

Implementing Logic Gates and the Deutsch–Jozsa Quantum Algorithm by Two-Dimensional NMR Using Spin- and Transition-Selective Pulses

T. S. Mahesh,* Kavita Dorai,* Arvind,† and Anil Kumar*^{‡,1}

*Department of Physics and ‡Sophisticated Instruments Facility, Indian Institute of Science, Bangalore 560012, India; and

†Department of Physics, Guru Nanak Dev University, Amritsar 143005, India

Received June 2, 2000; revised September 19, 2000

Quantum logical operations using two-dimensional NMR have recently been described using the scalar coupling evolution technique [J. Chem. Phys. 109, 10603 (1998)]. In the present paper, we describe the implementation of quantum logical operations using two-dimensional NMR, with the help of spin- and transition-selective pulses. A number of logic gates are implemented using two and three qubits with one extra observer spin. Some many-in-one gates (or Portmanteau gates) are also implemented. Toffoli gate (or AND/NAND gate) and OR/NOR gates are implemented on three qubits. The Deutsch–Jozsa quantum algorithm for one and two qubits, using one extra work qubit, has also been implemented using spin- and transition-selective pulses after creating a coherent superposition state in the two-dimensional methodology. © 2001 Academic Press

Key Words: two-dimensional NMR; quantum computation; logic gate; Toffoli gate; Deutsch–Jozsa algorithm.

INTRODUCTION

Quantum computing exploits the intrinsic quantum nature of physical systems and is therefore more powerful than classical computing for a certain class of problems. While some problems like factorization gain exponential speed up (1), some others like database-search gain polynomial speed up (2). A quantum computer works on two-level quantum systems known as “quantum bits” or “qubits.” The fact that qubits can exist in a coherent superposition of basis states is exploited in a quantum computer. However, retaining such a coherent superposition for a long enough time is one of the major hurdles in quantum computation. Since nuclear spins in liquids are efficiently isolated from the rest of their environment, the coherent superposition lasts for several hundreds of milliseconds. Therefore, high-resolution NMR of weakly coupled nuclear spins in liquids is one of the potential candidates for such studies. Nuclear spins with $I = \frac{1}{2}$ have generally been used, but recently, a quadrupole nuclear spin system with $I = \frac{3}{2}$ has also

been used (3). Several workers have contributed to quantum information processing using NMR (4–32).

Recently, logical gates have also been demonstrated using two-dimensional NMR (33). In the two-dimensional method, the states of the “computation” spins (or input spins) are encoded by the transitions of an extra “observer” spin (Fig. 1). The observer spin is first allowed to evolve for a time t_1 during which the input qubits remain in their initial state (Fig. 2). After the frequency labeling period t_1 , the computation is performed on the input qubits. The observer spin is again allowed to evolve for a time t_2 and detected. The experiment starts from a mixed input state, the computation is performed simultaneously on all the input states, and one gets a mixed output state. One can also create a superposition of input qubits in the beginning of the experiment by applying a $\pi/2$ pulse on the input qubits and subsequently killing the coherence by a gradient pulse (33). Various steps in two-dimensional NMR, namely preparation, evolution, mixing, and detection, have a close correspondence with the steps in quantum computing, namely creation of initial states, labeling of the initial states, computation, and reading of output states, respectively (33) (Fig. 2). In this respect, two-dimensional NMR offers some advantages. One of the advantages of the two-dimensional method is that it resolves all of the input and the output states and correlates them. This correlation between input and output states in the two-dimensional experiment makes the result of the computation graphic. For example, a SWAP gate, which exchanges the states of two qubits, can be implemented in NMR by selectively interchanging populations of zero-quantum levels. Since the one-dimensional NMR spectrum of homonuclear spins after the execution of a SWAP gate is indistinguishable from that of an equilibrium spectrum, the one-dimensional SWAP gate was earlier demonstrated after the creation of a nonequilibrium state (25, 34). However, as will be shown here (Fig. 3) and has been demonstrated earlier (33), the two-dimensional method starts from the equilibrium state and yields a spectrum characteristic of the SWAP gate.

Two-dimensional logic gates have been carried out so far by using the scalar coupling evolution technique (33). This paper

¹ To whom correspondence should be addressed. E-mail: anilnmr@physics.iisc.ernet.in.

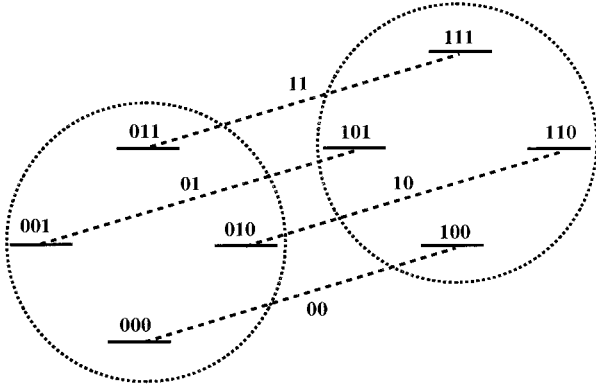


FIG. 1. Energy level diagram of a three-spin system ($I_0 I_1 I_2$) showing two two-spin subsystems corresponding to states 0 and 1 of observer spin I_0 . The transitions of I_0 are labeled by the states of input qubits ($I_1 I_2$).

describes a two-dimensional NMR implementation of several two- and three-qubit gates with one extra observer spin, using spin- and transition-selective pulses. The first implementation of the Deutsch–Jozsa quantum algorithm using two-dimensional NMR is also demonstrated here. One of the advantages of the selective pulse method is that it makes the computation simple and straightforward.

EXPERIMENTAL

A. Logic Gates

A weakly coupled three-spin system ($I_0 I_1 I_2$) can be viewed as a combination of two two-spin subsystems with the observer spin (I_0) being in state 0 or 1 (Fig. 1). There are four observer spin transitions labeled 11, 10, 01, and 00, which correspond to the states of the input qubits ($I_1 I_2$) in these transitions. During the computation (mixing) period, various transitions of the input qubits are inverted, yielding various gates as listed in Table 1. The results of 24 one-to-one reversible gates for the three-spin system having one observer and two input qubits are shown in Fig. 3. The NOP gate is implemented by doing no operation during the computation period. In this gate, each input state corresponds to same output state after the computation. Various NOT gates are implemented by inverting one or both of input qubits (I_1, I_2), using spin-selective π pulses. XOR (or control-NOT) and XNOR gates are implemented by inverting two similar transitions of the same input spin, one in each subsystem. We label XOR and XNOR gates as XOR1, XNOR1 and XOR2, XNOR2 depending on whether the result of the operation is stored on spin I_1 or I_2 , respectively. For example, the XOR1 gate is implemented by inverting transitions 001–011 and 101–111 of I_1 and the result of the gate is stored on I_1 (Table 1, Fig. 3). A SWAP gate can be implemented by selectively interchanging the populations of zero-quantum levels of each subsystem (Table 1, Fig. 3). This can be

achieved by the use of a cascade of three transition-selective, noncommuting π pulses on regressively connected transitions during the computation period (25, 34). It may be noted that while all the gates of Ref. (33) are reproduced here, with several new gates added, the methodology used for the computation is different except for the NOP and NOT gates.

Although a few basic gates such as NOT, AND, OR, and XOR are sufficient to carry out a given computation, other gates are also useful since they may reduce the number of pulses. Therefore “Portmanteau gates” which perform more than one operation have been implemented in one-dimensional NMR using selective pulses (25). Similar gates have been implemented here using two-dimensional NMR (Table 1, Fig. 3). Interchanging populations of double-quantum levels using a cascade of noncommuting π pulses on progressively connected transitions leads to a SWAP + NOT gate (Table 1, Fig. 3). SWAP + XOR and SWAP + XNOR gates have been implemented by inverting two pairs of regressively connected transitions in each case (Table 1, Fig. 3). Similarly, SWAP + NOT + XOR and SWAP + NOT + XNOR gates have been implemented by inverting two pairs of progressively connected transitions (Table 1, Fig. 3). The last 6 gates in Fig. 3 are direct combinations of two gates. It may be noted that the 24 gates shown in Fig. 3 form a complete set of two-qubit one-to-one mappings.

Figure 4 shows several three-qubit gates implemented on a four-spin system, using selective pulses. Once again, no operation during computation period yields the NOP gate and inverting spin I_1 yields the NOT(I_1) gate. The more interesting ones are the Toffoli gate (or AND/NAND gate) and the OR/NOR gate. The operations of Toffoli and OR/NOR gates (33, 35) are, respectively,

$$|s, t, u\rangle \rightarrow |s \oplus (t \wedge u), t, u\rangle \quad [1]$$

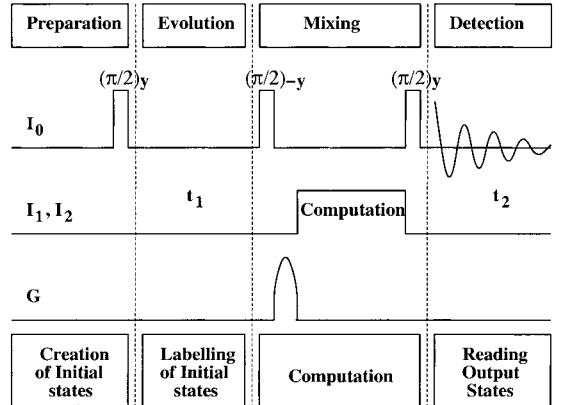


FIG. 2. Pulse scheme for the two-dimensional NMR implementation of quantum computing. The close correspondence between two-dimensional NMR and quantum computing is shown. I_0 is the observer spin and I_1 and I_2 are the input qubits. The gradient pulse G_z eliminates the unwanted transverse magnetization before the computation.

TABLE 1
Various Logic Gates with Boolean Algebra, Truth Table, and Operations Performed

Gate ^a	In	Out	Transitions inverted ^b	Gate ^a	In	Out	Transitions inverted ^b
NOP $ s, t\rangle \rightarrow s, t\rangle$	11 10 01 00	11 10 01 00	No operation	NOT (I_1) $ s, t\rangle \rightarrow \bar{s}, t\rangle$	11 10 01 00	01 00 11 10	All transitions of I_1
NOT (I_2) $ s, t\rangle \rightarrow s, \bar{t}\rangle$	11 10 01 00	10 11 00 01	All transitions of I_2	NOT (I_1, I_2) $ s, t\rangle \rightarrow \bar{s}, \bar{t}\rangle$	11 10 01 00	00 01 10 11	All transitions of I_1 and I_2
XOR1 $ s, t\rangle \rightarrow s \oplus t, t\rangle$	11 10 01 00	01 10 11 00	111 \leftrightarrow 101 011 \leftrightarrow 001	XOR2 $ s, t\rangle \rightarrow s, s \oplus t\rangle$	11 10 01 00	10 11 01 00	111 \leftrightarrow 110 011 \leftrightarrow 010
XNOR1 $ s, t\rangle \rightarrow \overline{s \oplus t}, t\rangle$	11 10 01 00	11 00 01 10	100 \leftrightarrow 110 000 \leftrightarrow 010	XNOR2 $ s, t\rangle \rightarrow s, \overline{s \oplus t}\rangle$	11 10 01 00	11 10 00 01	101 \leftrightarrow 100 001 \leftrightarrow 000
SWAP $ s, t\rangle \rightarrow t, s\rangle$	11 10 01 00	11 01 10 00	Invert ZQ: 110 \leftrightarrow 111 010 \leftrightarrow 011 101 \leftrightarrow 111 001 \leftrightarrow 011 110 \leftrightarrow 111 010 \leftrightarrow 011	SWAP + NOT $ s, t\rangle \rightarrow \bar{t}, \bar{s}\rangle$	11 10 01 00	00 10 01 11	Invert DQ: 110 \leftrightarrow 111 010 \leftrightarrow 011 100 \leftrightarrow 110 000 \leftrightarrow 010 110 \leftrightarrow 111 010 \leftrightarrow 011
SWAP + XOR1 $ s, t\rangle \rightarrow s \oplus t, s\rangle$	11 10 01 00	01 11 10 00	101 \leftrightarrow 111 001 \leftrightarrow 011 110 \leftrightarrow 111 010 \leftrightarrow 011	SWAP + XOR2 $ s, t\rangle \rightarrow t, s \oplus t\rangle$	11 10 01 00	10 01 11 00	110 \leftrightarrow 111 010 \leftrightarrow 011 101 \leftrightarrow 111 001 \leftrightarrow 011
SWAP + XNOR1 $ s, t\rangle \rightarrow \overline{s \oplus t}, s\rangle$	11 10 01 00	11 01 00 10	100 \leftrightarrow 110 000 \leftrightarrow 010 100 \leftrightarrow 101 000 \leftrightarrow 001	SWAP + XNOR2 $ s, t\rangle \rightarrow t, \overline{s \oplus t}\rangle$	11 10 01 00	11 00 10 01	100 \leftrightarrow 101 000 \leftrightarrow 001 100 \leftrightarrow 110 000 \leftrightarrow 010
SWAP + NOT + XOR1 $ s, t\rangle \rightarrow \bar{s} \oplus \bar{t}, \bar{s}\rangle$	11 10 01 00	00 10 11 01	101 \leftrightarrow 111 001 \leftrightarrow 011 100 \leftrightarrow 101 000 \leftrightarrow 001	SWAP + NOT + XOR2 $ s, t\rangle \rightarrow \bar{t}, \bar{s} \oplus \bar{t}\rangle$	11 10 01 00	00 11 01 10	110 \leftrightarrow 111 010 \leftrightarrow 011 100 \leftrightarrow 110 000 \leftrightarrow 010
SWAP + NOT + XNOR1 $ s, t\rangle \rightarrow \bar{s} \oplus \bar{t}, \bar{s}\rangle$	11 10 01 00	10 00 01 11	100 \leftrightarrow 110 000 \leftrightarrow 010 110 \leftrightarrow 111 010 \leftrightarrow 011	SWAP + NOT + XNOR2 $ s, t\rangle \rightarrow \bar{t}, \bar{s} \oplus \bar{t}\rangle$	11 10 01 00	01 10 00 11	100 \leftrightarrow 101 000 \leftrightarrow 001 101 \leftrightarrow 111 001 \leftrightarrow 011
NOT (I_1) + XOR2 $ s, t\rangle \rightarrow \bar{s}, \bar{s} \oplus t\rangle$	11 10 01 00	01 00 10 11	All I_1 transitions and 111 \leftrightarrow 110 011 \leftrightarrow 010	NOT (I_2) + XOR1 $ s, t\rangle \rightarrow s \oplus \bar{t}, \bar{t}\rangle$	11 10 01 00	10 01 00 11	All I_2 transitions and 111 \leftrightarrow 101 011 \leftrightarrow 001
NOT(I_1) + XNOR2 $ s, t\rangle \rightarrow \bar{s}, \bar{s} \oplus t\rangle$	11 10 01 11	00 01 11 10	All I_1 transitions and 101 \leftrightarrow 100 001 \leftrightarrow 000	NOT (I_2) + XNOR1 $ s, t\rangle \rightarrow \overline{s \oplus \bar{t}}, \bar{t}\rangle$	11 10 01 00	00 11 10 01	All I_2 transitions and 100 \leftrightarrow 110 000 \leftrightarrow 010
SWAP + NOT (I_1) $ s, t\rangle \rightarrow \bar{t}, s\rangle$	11 10 01 00	01 11 00 10	Invert ZQ and all transitions of I_1	SWAP + NOT (I_2) $ s, t\rangle \rightarrow t, \bar{s}\rangle$	11 10 01 00	10 00 11 01	Invert ZQ and all transitions of I_2

^a $|s, t\rangle$ represents the state of input qubits (I_1, I_2).

^b Order of transitions is important as all transitions may not commute.

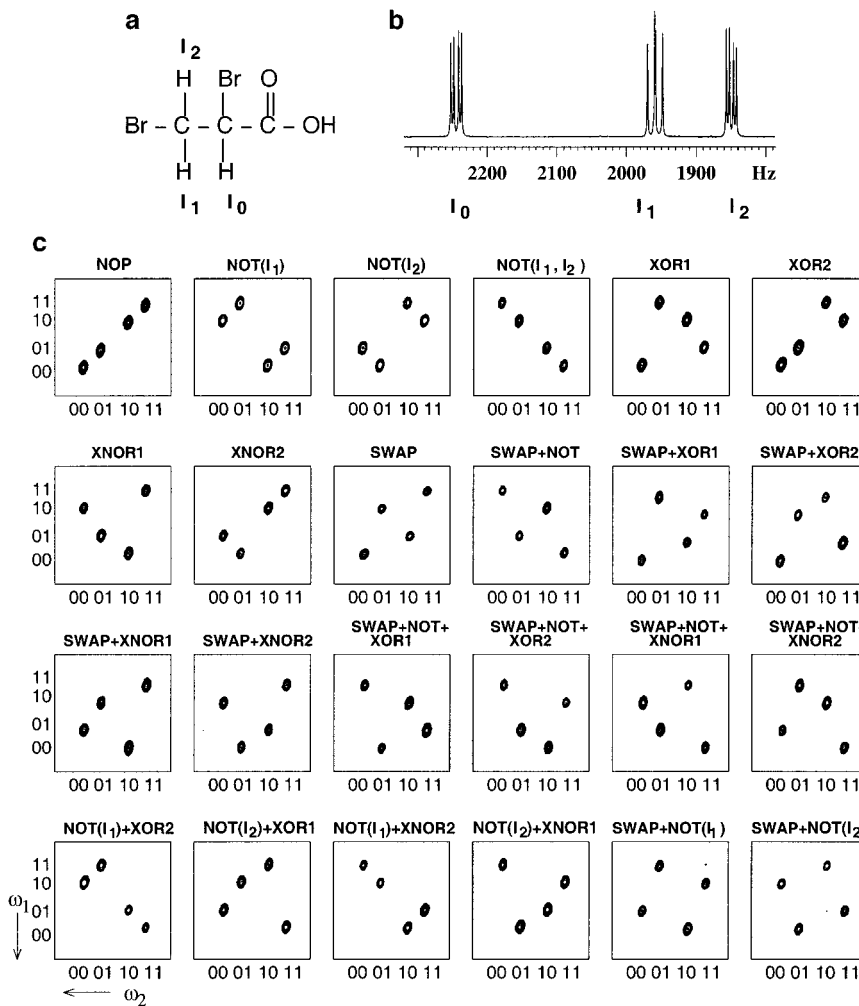


FIG. 3. ^1H NMR spectrum (b) of 2,3-dibromopropionic acid (a) in CDCl_3 recorded on a Bruker DRX-500 spectrometer at 300 K. (c) Shows observer spin (I_0) spectra corresponding to various gates implemented using spin- and transition-selective pulses. The pulse scheme used is shown in Fig. 2 and various transitions of input qubits (I_1, I_2) inverted during the computation are listed in Table 1. The spin-selective pulses were 10 ms long and the transition-selective pulses were 100–300 ms long. Low-power rectangular/Gaussian pulses were utilized for various gates. The phase of the computation pulses was cycled through $(x, -x)$ to suppress the distortions due to pulse imperfections. All experiments were carried out in the time domain with 256 t_1 values and 256 complex data points along t_2 and with two scans for each t_1 point. Zero filling to 512×512 complex data points was done prior to 2D Fourier transformation. All plots are shown in magnitude mode.

and

$$|s, t, u\rangle \rightarrow |s \oplus (t \vee u), t, u\rangle, \quad [2]$$

where $\oplus \equiv$ addition modulo 2, $\wedge \equiv$ AND, $\vee \equiv$ OR, and s, t , and u are the states of the control spin I_1 and the input spins I_2 and I_3 . The Toffoli gate is a universal gate for reversible computation. This gate can be implemented by inverting the control spin transitions 011 and 111, by using a pair of transition-selective π pulses. Inverting all transitions of the control spin except the transitions 010 and 110 leads to the OR/NOR gate (Fig. 4).

B. The Deutsch–Jozsa Algorithm

The Deutsch–Jozsa (DJ) algorithm demonstrates the power of quantum computing over classical computing (36). In this algo-

rithm, we consider functions from N -bit domain space to one-bit range space. A function is called constant if it gives the same output for any input and is called balanced if it gives one output for half the number of inputs and another for the remaining half. Given an N -bit binary function which is either constant or balanced, classically up to $2^{(N-1)} + 1$ operations are required to determine whether the function is constant or balanced, whereas the DJ algorithm requires only a single evaluation. The DJ algorithm has been implemented using one-dimensional NMR by several research groups (10, 11, 18, 25).

In the Cleve version of the DJ algorithm (37), a binary function f is encoded in a unitary transformation by the propagator U_f by including an extra work qubit such that

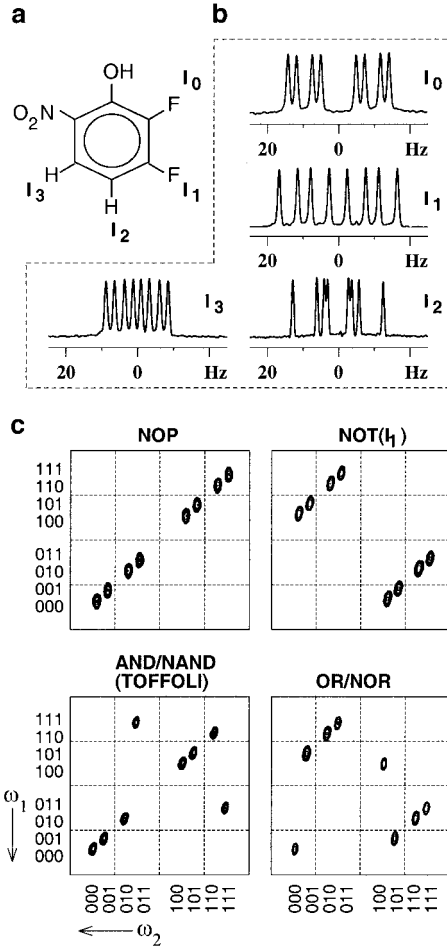


FIG. 4. ^{19}F and ^1H NMR spectra (b) of 2,3-difluoro-6-nitrophenol (a) in CDCl_3 (with one drop of D_2O to induce the exchange of the hydroxy proton and hence to suppress its coupling to fluorine nuclei) recorded on a Bruker DRX-500 spectrometer at 300 K. (c) Shows ^{19}F spectra of observer spin I_0 corresponding to various gates. The pulse scheme used is shown in Fig. 2 and various transitions of control qubit I_1 inverted during the computation period are described in the text. The spin-selective pulses were 1 ms long and the transition-selective pulses were 200 ms long. The phase of the computation pulses was cycled through $(x, -x)$ to suppress distortions due to pulse imperfections. All experiments were carried out in the time domain with 1024 t_1 values and 256 complex data points along t_2 and two scans for each t_1 point. Zero filling to 1024×1024 complex data points was done prior to 2D Fourier transformation. All plots are shown in magnitude mode.

$$|r\rangle|s\rangle \xrightarrow{U_f} |r \oplus f(s)\rangle|s\rangle, \quad [3]$$

where $|r\rangle$ and $|s\rangle$ are, respectively, the states of work qubit (I_0) and input qubit (I_1). The four possible functions for the single-bit DJ algorithm are listed in Table 2. This algorithm has been implemented on the thermal equilibrium state and does not require the creation of a pure initial state (11, 25). The two-dimensional pulse scheme used for implementing the DJ algorithm is shown in Fig. 5. The experiment begins with both

TABLE 2
The Four Possible Binary Functions (f_1 – f_4) for One-Bit DJ Algorithm

s	Const.		Bal.	
	f_1	f_2	f_3	f_4
0	0	1	0	1
1	0	1	1	0

qubits in a superposition of states, achieved by a nonselective $(\pi/2)_y$ pulse. This is followed by an evolution period t_1 , propagator U_f , and detection period t_2 . The transformations corresponding to f_1 and f_2 are, respectively, a unity operation and a spin-selective π_x pulse on the work qubit. The transformations corresponding to f_3 and f_4 are implemented by transition-selective π_x pulses, respectively, on the 10–11 and 00–01 transitions of the work qubit (25). The results of the algorithm for all four functions are shown in Fig. 6. The constant or balanced nature of the function is identified by the presence or absence of signal from the input qubit (I_1) (Fig. 6) (25). The expected signals are also shown schematically in Fig. 6. For two of the functions f_2 and f_4 , the expected signals are calculated in Table 3. The spin states ($|r\rangle|s\rangle$) before computation can be paired (connected by a curved line) in such a way that each pair represents a transition of the input qubit (I_1). Each transition of the input qubit is labeled by the state of the remaining spin, i.e., work qubit. Corresponding output pairs of spin states after computation can be calculated by using Expression [3]. From each output pair, the corresponding transition of the input spin after computation can be identified. If two spins have flipped in the output pair, then the transition becomes nonobservable and will be labeled as zero quantum (ZQ) or double quantum (DQ).

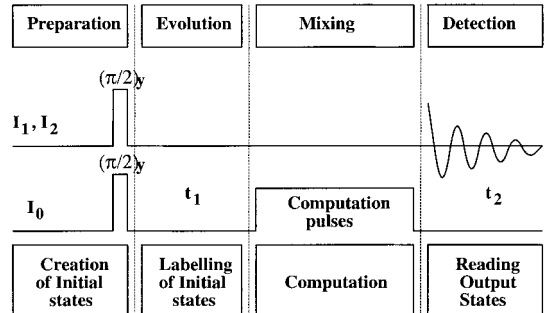


FIG. 5. Pulse scheme for the two-dimensional NMR implementation of the DJ algorithm. I_0 is the work qubit and I_1 and I_2 are the input qubits.

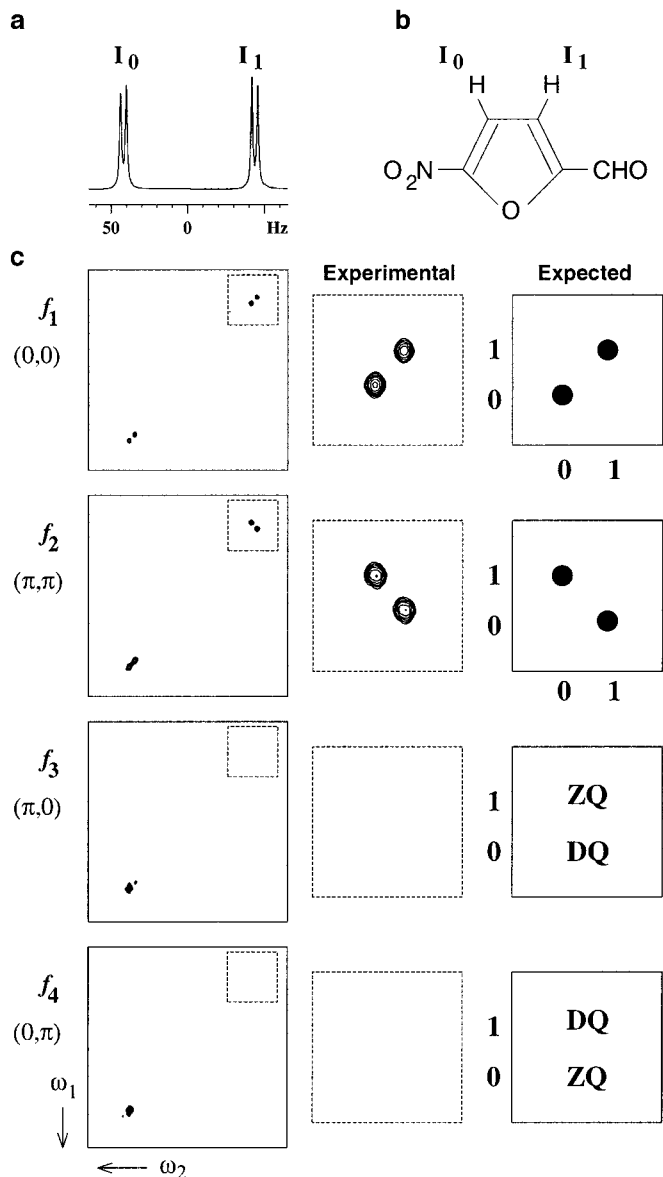


FIG. 6. ^1H NMR spectrum (a) of 5-nitrofuldehyde (b) in C_6D_6 on a Bruker DRX-500 spectrometer at 300 K. The results of the DJ algorithm for various functions f_1 – f_4 in Table 2 are shown in (c). I_0 is the work qubit and I_1 is the input qubit. Expansions of the I_1 parts of the spectra are shown in dotted boxes. The expected pattern is also shown in each case. The expected patterns for f_2 and f_4 are described in Table 3. The spin-selective pulses were 10 ms long and the transition-selective pulses were 100 ms long. The phase of the computation pulses was cycled through $(x, -x)$ to suppress the distortions due to pulse imperfections. All experiments were carried out in the time domain with 256 t_1 values and 512 complex data points along t_2 and two scans for each t_1 point. Zero filling to 512×512 complex data points was done prior to 2D Fourier transformation. All plots are shown in magnitude mode.

Implementing the DJ algorithm on two input qubits requires three qubits including one work qubit. The algorithm can be described as

TABLE 3
The Input–Output Correlations for the Functions f_2 and f_4 of the One-Bit DJ Algorithm

In		Out		
$ r\rangle s\rangle$	s -Transitions	f_2	$ f_2 \oplus r\rangle s\rangle$	s -Transitions
11	1	1	01	0
10		1	00	
01	0	1	11	1
00		1	10	

$ r\rangle s\rangle$	s -Transitions	f_4	$ f_4 \oplus r\rangle s\rangle$	s -Transitions
11	1	0	11	DQ
10		1	00	
01	0	0	01	ZQ
00		1	10	

$$|r\rangle|s\rangle|t\rangle \xrightarrow{U_f} |r \oplus f(s, t)\rangle|s\rangle|t\rangle, \quad [4]$$

where $|r\rangle$, $|s\rangle$, and $|t\rangle$ are the states of the work qubit (I_0) and of two input (I_1, I_2) qubits, respectively. The eight possible two-bit binary functions are listed in Table 4. The pulse scheme is the same as that in Fig. 5. Once again, the transformations corresponding to f_1 and f_2 are, respectively, unity operation and a spin-selective π_x pulse on the work qubit. The unitary transformations encoding the six balanced functions f_3 – f_8 are implemented by selective pulses on the transitions of the work qubit, taken two at a time, i.e., $[0, 0, \pi, \pi]$, $[\pi, \pi, 0, 0]$, $[\pi, 0, \pi, 0]$, $[0, \pi, 0, \pi]$, $[\pi, 0, 0, \pi]$, and $[0, \pi, \pi, 0]$, where 0 denotes no pulse on that particular transition (11, 25). The results of the algorithm for all eight functions are shown in Fig. 7. Once again, a function is constant only if signals from all the input qubits are present, otherwise the function is balanced (Fig. 7) (11, 25). Table 5 describes the input–output correlation for f_2 and f_4 in a two-qubit DJ algorithm. Here the s - and t -transitions are labeled by the states $|r\rangle|t\rangle$ and $|r\rangle|s\rangle$, respectively. Other details of Table 5 are similar to that of Table 3.

TABLE 4
The Eight Possible Binary Functions (f_1 – f_8) for the Two-Bit DJ Algorithm

s	t	Const.		Bal.					
		f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8
0	0	0	1	0	1	1	0	1	0
0	1	0	1	0	1	0	1	0	1
1	0	0	1	1	0	1	0	0	1
1	1	0	1	1	0	0	1	1	0

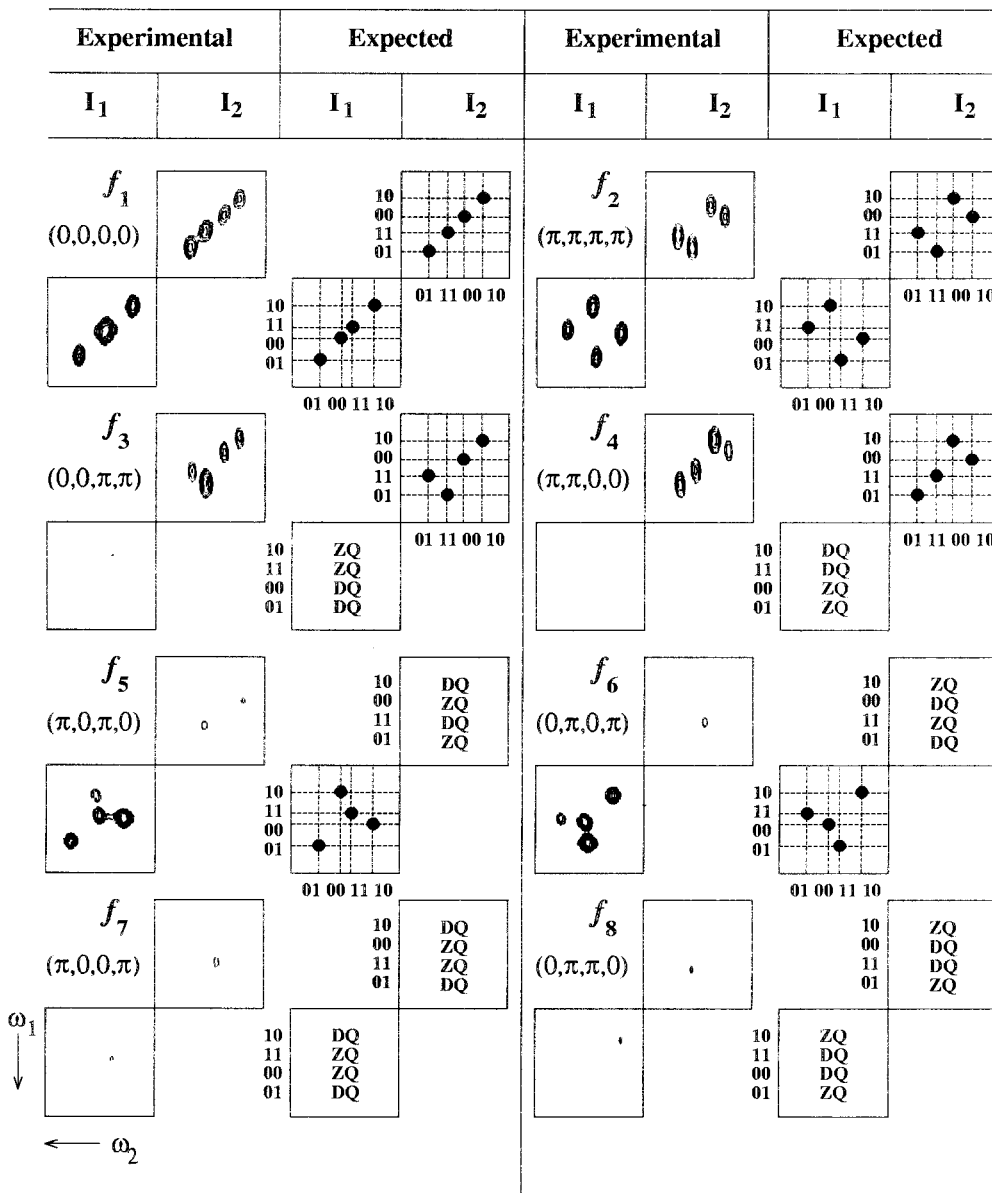


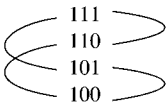
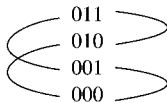
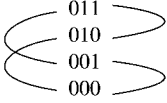
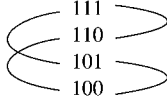
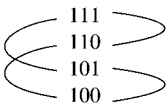
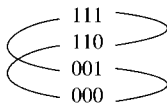
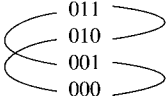
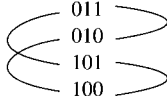
FIG. 7. The results of the DJ algorithm on 2,3-dibromopropionic acid (Figs. 3a and 3b) in CDCl_3 for various functions f_1 – f_8 listed in Table 4. I_0 is the work qubit and I_1 and I_2 are the input qubits. Only expansions of the I_1 and I_2 parts of the spectra are shown. The expected pattern is also shown in each case for comparison. The expected patterns for f_2 and f_4 are described in Table 5. All experiments were carried out on a Bruker DRX-500 spectrometer at 300 K. The spin-selective pulses were 10 ms long and the transition-selective pulses were 100–300 ms long. The phase of the computation pulses was cycled through $(x, -x)$ to suppress the distortions due to pulse imperfections. A total of 1024 complex data points along t_2 were acquired for 256 t_1 values with two scans for each t_1 point. Zero filling to 1024×1024 complex data points was done prior to 2D Fourier transformation. All plots are shown in magnitude mode.

CONCLUSIONS

Using two-dimensional NMR, several two- and three-qubit gates have been implemented on three and four weakly coupled spin systems and one- and two-qubit DJ algorithms on two and three weakly coupled spin systems by utilizing spin- and transition-selective pulses. The use of selective pulses over scalar evolution has several advantages as well

as some disadvantages. The advantages of selective pulses are that they lead to simple logic, need only a few pulses, and work very well. The disadvantages are that one needs long low-power RF pulses during which relaxation and RF inhomogeneity effects degrade the performance of the selective pulses and that ideal selectivity may be difficult to achieve. Furthermore, for the selective pulse experiments, one needs resolved transitions. However, many of these

TABLE 5
The Input–Output Correlations for the Functions f_2 and f_4 of the Two-Bit DJ Algorithm

In				Out		
s-Transitions	$ r\rangle s\rangle t\rangle$	t-Transitions	f_2	s-Transitions	$ f_2 \oplus r\rangle s\rangle t\rangle$	t-Transitions
11 10		11 10	1 1 1 1	01 00		01 00
01 00		01 00	1 1 1 1	11 10		11 10
s-Transitions	$ r\rangle s\rangle t\rangle$	t-Transitions	f_4	s-Transitions	$ f_4 \oplus r\rangle s\rangle t\rangle$	t-Transitions
11 10		11 10	0 0 1 1	DQ DQ		11 00
01 00		01 00	0 0 1 1	ZQ ZQ		01 10

difficulties can be overcome by the use of heteronuclear spins such as ^{15}N , ^{13}C , ^{19}F , and ^{31}P , where the magnitudes of the spin–spin couplings are large, the transitions are well spread out, and selectivity is easier to achieve using pulses of shorter durations. Further improvement can be achieved by orienting molecules in liquid crystal matrices (38, 39). Attempts are continuing in these directions in our laboratory as well as in others.

ACKNOWLEDGMENTS

The use of 400- and 500-MHz FTNMR spectrometers of the Sophisticated Instruments Facility (SIF), Indian Institute of Science, funded by Department of Science and Technology, New Delhi, is gratefully acknowledged. We also thank Prof. K. V. Ramanathan and Dr. G. A. Naganagowda of SIF for discussions.

REFERENCES

1. P. W. Shor, Polynomial-time algorithms for prime factorization and discrete algorithms on quantum computer, *SIAM Rev.* **41**, 303–332 (1999).
2. L. K. Grover, Quantum mechanics helps in searching for a needle in a haystack, *Phys. Rev. Lett.* **79**, 325–328 (1997).
3. A. K. Khitrin and B. M. Fung, Nuclear magnetic resonance quantum logic gates using quadrupolar nuclei, *J. Chem. Phys.* **112**, 6963–6965 (2000).
4. D. G. Cory, A. F. Fahmy, and T. F. Havel, Ensemble quantum computing by NMR spectroscopy, *Proc. Natl. Acad. Sci. USA* **94**, 1634–1639 (1997).
5. N. Gershenfeld and I. L. Chuang, Bulk spin-resonance quantum computation, *Science* **275**, 350–356 (1997).
6. I. L. Chuang, L. M. K. Vandersypen, X. Zhou, D. W. Leung, and S. Lloyd, Experimental realization of a quantum algorithm, *Nature* **393**, 143–146 (1998).
7. I. L. Chuang, N. Gershenfeld, and M. Kubinec, Experimental implementation of fast quantum searching, *Phys. Rev. Lett.* **80**, 3408–3411 (1998).
8. I. L. Chuang, N. Gershenfeld, M. G. Kubinec, and D. W. Leung, Bulk quantum computation with nuclear magnetic resonance: Theory and experiment, *Proc. R. Soc. Lond. A* **454**, 447–467 (1998).
9. J. A. Jones, R. H. Hansen, and M. Mosca, Quantum logic gates and nuclear magnetic resonance pulse sequences, *J. Magn. Reson.* **135**, 353–360 (1998).
10. J. A. Jones and M. Mosca, Implementation of a quantum algorithm on a nuclear magnetic resonance quantum computer, *J1. Chem. Phys.* **109**, 1648–1653 (1998).
11. N. Linden, H. Barjat, and R. Freeman, An implementation of the Deutsch–Jozsa algorithm on a three-qubit NMR quantum computer, *Chem. Phys. Lett.* **296**, 61–67 (1998).
12. N. Linden, H. Barjat, R. J. Carbajo, and R. Freeman, Pulse sequences for NMR quantum computers: How to manipulate nuclear spins while freezing the motion of coupled neighbours, quant-ph/9811043.
13. S. S. Somaroo, D. G. Cory, and T. F. Havel, Expressing the operations of quantum computing in multiparticle geometric algebra, *Phys. Lett. A* **240**, 1–7 (1998).
14. D. G. Cory, A. E. Dunlop, T. F. Havel, S. S. Somaroo, and W. Zhang, The effective Hamiltonian of the Pound–Overhauser controlled-NOT gate, quant-ph/9809045.
15. N. Linden, E. Kupce, and R. Freeman, NMR quantum logic gates for homonuclear spin systems, quant-ph/9907003.

16. M. D. Price, S. S. Somaroo, C-H. Tseng, J. C. Gore, A. Famy, T. F. Havel, and D. G. Cory, Construction and implementation of NMR quantum logic gates for two spin systems, *J. Magn. Reson.* **140**, 371–378 (1999).
17. M. D. Price, S. S. Somaroo, A. E. Dunlop, T. F. Havel, and D. G. Cory, Generalized methods for the development of quantum logic gates for an NMR quantum information processor, *Phys. Rev. A* **60**, 2777–2780 (1999).
18. Arvind, Kavita Dorai, and Anil Kumar, Quantum entanglement in the NMR implementation of the Deutsch–Jozsa algorithm, quant-ph/9909067.
19. J. A. Jones and E. Knill, Efficient refocussing of one spin and two spin interactions for NMR quantum computation, *J. Magn. Reson.* **141**, 322–325 (1999).
20. F. M. Woodward and B. Bruschweiler, Solution of the Deutsch–Jozsa problem by NMR ensemble computing without sensitivity scaling, quant-ph/0006024.
21. D. Collins, K. W. Kim, W. C. Holton, H. Sierzputowska-Gracz, and E. O. Stejskal, NMR quantum computation with indirectly coupled gates, *Phys. Rev. A* **62**, 022304/1–4 (2000).
22. R. Marx, A. F. Fahmy, J. M. Myers, W. Bermel, and S. J. Glaser, Approaching 5-bit NMR quantum computing, *Phys. Rev. A* **62**, 012310/1–8 (2000).
23. J. A. Jones, V. Vedral, A. Ekert, and G. Castagnoli, Geometric quantum computation using nuclear magnetic resonance, *Nature* **403**, 869–871 (2000).
24. H. K. Cummins and J. A. Jones, Resonance offset tailored pulses for NMR quantum computation, quant-ph/0008034.
25. Kavita Dorai, Arvind, and Anil Kumar, Implementing quantum-logic operations, pseudopure states, and the Deutsch–Jozsa algorithm using noncommuting selective pulses in NMR, *Phys. Rev. A* **61**, 042306/1–7 (2000).
26. F. M. Woodward and R. Bruschweiler, Solution of the Deutsch–Jozsa problem by NMR ensemble computing without sensitivity scaling, quant-ph/0006024.
27. M. D. Price, T. F. Havel, and D. G. Cory, Multi-qubit logic gates in NMR quantum computing, *N. J. Phys.* **2**, 10/1–9 (2000).
28. Y. Sharf, T. F. Havel, and D. G. Cory, Spatially encoded pseudopure states for NMR quantum information processing, quant-ph/0005076.
29. D. G. Cory, R. Laflamme, E. Knill, L. Viola, T. F. Havel, N. Boulant, G. Boutis, E. Fortunato, S. Lloyd, R. Martinez, C. Negrevergne, M. Pravia, Y. Sharf, G. Teklemariam, Y. S. Weinstein, and W. H. Zurek, NMR based quantum information processing: Achievements and prospects, quant-ph/0004104.
30. Y. Sharf, D. G. Cory, S. S. Somaroo, T. F. Havel, E. Knill, and R. Laflamme, A study of quantum error correction by geometric algebra and liquid-state NMR spectroscopy, *Mol. Phys.* **98**, 1347 (2000).
31. L. M. K. Vandersypen, M. Steffen, G. Breyta, C. S. Yannoni, R. Cleve, and I. L. Chuang, Experimental realization of order-finding with a quantum computer, quant-ph/0007017.
32. X. Zhu, X. Fang, M. Feng, F. Du, K. Gao, and X. Mao, Experimental realization of a highly structured search algorithm, quant-ph/0007040.
33. Z. L. Madi, R. Bruschweiler, and R. R. Ernst, One- and two-dimensional ensemble quantum computing in spin Liouville space, *J. Chem. Phys.* **109**, 10603–10611 (1998).
34. Kavita Dorai and Anil Kumar, Cascades of selective pulses on connected single-quantum transitions leading to the selective excitation of multiple-quantum coherences, *J. Magn. Reson.* **114**, 155–162 (1995).
35. T. Toffoli, Languages and programming, in “Automata” (J. W. de Bakker and J. van Leeuwen, Eds.), pp. 632–644, Springer-Verlag, New York, 1980.
36. D. Deutsch and R. Jozsa, Rapid solution of problems by quantum computation, *Proc. R. Soc. Lond. A* **439**, 553–558 (1992).
37. R. Cleve, A. Ekert, C. Macchiavello, and M. Mosca, Quantum algorithms revisited, *Proc. R. Soc. Lond. A* **454**, 339–354 (1998).
38. C. S. Yannoni, M. H. Sherwood, D. C. Miller, I. L. Chuang, L. M. K. Vandersypen, and M. G. Kubinec, Nuclear magnetic resonance quantum computing using liquid crystal solvents, *Appl. Phys. Lett.* **75**, 3563–3565 (1999).
39. M. Marjanska, I. L. Chuang, and M. G. Kubinec, Demonstration of quantum logic gates in liquid crystal nuclear magnetic resonance, *J. Chem. Phys.* **112**, 5095–5099 (2000).