Scaling issues in ensemble implementations of the Deutsch-Jozsa algorithm

Arvind* and David Collins†
Department of Physics, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213, USA
(Received 24 July 2003; published 3 November 2003)

We discuss the ensemble version of the Deutsch-Jozsa (DJ) algorithm which attempts to provide a ''scalable'' implementation on an expectation-value NMR quantum computer. We show that this ensemble implementation of the DJ algorithm is at best as efficient as the classical random algorithm. As soon as any attempt is made to classify all possible functions with certainty, the implementation requires an exponentially large number of molecules. The discrepancies arise out of the interpretation of mixed state density matrices.

DOI: 10.1103/PhysRevA.68.052301 PACS number(s): 03.67.Lx

Conventional NMR implementations of quantum computing algorithms require the preparation of pseudopure states [1–5]. There have been a few proposals to efficiently implement the Deutsch-Jozsa (DJ) algorithm on an NMR quantum information processor using highly mixed states [6,7]. The basic idea in these ensemble schemes is to avoid pseudopure state preparation which would require exponential resources and instead work with highly mixed states close to thermal equilibrium. These schemes need to be carefully examined for their ''quantum character'' and their efficiencies compared to classical random algorithms. In this paper we show that for the DJ problem, a parallel can be drawn between these ensemble implementations and classical random algorithms.

We begin with a brief recapitulation of the DJ problem. Consider the set of functions $f:\{0,1\}^n \rightarrow \{0,1\}$. If all 2^n inputs map to the same output then the function is ''constant'' and if half the outputs map to 0 while the other half to 1 then the function is ''balanced.'' Functions which are neither constant nor balanced are not considered here. The task here is to determine the constant or balanced nature of a given function. Given an oracle which evaluates $f(x)$ at an input $x \in \{0,1\}^n$, no deterministic classical algorithm can carry out such a classification with certainty without using at least $2^{n-1}+1$ invocations of the oracle. The quantum DJ algorithm on the other hand accomplishes the classification task by invoking the oracle only once [8,9]. This it does by using a quantum oracle defined through the unitary transformation on an n qubit argument $|x\rangle$ and a one qubit target $|y\rangle$,

$$|x\rangle|y\rangle \xrightarrow{\hat{U}_f} |x\rangle|y \oplus f(x)\rangle. \tag{1}$$

If a query to the oracle is assumed to come at a unit cost then the quantum DJ algorithm provides an exponential speedup over its classical counterpart. The $n+1$ qubits need to be in a pure quantum state for the algorithm to work. In liquid state NMR at room temperatures, which is the most successful implementation of quantum information processing to date, the quantum states of the spins are far from pure. Therefore, to emulate the standard version of the DJ algorithm, one has to prepare the system in a special 'pseudopure' state where the ensemble is divided into two parts: a small subset in a given pure state, and the rest acting as a uniform background with no contribution to the signal. However, the present preparation schemes for such states lead to an exponential loss of signal because the subset of spins which can be prepared in a ''pure'' state decreases exponentially with the number of qubits [5,10].

Recently, alternative schemes have been proposed to circumvent this difficulty [6,7]. These effectively use a computer with $n+1$ qubits, where the first n qubits are represented by a density matrix $(1/2)I$ (a fully mixed state) and the last qubit is in the pure state $|0\rangle$. The initial state thus is

$$\rho_{\text{in}} = \frac{1}{2^n} I \otimes I \otimes \cdots I \otimes |0\rangle\langle 0| \tag{2}$$

and rewriting it in the computational basis $\{|x\rangle|0\rangle | x=0 \cdots x=2^n-1\}$ gives

$$\rho_{\text{in}} = \frac{1}{2^n} \sum_{x=0}^{2^n-1} |x\rangle\langle x| \otimes |0\rangle\langle 0|. \tag{3}$$

This preparation is followed by the standard quantum oracle query described in Eq. (1), yielding

$$\rho_{\text{out}} = \frac{1}{2^n} \sum_{x=0}^{2^n-1} |x\rangle\langle x| \otimes |f(x)\rangle\langle f(x)|. \tag{4}$$

Before we extract the constant or balanced nature of the function we note that this is not an entangled state. The entanglement is missing because of the special choice of the initial state. As a matter of fact the oracle is capable of generating entanglement and the standard pure state version of the quantum algorithm relies on entanglement [8,9].

The information about the function is contained entirely in the target qubit whose reduced density matrix is

*Permanent address: Department of Physics, Guru Nanak Dev University, Amritsar 143005, India; Electronic address: xarvind@andrew.cmu.edu

†Present address: Department of Physics, Bucknell University, Lewisburg, PA 17837, USA. Electronic address: dcollins@bucknell.edu

$$\rho_{\text{target}} = \frac{1}{2^n} \sum_{x=0}^{2^n-1} |f(x)\rangle\langle f(x)|. \tag{5}$$

The expectation value of σ_z in this state will immediately reveal the constant or balanced nature of the function f, with the result

$$\langle \sigma_z \rangle = \begin{cases} \pm 1 & \text{constant} \quad f \\ 0 & \text{balanced} \quad f. \end{cases} \tag{6}$$

By actually carrying out such a measurement the function can be classified with a single invocation of the quantum oracle without the associated problems of preparing pure or pseudopure states. This scheme is particularly suitable for implementation on a NMR quantum information processor, where the thermal equilibrium state can be easily transformed into the maximally mixed state of Eq. (2) and expectation value measurements are natural.

For comparison we describe a classical scenario which in essence mimics the ''quantum'' scheme described above. Instead of the NMR qubits, consider classical bit strings of length $n+1$. Further, assume that we have 2^n such strings and each string is in a different state for the first n bits, thereby providing representation to all possible states of the first n bits. The $(n+1)$th bit in each string is set to 0 and acts as the target bit. Now the application of the classical oracle $[x \rightarrow f(x)$ with $f(x)$ appearing on the target] to all the copies will yield the function values at all the 2^n input points and this value will be stored in the target bit in each copy. The constant or balanced nature of the function can then be obtained by adding these values by appropriate gates. The values add to 2^n or 0 for a constant function and to $2^n/2$ for a balanced function. This is analogous to the case of the expectation value quantum algorithm using maximally mixed states. In our view this scheme is thus fully classical, using separable states at all stages and camouflaged in the language of quantum mechanics. The exponential resource is explicit in the classical situation while it is hidden in the definition of the input density matrix of Eq. (2) for the ensemble quantum case. The deceptively simple fact that one has effectively prepared the state of the first n qubits in the density matrix $(1/2^n)(I \otimes I \otimes I \ldots \otimes I)$ requires that we have at least 2^n molecules!

Every density operator can be viewed as an ensemble of pure states occurring with certain probabilities. The existence of a decomposition

$$\rho = \sum_i p_i |\psi_i\rangle\langle\psi_i| \quad \text{with} \quad p_i \geqslant 0, \quad \sum p_i = 1 \tag{7}$$

implies an ensemble $\{p_i, |\psi_i\rangle\}$ for ρ, where the state $|\psi_i\rangle$ occurs with probability p_i. It is to be noted that the states $|\psi_i\rangle$ need not be orthogonal and the decomposition given above is not unique for mixed states. However, whatever can be determined from ρ can be consistently thought of as deriving from any one of the ensembles. The ensemble scheme that culminates in Eq. (6) is illustrated by considering the situation in which each ensemble member is taken to be in one of the computational basis (pure) states, $\{|x\rangle|0\rangle|x = 0 \ldots 2^n - 1\}$, each of which occurs with probability $p_x = 1/2^n$. This corresponds to the input density operator of Eq. (3). Now view the scheme as it is applied on each member of the ensemble, which means that each member of the ensemble independently computes the function on the input state of that member and the output appears on the $(n+1)$th qubit of each member molecule. The measurement of the average of σ_z then ostensibly reveals the constant or balanced nature of the function as described in Eq. (6).

Now imagine that one chooses to work with a fixed number of molecules M. As the number of qubits n increases, soon one will reach a stage when $M < 2^n/2$. In this case more than half the computational basis states cannot find representation in the ensemble because there are simply not enough molecules! Therefore there are always balanced functions which will have the same value over these M states and will get classified as constant, despite assuming the best situation, where all the molecules are assumed to be in different states. We will see later that the actual scheme is even more inefficient because all the molecules cannot be in different computational basis states. Thus, for the scheme to work for all functions one needs at least $2^n/2$ molecules in the ensemble, a number which grows exponentially with n. In other words, when the number of molecules is smaller than 2^n, there is no way one can prepare the input density matrix of Eq. (2). We will return to this point later.

How many balanced functions escape classification for a given M? Assume $N = 2^n$ is the input set size. The number of constant functions is 2 which is independent of n while the number of balanced function is ${}^N C_{N/2}$. If the scheme is used with a number of molecules $1 \leqslant M \leqslant N/2$, the balanced functions which escape classification are the ones which have same value (0 or 1) for the first M inputs. The number of functions which have the value 1 (0) for the first M inputs is the same as the number of ways one can distribute the remaining $N/2 - M$ 1's (0's) on $N - M$ inputs, giving:

$$\begin{pmatrix} \text{Number of Balanced Functions} \\ \text{Classified as Constant} \end{pmatrix} = 2 \begin{pmatrix} N-M \\ N/2-M \end{pmatrix}. \tag{8}$$

Dividing this by the total number of balanced functions gives the fraction of balanced functions for which the schemes fails

$$(\text{Failure Fraction}) = 2 \begin{pmatrix} N-M \\ N/2-M \end{pmatrix} \Big/ \begin{pmatrix} N \\ N/2 \end{pmatrix}. \tag{9}$$

This fraction diminishes quite fast as M increases from 0 toward $N/2$, increasing the efficiency of the algorithm. The fact that for most cases the scheme will work with a relatively small number of molecules has nothing to do with quantum mechanics. The classical randomized algorithm too will work to the same extent. In fact the above counting is valid for the classical algorithm as well. It is well known that there is an efficient randomized classical algorithm for the DJ problem and we conclude that the expectation value ensemble scheme in the best case is equivalent to it [11].

Even if we use a molecule number $M>2^n$ how safely can we say that the initial maximally mixed state has been realized and we are able to classify all functions with certainty? It is possible that, in a given experimental implementation of the algorithm, all ensemble members are in the same initial state $|x'\rangle|0\rangle$. In this event, the algorithm only evaluates $f(x')$ and the measurement outcome will be that for a constant function regardless of the nature of f. Thus, in contrast to the conventional DJ algorithm, there can be no way of determining the function type with certainty. In this sense, this ensemble algorithm for solving the DJ problem is not deterministic and must be compared to probabilistic classical algorithms. We shall consider the probability with which each correctly determines the function type and show that, regardless of the ensemble size, the standard probabilistic classical algorithm is superior to this ensemble quantum algorithm.

Suppose that the ensemble consists of M identical, independent $n+1$ qubit molecules. Each member of the ensemble will be subject to the unitary of Eq. (1), which can be re-expressed as

$$\hat{U}_f=\hat{P}_0(f)\otimes I+\hat{P}_1(f)\otimes\sigma_x, \tag{10}$$

where

$$\hat{P}_0(f)\coloneqq\sum_{x:f(x)=0}|x\rangle\langle x|,$$

$$\hat{P}_1(f)\coloneqq\sum_{x:f(x)=1}|x\rangle\langle x|, \tag{11}$$

project onto subspaces of n qubit argument while I and σ_x act on the target qubit. For the target qubit, $\langle\sigma_z\rangle$ is approximated by

$$\overline{z}\coloneqq\frac{1}{M}\sum_{j=1}^{M}z_j, \tag{12}$$

where $z_j=\pm1$ are the outcomes of projective measurement ($z_j=1$ corresponding to $|0\rangle\langle0|$ and $z_j=-1$ to $|1\rangle\langle1|$) on the target qubit for individual ensemble members.

The only assumption that we make about the ensemble members' initial states is that they occur with probabilities described by the density operator of Eq. (3). Then, for any ensemble member j,

$$\mathrm{Prob}(z_j=+1|f)=\mathrm{Tr}_{\arg}(\hat{P}_0(f)\rho_{\mathrm{in}}), \tag{13a}$$

$$\mathrm{Prob}(z_j=-1|f)=\mathrm{Tr}_{\arg}(\hat{P}_1(f)\rho_{\mathrm{in}}), \tag{13b}$$

where measurements are performed immediately after algorithm unitaries and the traces are taken over the argument register only. Note that each constant function yields one measurement outcome with certainty: $z_j=+1$ for $f=0$ and $z_j=-1$ for $f=1$. Thus $\overline{z}=+1$ for $f=0$ and $\overline{z}=-1$ for $f=0$. Whenever $\overline{z}$ departs from ±1 it is clear that f is balanced. However, the extent to which such a departure is noticeable depends on the available measurement resolution, which can be expressed in terms of outcomes of function register measurements on individual ensemble members. Suppose that it is possible to distinguish two ensemble averages only when they differ in R (out of M) or more individual measurement outcomes. Then we regard two ensemble averages as distinct provided that $|\overline{z}-\overline{z'}|\geqslant R/M$. This motivates the following protocol for deciding the algorithm outcome:

$$\overline{z}\geqslant1-R/M\Rightarrow f=0, \tag{14a}$$

$$\overline{z}\leqslant-1+R/M\Rightarrow f=1, \tag{14b}$$

$$1-R/M>\overline{z}>-1+R/M\Rightarrow f \text{ balanced}. \tag{14c}$$

The issue is to determine the probability with which this protocol will correctly identify the function type. Constant functions will always be identified correctly and we need only to find the probability that a balanced function will give $\overline{z}\geqslant1-R/M$ or $\overline{z}\leqslant-1+R/M$. These are the probabilities that a balanced function will return $z_j=-1$ or $z_j=+1$ at most $R-1$ times, respectively. For a balanced function $\mathrm{Prob}(z_j=+1)=\mathrm{Prob}(z_j=-1)=1/2$. The probability that we incorrectly declare a balanced function to be constant is

$$p_{\mathrm{fail}}=2\sum_{k=0}^{R-1}\binom{M}{k}\left(\frac{1}{2}\right)^k\left(\frac{1}{2}\right)^{M-k}=\frac{1}{2^{M-1}}\sum_{k=0}^{R-1}\binom{M}{k}. \tag{15}$$

In the best conceivable case $R=1$, giving $p_{\mathrm{fail}}=1/2^{M-1}$.

To account for the spatial resources offered by the ensemble we consider the application of $\hat{U}_f$ to the ensemble containing M members as equivalent to M oracle calls. We must then compare this ensemble algorithm to a classical random algorithm that uses M oracle calls. In the classical random algorithm one begins by choosing x_1 randomly and evaluating $f(x_1)$. The next step is to choose $x_2\neq x_1$, evaluate $f(x_2)$ and compare the result to $f(x_1)$. If the two differ then f is balanced. If not, pick x_3 which differs from both x_1 and x_2, and compare $f(x_3)$ to $f(x_2)$ and $f(x_1)$, etc. The algorithm terminates when f returns different outcomes or has been evaluated on $2^n/2+1$ different inputs. The classical random algorithm never misidentifies a constant function and identifies a balanced function f as constant only when $f(x_1)=f(x_2)=\cdots=f(x_M)$. The probability of failure is the probability with which this occurs. The outcome $f(x_1)=0$ occurs with probability $(N/2)/N$. Given $f(x_1)=0$, $f(x_2)=0$ occurs with probability $(N/2-1)/(N-1)$. Continuing, the probability that $x_1,\ldots,x_M$ are all such that $f(x_k)=0$ is

$$p^0_{\mathrm{classical}}=\frac{N/2}{N}\frac{N/2-1}{N-1}\cdots\frac{N/2-M+1}{N-M+1}=\binom{N/2}{M}\Big/\binom{N}{M}. \tag{16}$$

Similarly the probability $p^1_{\mathrm{classical}}$ that $x_1,\ldots,x_M$ are all such that $f(x_k)=1$ can be computed and it turns out to be same as $p^0_{\mathrm{classical}}$. Thus the probability of failure is

$$p_{\text{fail}}^{\text{classical}} = 2\binom{N/2}{M} \Big/ \binom{N}{M}. \tag{17}$$

As expected, this turns out to be same as the failure fraction given in Eq. (9) for the quantum ensemble version with the unrealistic assumption that every molecule is in a different computational basis state. However, note that for $N/2 > k > 0$, $(N/2-k)/(N-k) < 1/2$, which implies that

$$p_{\text{fail}}^{\text{classical}} < 2\left(\frac{1}{2}\right)^M \leqslant p_{\text{fail}}. \tag{18}$$

Thus the probability of failure for the classical random algorithm is strictly less than that of the ensemble quantum version discussed here.

The central issue is therefore one of interpreting density matrices. What is relevant here are the inferences that can be drawn from measurement outcomes on quantum systems whose states are described by density matrices. In all cases the density matrix merely provides the probability distribution for outcomes of various measurements. The accuracy with which such a distribution is realized improves with an increasingly large ensemble. Imagine a single quantum system which is handed over to us with no information about it. What quantum state or density matrix will we be able to assign to it? To express our complete lack of information about this system we have to assign equal weightage to all possible outcomes in all bases and therefore a density matrix proportional to identity is the best choice. In this extreme case, measurement yields one of all possible outcomes and one cannot reliably infer anything from this. The density operator merely reflects our lack of knowledge in the state of the system. It is only when measurements are performed on many copies described by the same density operator that outcomes or more precisely, the average outcome, carry any meaningful information. This is no more than standard statistical sampling and for an n qubit density matrix proportional to the identity, the variance scales as 2^{2n}, indicating that one typically needs $O(2^n)$ samples (measurement outcomes) to make sensible inferences from measurements. Accordingly the ensemble size would have to scale as $O(2^n)$ before we can consider this density operator to have been realized accurately, at least in terms of measurement outcome averages. Here the ensemble begins to appear as a collection of quantum systems with states described according to the density matrix. It should be noted that, for this version of the DJ algorithm, the situation is less dire since inferences are made from measurements on the target qubit alone. Hence n does not appear in the failure probability in Eq. (15). However, as clear from Eq. (18) a classical random algorithm does the task better.

These ensemble computing ideas might work for other algorithms and give a genuine speed up over classical or classical random algorithms. One possibility is efficient simulation of quantum systems [12] and our result does not pertain to this. It is worthwhile to explore the exact implications of this model which will be taken up elsewhere.

Numerous stimulating discussions with R.B. Griffiths and Kavita Dorai are acknowledged. The research effort is funded by the NSF Grant No. 0139974.

[1] E. Knill, I. Chuang, and R. Laflamme, Phys. Rev. A **57**, 3348 (1997).
[2] I.L. Chuang, N. Gershenfeld, M.G. Kubinec, and D.W. Leung, Proc. R. Soc. London, Ser. A **454**, 447 (1998).
[3] D.G. Cory, M.D. Price, and T.F. Havel, Physica D **120**, 82 (1998).
[4] L. J. Schulman and U. Vazirani, in *Proceedings of the 31st Annual ACM Symposium on the Theory of Computation (STOC), El Paso, Texas, 1999* (ACM Press, New York, 1999), p. 322, e-print quant-ph/9804060.
[5] P.O. Boykin, T. Mor, V. Roychowdhury, F. Vatan, and R. Vrijen, Proc. Natl. Acad. Sci. U.S.A. **99**, 3388 (2002).
[6] F.M. Woodward and R. Brüschweiler, e-print quant-ph/0006024.
[7] J.M. Myers, A.F. Fahmy, S.J. Glaser, and R. Marx, Phys. Rev. A **63**, 032302 (2002).
[8] R. Cleve, A. Ekert, C. Macchiavello, and M. Mosca, Proc. R. Soc. London, Ser. A **454**, 339 (1998).
[9] D. Deutsch and R. Jozsa, Proc. R. Soc. London, Ser. A **439**, 553 (1992).
[10] W.S. Warren, Science **277**, 1688 (1997).
[11] A. Brazier and M.B. Plenio, e-print quant-ph/0304017.
[12] E. Knill and R. Laflamme, Phys. Rev. Lett. **81**, 5672 (1998).