# Optical simulation of quantum algorithms using programmable liquid-crystal displays

Graciana Puentes,[1] Cecilia La Mela,[1] Silvia Ledesma,[1] Claudio Iemmi,[1] Juan Pablo Paz,[1,2] and Marcos Saraceno[3]
[1]*Departamento de Física "J. J. Giambiagi," FCEN, UBA, Pabellón 1, Ciudad Universitaria, 1428 Buenos Aires, Argentina*
[2]*Theoretical Division, MSB213, Los Alamos National Laboratory, Los Alamos, New Mexico 87545, USA*
[3]*Departamento de Física, Tandar, CNEA, Av del Libertador 8250, 1429 Buenos Aires, Argentina*
(Received 26 May 2003; published 27 April 2004)

We present a scheme to perform an all optical simulation of quantum algorithms and maps. The main components are lenses to efficiently implement the Fourier transform and programmable liquid-crystal displays to introduce space dependent phase changes on a classical optical beam. We show how to simulate Deutsch-Jozsa and Grover's quantum algorithms using essentially the same optical array programmed in two different ways.

DOI: 10.1103/PhysRevA.69.042319 PACS number(s): 03.67.Lx, 42.25.−p, 42.30.Kq, 61.30.−v

The classical wave optics analogy of quantum information processing [1] is based on the fact that the quantum state of a system evolves according to a wave equation and satisfies the superposition principle. In recent years, this analogy has been studied in detail due to the current interest in quantum information and computation. It has been established that using classical optical waves it is possible to simulate the behavior of quantum computers [2–6]. This kind of simulation is inefficient since it requires a number of classical resources that scales exponentially with the number of quantum bits (qubits) being simulated. A possible strategy to do this for a system of $n$ qubits is to consider the profile of the classical electric-field amplitude in a laser beam $E(x)$ as the analog of the probability amplitude of a quantum state. As position is used to label the states one generally speaks in this case of position cbits [2,3]. The maximum number of orthogonal quantum states we could accommodate is fixed by the width of the beam, which must grow exponentially with the number of qubits. This imposes stringent limitations on the size of the largest quantum computer that could in principle be simulated in this way [3,6,7]. However, optical simulations of quantum algorithms not only constitute a beautiful and simple way to illustrate the power of quantum computers but also provide a way to shed light on some of their basic properties. Optical implementations of quantum algorithms performed in recent years include the pioneering work of Kwiat *et al.* [4,5] and the all optical simulation of several iterations of Grover's search algorithm [1,8] performed by Bhattacharya *et al.* [6]. In our work we present a scheme based on the use of optoelectronic devices (liquid-crystal TVs, LCTVs) which could enable us to simulate a large variety of quantum algorithms and quantum maps in a programmable way. The idea is very simple: We use position cbits to represent the quantum amplitude of a state using the (complex) electric-field amplitude of a laser beam $E(x)$. It should be noted that in the simulated quantum computer the $2^n$ classical wave amplitudes at given pixels represent the $2^n$ probability amplitudes in a given basis. This shows clearly the exponential scale up of resources, since only $n$ qubits (rather than $2^n$ pixels) would be needed in a true quantum computer. As basic tools to simulate an algorithm we use sequences of lenses and LCTV displays. Lenses will be used as an efficient way to implement the Fourier transform of the incoming electric-field amplitude (since the electric amplitude in the focal plane of a spherical lens is the Fourier transform of the incident amplitude). A set of LCTVs, polarizers, and retarder plates are used to implement an operation whose only effect is to change the phase of the electric field in a position dependent way. By using sequences of phase modulators and lenses we could, in principle, optically simulate any unitary operator. We will use the standard notation for quantum information. So, the state of a quantum bit will be noted using the Dirac notation between brackets "$|\rangle$". For example, the most general state $|\phi\rangle$ for $n=1$ will be represented by a normalized superposition $|\phi\rangle=\alpha|0\rangle+\beta|1\rangle$ of the states of the computational basis $|0\rangle$ and $|1\rangle$. The states of the computational basis will be noted generically as $|x\rangle$. In order to see how the optical simulation acts, one can reason as follows: A phase modulator acts as an operator which is diagonal in the $|x\rangle$ basis and can be written as $U_V=\exp[-iV(x)]$. Such kind of modulator in the frequency plane (i.e., the Fourier transform of $|x\rangle$) acts as an operator which is diagonal in the momentum basis and can be written as $U_K=\exp[-iT(p)]$. By applying sequences of appropriately chosen operators like $U_V$ and $U_K$ one can approximate any unitary operator [1]. The algorithm one is simulating can be changed online by programming the way in which the LCTV screens act (i.e., the local phase shifts they introduce).

In practice, there are limitations on the number of iterations due to the decay of the beam intensity after each optical component which is due to absorption and diffractive effects. However, current technology would enable one to perform a few iterations of several interesting quantum maps using this idea.

Here, we illustrate this basic scheme implementing an all optical simulation of two quantum algorithms that can be very simply presented in term of a sequence of operations like the above: (i) Deutsch-Jozsa algorithm and (ii) Grover's search algorithm. Let us describe these two algorithms very briefly. In both cases the algorithms use an "oracle" which evaluates a certain function. The goal of both algorithms is to find a property of the function by only calling the oracle. In case (i) (Deutsch-Jozsa) the function takes values over the first $N$ integers and it is either equal to 0 or to 1. Moreover, there is a promise: The function is either constant (it takes the same value for all integers) or balanced (it is equal to 0 for half the points in the domain and to 1 for the other half of the domain points). Our task is to discriminate between these two *global* properties of the function. Classically we would need to evaluate $f(x)$, at worst, $N/2+1$ times. The quantum algorithm [1] requires only one query to the quantum oracle [the quantum oracle is an operator $U_{\mathrm{oracle}}$ which transforms the state $|x\rangle$ into $(-1)^{f(x)}|x\rangle$]. The algorithm works as follows: The initial state is an equally weighted superposition of $N$ quantum states $\Sigma_x|x\rangle$ (we skip an overall normalization for notation convenience). Then we call the oracle that, as mentioned above, applies a phase shift to each state depending on the value of the function. The state is transformed into $\Sigma_x(-1)^{f(x)}|x\rangle$. Finally, we perform the Fourier transform of this state (a Hadamard transform was used in the original proposal but this can be changed into a Fourier transform without any change in the algorithm). It turns out that the probability to detect the state $|x=0\rangle$ is simply equal to $[\Sigma_x(-1)^{f(x)}/N]^2$. This probability is therefore equal to zero if the function is balanced and to unity if the function is constant. Thus, by detecting the final state we find out what class does the function in the oracle belongs to.

The second quantum algorithm we simulate is Grover's search algorithm [1,8] which is useful to find a marked item in a database with $N$ registers. The marked item is associated with a function $f(x)$, which is zero for all registers except for the marked one, where it is equal to 1. The goal of the algorithm is to find for what integer is the function equal to 1. Classically, on the average we would need to evaluate this function $N/2$ times to find the marked item. Grover's algorithm finds this item with a number of queries to the quantum oracle that scales as $\sqrt{N}$. As before, the initial state is an equally weighted superposition of all $N$ quantum states. Each iteration of the algorithm consists of applying two operators $U_{oracle}$ and $U_{IAA}$, where IAA denotes the inverse around average operation. Grover's oracle $U_{oracle}$ acts in the same way as the Deutsch-Jozsa oracle: it maps the state $|x\rangle$ into $(-1)^{f(x)}|x\rangle$. The operator $U_{IAA}$ is diagonal in momentum basis and can be written as $F^{-1}U_0F$ where $F$ is a Fourier transform and $U_0$ is the operator that changes the sign of the state $|x=0\rangle$. One can show that the effect of this operator is to invert all amplitudes about the average one. The algorithm can be extended to a database with $m$ marked items. In such case, after approximately $\pi\sqrt{N/m}/4$ iterations, all the amplitude is concentrated in the marked items. For the particular case of $N=4$ (or $N/m=4$) the algorithm finds the marked item after a single iteration [8].

The experimental setup used to optically simulate both quantum algorithms is very similar. For the Deutsch-Jozsa algorithm it is basically sketched in Fig. 1. A beam of the 457 nm line from an argon laser is expanded and filtered so that we can choose a homogeneous portion of the wave front to represent a quantum state, initially prepared in an equally weighted superposition of all inputs. To build the Deutsch-Jozsa oracle, we used the first lens to collimate the beam and a spatial light modulator (SLM), consisting of a Sony LCTV. This device, in combination with two polarizers and two wave plates, can act as a pure phase modulator [10]. The LCTV (model LCX012BL) was extracted from a commercial video projector and is a VGA resolution panel ($640\times480$ pixels) with square pixels of 34 $\mu$m size separated by a distance of 41.3 $\mu$m. After the SLM, we introduce a spherical lens (135 mm of focal length) that Fourier transforms the amplitude. Finally we detect the intensity distribution using a charge-coupled device (CCD) camera placed behind an objective of a microscope to get a magnified image. The CCD camera is driven by an 8 bits frame grabber. Some experimental considerations can be done. In order to simulate the quantum algorithms we must identify the two logical levels "zero" and "one." To this end, we measure the light captured by the CCD camera. For each experiment we select a dynamical range where the response of the CCD is linear. Working with the highest dynamical range that verifies this condition, we measure the light arriving to the CCD through the complete optical system. This operation lets us normalize the light in each CCD pixel independently of the losses due to reflections and absorptions in the optical components and the deformations introduced by aberrations of the optical system. Also we can correct the small separation of the beam profile from an homogeneous one. In other terms, the CCD camera registers a value of 0 for the "zero" logical level and a value in the range [0, 255] for the logical level "one." After the normalization the obtained values are 0 and 1 within the error provided by the frame grabber.

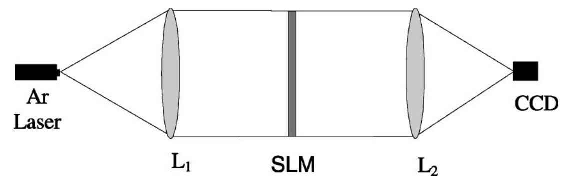


FIG. 1. Configuration for Deutsch–Jozsa's algorithm. The 457 nm line of an Argon Laser is filtered, expanded, and collimated in order to illuminate homogeneously the oracle simulated by the spatial light modulator (SLM). In the focal plane of the second lens, a CCD captures the intensity distribution around the axis.

We tested the simulation of Deutsch-Jozsa algorithm for various input functions by programming the LCTV to simulate different oracles. The phase distribution introduced by the oracle is always $\exp\,(i\pi(f(x))$. Thus, to represent this profile on the SLM we program the phase of all the pixels corresponding to points where $f(x)=1$ $[f(x)=0]$ to be equal to $\pi$ (0). A squared entrance pupil illuminated homogeneously with a precision better than 5% and with an arbitrary side size of 88 pixels, determines the size of the optically simulated Hilbert space. With this type of LCTV we could simulate the Deutsch-Jozsa algorithm in a Hilbert space of dimensions up to $N=640\times480$. In the upper part of Fig. 2 we show a sample of the oracle operator corresponding either to constant (a) or to balanced functions (b), (c), and (d). White and black areas, respectively, correspond to a phase change of 0 or $\pi$. Below we show the intensity of light registered by the CCD camera. In Fig. 3, we show separately a more entropic function where each pixel takes a random value (0 or 1), with the constraint that the function is balanced.

The algorithm is such that there is no light around the origin if and only if the input function is balanced. On the

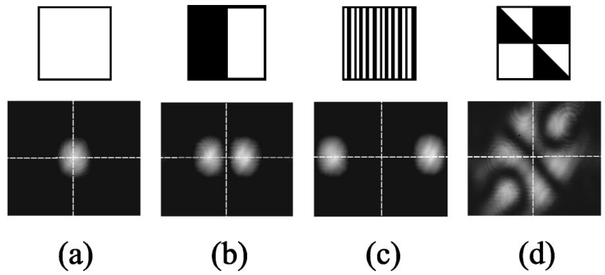


FIG. 2. Top: Different oracle configurations in Deutsch–Jozsa algorithm corresponding to various setups of the programmable LCTV. They are associated with constant (a) or balanced functions (b), (c), and (d). Bottom: The intensity in the CCD camera is nonzero at the zero frequency only if the function is constant, as predicted by the algorithm. The center of the dotted crosses indicates this frequency.

contrary, all the light is concentrated in the origin if the function is constant. This effect is clearly seen in Figs. 2 and 3. From the optical point of view the algorithm has a very clear interpretation (which makes it look almost trivial). The amplitude near the origin in the focal plane of the second lens ($L_2$) is proportional to the averaged amplitude after the SLM. This is the case because the lens implements a Fourier transform and the zero mode of the Fourier transform is just the average of the transformed function. Thus, to determine if the function is constant or balanced Deutsch-Jozsa algorithm simply computes the average value of this function (which is equal to 0 for balanced functions and to 1 for constant ones). The architecture used to simulate a single iteration of Grover's algorithm consists of a convergent optical processor, basically sketched in Fig. 4.

This configuration allows us to obtain the Fourier transform of the input in the conjugate plane of the source with an arbitrary magnification (which depends on the input position) and the image of the input in its conjugate plane. We can see that the first part of the system is a modified version of that used for the Deutsch-Jozsa's algorithm. The advantage of this setup becomes clear when one needs to manipulate the spatial frequencies of the input in the Fourier domain. Again, the homogeneous portion of the wave front defines an initial state with equal amplitude for every item. The first SLM is used to represent the Grover's oracle. The distribution emerging from this oracle is collected by the lens $L_1$ (135 mm focal length). In the focal plane of this lens the Fourier transform of the oracle is obtained. A second LCTV

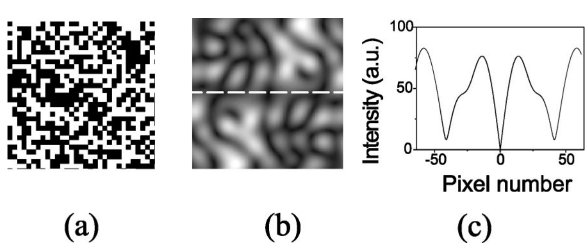


FIG. 3. (a) Oracle configuration in Deutsch-Jozsa algorithm corresponding to a random balanced function. (b) Intensity in the CCD camera, (c) intensity profile corresponding to the dashed line marked in (b).

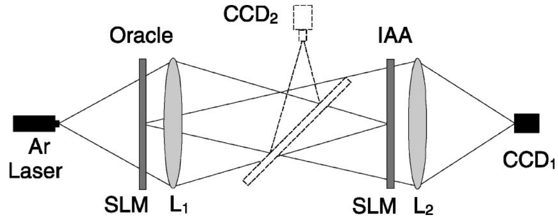


FIG. 4. Convergent optical processor for the simulation of Grover's algorithm. The first SLM modulates the phase of the initial homogenous wave front to mark the items being searched for. Next, the IAA operation is carried out by the second SLM. The CCD1 captures the output signal. In dotted line we show how the addition of a flipper mirror allows us to simulate the Deustch-Jozsa algorithm by using the same configuration.

identical to that used in the first SLM is placed in that plane to represent the IAA operation. Again, a set of polarizers and wave plates are included in order to reach the desired phase modulation. The procedure to program the LCTV's to act as Grover's oracle, as IAA, or as Deutsch-Jozsa oracle is essentially the same. This shows the versatility of LCTV's as basic components of optical simulations of quantum algorithms. Since the phase modulation introduced in both SLM's is digitally controlled, the phase matching condition [11] is easily satisfied. Finally, the emerging signal is collected by a second lens $L_2$ (250 mm focal length) and the final image is registered by the CCD1. The recorded image is spatially inverted in order to obtain $F^{-1}$. The sequence of operations performed by the optical setup is therefore $F^{-1}U_0FU_{oracle}$ which indeed represents one iteration of Grover's algorithm. For the correct performance of this optical simulation, it is crucial to have good spatial matching between the oracle and the IAA. To achieve this we used an alignment test based on spatial filtering techniques specially developed for optical processors using two LCTVs [12].

With the described setup we perform only one iteration of Grover's algorithm. In this case the probability to find the marked item at the output is equal to one when the ratio between the size of the base ($N$) and the number of marked states ($m$) is $N/m=4$. We verified this by using different base sizes: For a four-state database we chose items to be represented by one-dimensional 1D–strips ($88\times 22$ pixels). Using the same entrance pupil ($88\times 88$ pixels) we can accommodate a 2D-array of 16–state database with a side size is 22 $\times 22$ pixels. In both cases we used a square of $4\times 4$ pixels with a phase shift of $\pi$ to perform the IAA operation in the Fourier plane. In order to obtain maximum contrast, the optimal size of the phase shifting dot was determined considering the diffraction effects, i.e., we chose that the amount of energy affected by this phase shifting area be equal to the amount of energy left unaffected. The results obtained for the different bases are shown in Fig. 5. In Fig. 5(a) we show the way in which a square entrance pupil is used to represent the four items in the database. The four possible outputs, after an iteration of Grover's algorithm for each oracle, are shown in Fig. 5(b). The intensity profiles for these outputs are shown in Fig. 5(c). So, the marked item is indeed found after one iteration (and the output energy does not depend on the marked item). Finally, in Fig. 6(a) a scheme of the 2D 16–

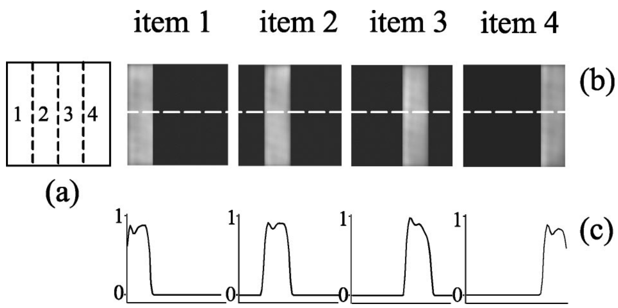


FIG. 5. (a) The different items of the four-state database. The upper images (b) are registered when the oracle marks one by one the four items. The lower intensity profiles (c) correspond to the lines marked on the images.

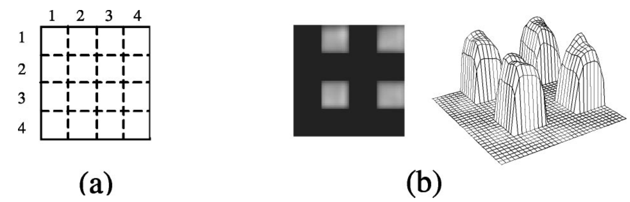


FIG. 6. (a) Squared entrance pupil, where the different numbers label the items of the 2D 16–state database. (b) Intensity output, and the corresponding profile, when the oracle marks 1–2, 1–4, 3–2, and 3–4 labeled states.

item database is presented. The output intensity obtained when marking multiple items is shown in Fig. 6(b). It can be seen that the output energy when four items are marked is equal to the output energy in the runs shown in Fig. 5.

It is clear that the optical array shown in Fig. 4 can be easily adapted to simulate Deutsch-Jozsa algorithm. In order to register the Fourier transform of the input database the light which would be focused onto the second SLM is redirected by a mirror to a camera. To this end we can use a flipper mirror and switch the position of the camera from CCD1 to CCD2. Then the Deutsch-Jozsa oracle is programmed onto the first SLM. Therefore, the same optical array can be programmed to simulate two algorithms (more generally, it could be programmed to simulate a large variety of quantum unitary maps, see below). To iterate these algorithms one needs to build a sequence of such units. The strategy followed by Bhattacharya *et al.* [6] based on a laser pulse bouncing in a lossy cavity is not feasible here due to the unidirectional nature of the LCTVs we used. However, iterations are possible by using a ringlike configuration (collecting light from a beam splitter) or by other similar configurations that are under study. Although for the Grover algorithm presented here the inversion of the Fourier transform has been digitally obtained, if an iterative process is carried out the inversion must be optically done. This could be implemented by using an additional lens at the final step of the optical circuit. Nevertheless, it should be pointed out that intensity losses due to the pixelated structure of the LCTVs are an important limitation to the number of possible iterations available with current LCTVs. Our estimate for this is of the order of four round trips for the present optical elements. It is well known that optical simulations of quantum algorithms are exponentially inefficient. In optical architectures the scalability is limited not only by the size and resolution of the display used to represent the quantum states, but by the validity of the paraxial approximation (required for the lenses to accurately implement the Fourier transform). In our case we could represent up to nine qubits in a 1D array (512 pixels) using almost all the LCTV width. Even in this case the paraxial approximation remains valid in our setup with an error lower than 0.01%.

Both algorithms simulated here belong to the class of kicked unitary maps, which can be constructed as a sequence of operators diagonal in position ($U_V$) or in momentum basis ($U_K$). As mentioned above, our setup allows, in principle, the optical simulation of all such unitary operators. Some of them are of great interest in the context of the study of quantum chaos. The use of this scheme to optically implement simulations of quantum analogs of classically chaotic dynamics is an interesting avenue that will be pursued in the future (see Ref. [13] for connected ideas on the optical implementation of baker's map).

We specially thank Professor Mario Marconi for a fruitful discussion. This work was supported by grants from ANPCyT (Grant No. PICT 09000), Ubacyt, Fundacion Antorchas, and Conicet (Grant No. PIP420/98).

[1] *Quantum Information and Computation*, edited by M. Nielsen and I. Chuang (Cambridge University Press, Cambridge, 2000).

[2] R. J. C. Spreeuw, Found. Phys. **28**, 361 (1998).

[3] R. J. C. Spreeuw, Phys. Rev. A **63**, 062302 (2001).

[4] N. J. Cerf, C. Adami, and P. G. Kwait, Phys. Rev. A **57**, R1477 (1998).

[5] P. Kwiat, J. Mitchell, P. Schwindt, and A. White, J. Mod. Opt. **47**, 257 (2000).

[6] N. Bhattacharya, H. B. van Linden van den Heuvell, and J. R. C. Spreew, Phys. Rev. Lett. **88**, 137901 (2002).

[7] R. Blume-Kohout, C. M. Caves, and I. H. Deutsch, Found. Phys. **32**, 1641 (2002).

[8] L. K. Grover, Phys. Rev. Lett. **79**, 325 (1997).

[9] In the original presentation of both algorithms Hadamard transforms are used instead of Fourier transforms. However, in both cases these transformations can be interchanged with no alterations. Moreover, they can be implemented with a lens which performs a 2D Fourier transform. In the case of general algorithms (or maps) one can use lenses to implement the unidimensional Fourier transform provided one uses a spatial encoding of states which is translation invariant along one axis.

[10] A. Marquez, C. Iemmi, I. Moreno, J. A. Davis, J. Campos, and M. J. Yzuel, Opt. Eng. **40**, 2558 (2001).
[11] G. L. Long, Y. S. Li, W. L. Zhang, and L. Niu, Phys. Lett. A **262**, 27 (1999); P. Hoyer, Phys. Rev. A **62**, 052304 (2000).
[12] J. Nicolás, J. Campos, C. Iemmi, J. Moreno, and M. J. Yzuel, Appl. Opt. **41**, 1505 (2002).
[13] J. H. Hannay, M. V. Berry, Physica D **1**, 267 (1980); J. H. Hannay and J. P. Keating, and A. Ozorio de Almeida, Nonlinearity **7**, 1327 (1994).