,r)}V\right] = \exp\left[-2\pi i\frac{(mr-ns)}{N}\right]d_{m,n}\,. \quad (32)
$$

Finally, the output of the QID can be rewritten in the form

$$
P_{123}|\Psi\rangle_1|\Xi_V\rangle_{23} = \frac{1}{N}\sum_{r,s=0}^{N-1}\left[U^{(s,r)}V\left(U^{(s,r)}\right)^{\dagger}\right]|\Psi\rangle_1|\Phi_{rs}\rangle_{23}\,, \quad (33)
$$

from which it is clear that if the result of the measurement of the two program qudits is $|\Phi_{rs}\rangle_{23}$, then the system (data) is left in the state $\left[U^{(s,r)}V\left(U^{(s,r)}\right)^{\dagger}\right]|\Psi\rangle_1$. Obviously, if $s=r=0$, then the operator V is applied on the data qudit. The probability of this outcome is $1/N^2$. For all other results of the measurement the data qudit is left in the state given above. One can use these output states with a modified program state to improve the performance of the programmable processor. Specifically, we have to use the new program state $|\Xi_V^{(r,s)}\rangle$ that is chosen after taking into account the result of the previous measurement. This program state has first to "correct" the wrong realization of the operation V during the previous "run" of the processor and then apply (probabilistically), the original operation V. For this reason, the new program state has to perform the operation

$$
V^{(r,s)} = V\left[U^{(s,r)}V\left(U^{(s,r)}\right)^{\dagger}\right]^{-1}\,. \quad (34)
$$

This process of error correction (conditional loops) can be used K times and the technique of conditioned loops can be exploited in order to amplify the probability of success. Applying the processor K times the probability of a successful application of the desired SU(N) operation V reads $p(K) = 1-(1-1/N^2)^K$.

5 Conclusions

In this paper we have analyzed a probabilistic programmable quantum processor. We have shown how to encode information about the quantum dynamics V to be performed on a quantum system (data register) in the state of another quantum system (program register). This information is stored in such a way that the program can be used to probabilistically perform the stored transformation on the data. In our paper

we have analyzed systematically how to perform U(1) rotations of qubits and qudits. We have generalized the whole problem and we have shown that one can use a very simple quantum processor, the so called quantum information distributor, to perform arbitrary SU(2) rotations of qubits as well as SU(N) rotations of qudits using the probabilistic programmable processor with the quantum program register initially prepared in states that carry the information about the operation to be performed on the data. It is also possible to use enlarged programs to increase the probability of success without the use of loops. In this case the measurement performed on the program register has to be modified accordingly. We have shown that if the processor is used in loops with properly chosen program states one can improve the performance of the quantum programmable processor so that the probability of failure decreases exponentially with the number of program qudits that store the information about transformation on the data qudit.

Acknowledgements This work was supported in part by the European Union projects QGATES and CONQUEST, by the National Science Foundation under grant PHY-0139692, and by the Slovak Academy of Sciences.

References

[1] M. Nielsen and I. L. Chuang, Phys. Rev. Lett. **79**, 321 (1997).
[2] M. Hillery, V. Bužek, and M. Ziman, Fortschr. Phys. **49**, 987 (2001).
[3] M. Hillery, M. Ziman, and V. Bužek Phys. Rev. A **66**, 042302 (2002).
[4] J. Preskill, Proc. R. Soc. Lond. A **454**, 385 (1998).
[5] G. Vidal, L. Masanes, and J. I. Cirac, Phys. Rev. Lett. **88** 047905 (2002).
[6] M. Hillery, V. Bužek, and M. Ziman, Phys. Rev. A **65**, 022301 (2002).
[7] M. Dušek and V. Bužek, Phys. Rev. A **66**, 022112 (2002).
[8] A. K. Ekert, C. M. Alves, D. K. L. Oi, M. Horodecki, P. Horodecki, and L. C. Kwek, Phys. Rev. Lett. **88**, 217901 (2002).
[9] J. P. Paz and A. Roncaglia, Phys. Rev. A **68**, 052316 (2003).
[10] S. L. Braunstein, V. Bužek, and M. Hillery, Phys. Rev. A **63**, 052313 (2001).
[11] V. Vedral, A. Barenco, and A. Ekert, Phys. Rev. A **54**, 147 (1996).