# Liquid-State NMR Quantum Computing

**Lieven M. K. Vandersypen**
*TU Delft, Delft, the Netherlands*

**Isaac L. Chuang**
*Massachusetts Institute of Technology, Cambridge, MA, USA*

&

**Dieter Suter**
*TU Dortmund, Dortmund, Germany*

## 1 INTRODUCTION

Since its invention, NMR spectroscopy has developed from a technique for studying physical phenomena such as magnetism into a tool for acquiring information about molecules in chemistry and biology. Furthermore, it was pointed out early on (1955), almost as an anecdote, that nuclear spins could also be used for storing information using spin echoes.[1]

This insight beautifully illustrated a notion that was developed in a very different context: information is physical and cannot exist without a physical representation.[2] In recent decades, the relationship between physics and information has been revisited from a new perspective: could the laws of physics play a role in *how* information is processed? Research into the physics of computation has shown that the answer is yes. If information is represented by systems governed by the laws of quantum mechanics, such as nuclear spins, an entirely new way of doing computation becomes possible, which is known as *quantum computation* (*QC*). Quantum computing is not just different or new; it offers an extraordinary promise, the capability of solving certain problems that are beyond the reach of any machine relying on the laws of classical physics. Apart from these promises for useful applications, the quantum mechanical perspective has many fundamentally different perspectives on ways to store, distribute, and process information, including different computational models or computation without generating entropy.

The practical realization of QC is still in its infancy. From the perspective of magnetic resonance, it is interesting that more than 40 years after the initial suggestion of using spins to represent (classical) information, NMR actually became the first technique capable of implementing QC. Since the initial demonstrations in 1997, NMR has been used to demonstrate and test most conceptual advances in the fledgling field of QC.

In this article, we first explain how quantum computers work and why they could solve certain problems so much faster than any classical computer. Next, we describe how quantum computers can be implemented using NMR techniques and what is involved in designing and implementing QC pulse sequences, preparing a suitable initial state and interpreting the output spectra. We conclude with an overview of the state of the art and the prospects for NMRQC and other QC implementations.

Good reviews of QC and information can be found in Refs 3–6. Some useful reviews of NMRQC are Refs 7–9.

## 2 QUANTUM COMPUTATION

### 2.1 Bits, Bytes, and Logic Gates

Classical and quantum computers both represent information in the form of binary digits (bit). When we use spins for representing information, we arbitrarily assign “0” to a spin up and “1” to a spin down. As shown in Figure 1, we need a sufficiently large number of spins to store the information that is to be processed. Together, they form the quantum register. Before the actual computation can start, it is necessary to initialize this quantum register into a well-defined state. Then, we process the information by applying unitary transformations $U_\mathrm{i} = \exp(-i H_\mathrm{i}\tau_\mathrm{i})$; in NMR, these unitary transformations are implemented as pulse sequences.

We may grasp the relevant aspects of quantum information processing if we start from a place that is familiar for many NMR spectroscopists, the INEPT pulse sequence (*see* ***INEPT***). This sequence was designed to transfer polarization from a high $\gamma$ nucleus to a low $\gamma$ nucleus. However, it can also be viewed as a *logic gate* (Figure 2), which flips one spin conditioned upon the orientation of a neighboring spin. This is an elementary two-bit operation known as the *controlled-NOT* or in short *CNOT* gate (within the phase corrections discussed in the section “Pulse Sequence Design”). The CNOT gate performs a NOT operation on one bit, flipping it from “0” to “1” or from “1” to “0,” if and only if the value of a second bit is “1.” The input to the logic gate is the initial state of the spins, and the output is the final state of the spins. The four possible input values and the corresponding output values are tabulated in Figure 3.

The CNOT combined with single-spin rotations provides for a *universal* set of logic gates. This means that *any* computational task can be implemented using a sufficiently large number of nuclear spins simply by concatenating CNOTs and single-spin rotations in the proper way.[10–12] In summary, *spin-1/2 nuclei in a molecule can serve as bits in a computer, and pulses and delay times provide a universal set of logic gates.*

### 2.2 Quantum Parallelism

So far, our discussion was purely classical. The Bloch-sphere picture of Figure 2 reinforces this classical view of the spins; however, nuclear spins are really quantum objects. In Dirac notation, the state of a spin can be denoted by $|0\rangle$ for a spin in the ground state (along $z$), and by $|1\rangle$ for a spin in the excited state (along $-z$), corresponding to the two classical

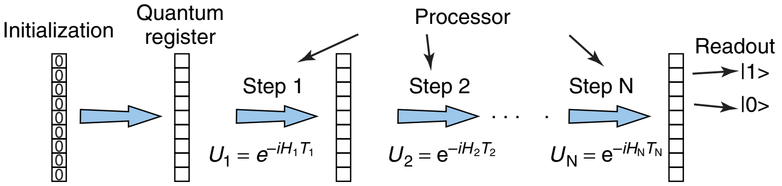

**Figure 1** Relevant steps in quantum information processing: The information is represented by the state of a set of coupled spins; together they form the quantum register. Before the computation starts they must be initialized into a well-defined state. The information is then processed by applying unitary transformations $U_i$, as required by the quantum algorithm. At the end of the computation, the result is converted into classical information in the readout process

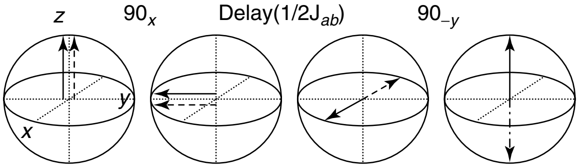

**Figure 2** The evolution of one of two coupled heteronuclear spins during an INEPT type pulse sequence, when the other spin is up (solid line) or down (dashed line). The rotating frame is set on resonance with the first spin so there is no need to refocus chemical shift. The usual readout pulse is left out. The same pulse sequence can be applied to two homonuclear spins using spin-selective pulses

| In | Out |
|---|---|
| ↑ ↑ | ↑ ↑ |
| ↑ ↓ | ↑ ↓ |
| ↓ ↑ | ↓ ↓ |
| ↓ ↓ | ↓ ↑ |

(a)

| In | Out |
|---|---|
| 0 0 | 0 0 |
| 0 1 | 0 1 |
| 1 0 | 1 1 |
| 1 1 | 1 0 |

(b)

**Figure 3** (a) Input and output states for the INEPT pulse sequence and (b) for the corresponding CNOT gate

values for a bit ("0" and "1"). Now, a spin said to be "along the $x$ axis" is, in reality, in a coherent superposition state of spin up and spin down, written as $(|0\rangle + |1\rangle)/\sqrt{2}$, a spin "along the $y$ axis" is in the state $(|0\rangle + i|1\rangle)/\sqrt{2}$, etc. A spin-1/2 particle is thus more than just an ordinary bit. *Any two-level quantum system, such as a spin-1/2 particle, can serve as a quantum bit (qubit).*

The difference between the quantum and classical descriptions becomes clear as soon as more than one quantum particle is considered. For example, it is well known that the state of $n$ interacting spins-1/2 cannot be described simply by $n$ sets of coordinates on the Bloch sphere. In order to include phenomena such as multiple-quantum coherence, we need recourse to $4^n - 1$ real numbers in the product operator expansion or equivalently to density matrices of dimension $2^n \times 2^n$. Furthermore, the evolution of a closed system of $n$ spins can only be described by $2^n \times 2^n$ unitary matrices (*see* ***Liouville Equation of Motion***). The number of degrees of freedom that need to be specified in a classical description of the state and dynamics of $n$ coupled spins thus increases exponentially with the number of spins.

Richard Feynman proposed, in 1982, that the exponential complexity of quantum systems might be put to good use for simulating the dynamics of another quantum system,[13] a task that requires exponential effort on a classical computer. David Deutsch extended and formalized this idea in 1985, and introduced the notion of "quantum parallelism".[14]

Consider a (classical) logic gate that implements a function $f$ with one input bit $x$ and one output bit $f(x)$. If $x = 0$, the gate output will be $f(0)$; if $x = 1$, the output will be $f(1)$. The analogous quantum logic gate is described by a unitary operation that transforms a qubit as

$$|0\rangle \mapsto |f(0)\rangle \quad \text{and} \quad |1\rangle \mapsto |f(1)\rangle \tag{1}$$

However, because of the possibility of preparing coherent superposition states and to the linearity of quantum mechanics, the same gate also performs the transformation

$$\frac{|0\rangle + |1\rangle}{\sqrt{2}} \quad \mapsto \quad \frac{|f(0)\rangle + |f(1)\rangle}{\sqrt{2}} \tag{2}$$

In this sense, it is possible to evaluate $f(x)$ for both input values in one step. Next, consider a different logic gate, which implements a function $g(x)$ with two input bits. We can prepare each qubit in a superposition of "0" and "1." Formally, the state of the joint system is then written as $(|0\rangle + |1\rangle) \otimes (|0\rangle + |1\rangle)/2$. Leaving out the tensor product symbol as well as any normalization factors, the state can be written as $(|0\rangle + |1\rangle)(|0\rangle + |1\rangle)$, or $|0\rangle|0\rangle + |0\rangle|1\rangle + |1\rangle|0\rangle + |1\rangle|1\rangle$, which is further abbreviated to $|00\rangle + |01\rangle + |10\rangle + |11\rangle$. Therefore, a set of two spins can be in a superposition of the four states "00", "01", "10", and "11." A quantum logic gate implementing $g(x)$ then transforms this state as

$$\begin{aligned} |00\rangle + |01\rangle + |10\rangle + |11\rangle &\mapsto |g(00)\rangle + |g(01)\rangle \\ &\quad + |g(10)\rangle + |g(11)\rangle \end{aligned} \tag{3}$$

Thus the function has been evaluated for the four possible input values in parallel. In general, a function of $n$ qubits implemented on a quantum computer can be evaluated for all $2^n$ input values in parallel. In contrast to classical computers, for which the number of parallel function evaluations increases at best linearly with their size, *the number of parallel function evaluations grows exponentially with the size of the quantum computer (the number of qubits).*

Obviously, this is true only so long as the coherent superposition states are preserved throughout the computation. This means that the computation should be completed before quantum coherence is lost due to "decoherence" processes (in NMR spin–spin and spin–lattice relaxation; *see* ***Relaxation: An Introduction***). Since some degree of decoherence is unavoidable, practical quantum computers appeared virtually impossible to build, until quantum error correction was conceived, as discussed in the section "Quantum Error Correction."

Even if the coherence time is long compared to the duration of a typical logic gate, and quantum error correction is employed, can we really access the exponential power exhibited by quantum systems? The postulates of quantum mechanics dictate that an ideal measurement of a qubit in a superposition state $|f(0)\rangle + |f(1)\rangle$ will give either "*f*(0)" or "*f*(1)," with equal probabilities, while causing instantaneous and complete decoherence. Similarly, after doing $2^n$ computations all at once, resulting in a superposition of $2^n$ output values, a measurement of the qubits randomly returns a single output value. A more clever approach is thus needed: *exploiting quantum parallelism requires the use of quantum algorithms.*

### 2.3 Quantum Algorithms

Special quantum algorithms allow one to take advantage of quantum parallelism in order to solve certain problems in far fewer steps than is possible classically. When comparing the capability of two computers to solve a certain type of problem, *the relevant criterion is not so much what resources (time, size, signal-to-noise ratio, …) are required to solve a specific instance of the problem but rather how quickly the required resources grow with the problem size.*

A particularly important criterion is whether the required resources increase exponentially or polynomially with the problem size. Exponentially difficult problems are considered intractable – they become simply impossible to solve when the problem size is large. In contrast, polynomially difficult problems are considered tractable or possible to solve. The interest in quantum computing is based on the fact that *certain problems that appear intractable (resources grow exponentially with problem size) on any classical computer are tractable on a quantum computer.*

This was shown in 1994 by Peter Shor, almost 10 years after Deutsch introduced quantum parallelism. Shor's quantum algorithm[15] allows one to find the period of a function exponentially faster than any classical algorithm. The importance of period-finding lies in that it can be translated, using some results from number theory, to finding the prime factors of integer numbers, and thus also to breaking widely used cryptographic codes. These codes are based precisely on the fact that no *efficient* classical algorithm is known for period-finding or factoring, i.e., the effort required to factor a number on classical computers increases exponentially with the number of digits of the integer to be factored. In contrast, Shor's algorithm is efficient: on a quantum computer, the effort to factor an integer increases only polynomially with the number of digits of the integer. As a result, while factoring a 1000-digit number is believed to be beyond the reach of any machine relying on the classical laws of physics, such a feat could be accomplished on a quantum computer.

The first quantum algorithm was invented by Deutsch[14] and generalized by Deutsch and Jozsa.[16] This algorithm allows a quantum computer to solve with certainty an artificial mathematical problem known as *Deutsch's problem*. Even though this algorithm does not have practical applications, it is historically significant as it provided the first steps toward Shor's algorithm, and because it is a simple quantum algorithm that can be experimentally tested.

Another class of quantum algorithms was discovered in 1996 by Lov Grover. These algorithms[17] allow quadratic speedups of certain search problems, for which there is no better approach classically than to try all $L$ candidate solutions one at a time. A quantum computer using Grover's algorithm needs to make only $\sqrt{L}$ such trials. Even though this speedup is only quadratic rather than exponential, it is still significant.

Outside the physics community, the algorithms of Shor and Grover are probably the most popular ones. However, for physicists, another class of algorithms looks much more promising: Along the lines of Feynman's original suggestion,[13] a number of algorithms have been proposed and implemented, which use quantum systems to simulate other quantum systems.[18–21] Not only is the behavior of quantum mechanical systems inherently more interesting to most physicists than the factorization of large numbers but these applications also require much smaller numbers of qubits to become competitive with classical computers. Estimates vary between 20 and 50 qubits for a "useful" quantum simulator versus thousands of qubits for factorization. As a result, quantum simulations have become an important subfield of quantum information processing. In particular, the simulation of quantum phase transitions has become an important issue.[22,23]

In the remainder of this section, we briefly review the structure of Shor's algorithm, because it is so important and at the same time gives good insight into how quantum computing works (for a more detailed explanation, see Refs 4 and 15). The crucial step in Shor's factoring algorithm is the use of the quantum Fourier transform (QFT) to find the period $r$ of the function $f(x) = a^x \bmod M$, which means $f(x)$ is the remainder after division of $a^x$ by $M$, where $M$ is the integer to be factored, and $a$ is an integer which is more or less randomly chosen.[4,15]

The QFT performs the same transformation as the (classical) discrete Fourier transform (DFT), but can be computed exponentially faster. As always, we do not have access to all the individual output values; the QFT merely allows us to *sample* the DFT but this suffices for period-finding. The $\mathrm{DFT}_N$ takes as input a string of $N$ complex numbers $x_j$ and produces as output another string of $N$ complex numbers $y_k$, such that

$$y_k = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} x_j e^{2\pi i jk/N} \tag{4}$$

For an input string with numbers that repeat themselves with period $r$, the $\mathrm{DFT}_N$ produces an output string with period $N/r$, as illustrated in the following four examples for $N = 8$ (the output strings have been rescaled for clarity)

| $r$ | Input string | | Output string | $N/r$ | |
|---|---|---|---|---|---|
| 8 | 1 0 0 0 0 0 0 0 | ↦ | 1 1 1 1 1 1 1 1 | 1 | (a) |
| 4 | 1 0 0 0 1 0 0 0 | ↦ | 1 0 1 0 1 0 1 0 | 2 | (b) |
| 2 | 1 0 1 0 1 0 1 0 | ↦ | 1 0 0 0 1 0 0 0 | 4 | (c) |
| 1 | 1 1 1 1 1 1 1 1 | ↦ | 1 0 0 0 0 0 0 0 | 8 | (d) |

If $r$ does not divide $N$, the inversion of the period is approximate. Furthermore, the fast Fourier transform (FFT) turns shifts in the locations of the numbers in the input string into phase factors in front of the numbers in the output string:

| | | | |
|---|---|---|---|
| 1 0 0 0 1 0 0 0 | ↦ | 1 0 1 0 1 0 1 0 | (e) |
| 0 1 0 0 0 1 0 0 | ↦ | 1 0 −i 0 −1 0 i 0 | (f) |
| 0 0 1 0 0 0 1 0 | ↦ | 1 0 −1 0 1 0 −1 0 | (g) |
| 0 0 0 1 0 0 0 1 | ↦ | 1 0 i 0 −1 0 −i 0 | (h) |

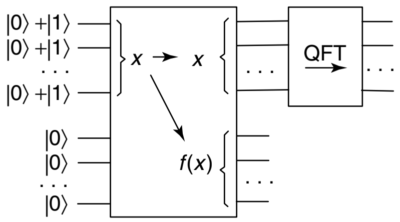

**Figure 4** Schematic diagram of the main steps in quantum algorithms for period-finding

The QFT performs exactly the same transformation, but differs from the DFT in that the complex numbers are stored in the amplitude and phase of the terms in a superposition state. The amplitude of the $|000\rangle$ term represents the first complex number, the amplitude of the $|001\rangle$ term the second number, and so forth. For clarity, we label the states $|000\rangle, |001\rangle, \ldots |111\rangle$ as $|0\rangle, |1\rangle, \ldots |7\rangle$. As an example, we see from ($f$) that the QFT transforms the state $|1\rangle + |5\rangle$ to the state $|0\rangle - \mathrm{i}|2\rangle - |4\rangle + \mathrm{i}|6\rangle$.

The QFT is incorporated in actual quantum algorithms as outlined in Figure 4. A first register (group of qubits) is prepared in a superposition of all its possible states. A second register is initialized to the ground state (for factoring a number $M$, the size of the second register must be at least $\log_2 M$, and the first register must be at least twice as large). For example, if register 1 has 3 qubits and register 2 has 2 qubits, the state of the system is prepared in

$$(|0\rangle + |1\rangle + |2\rangle + |3\rangle + |4\rangle + |5\rangle + |6\rangle + |7\rangle)|0\rangle \qquad (5)$$

Then the function $f(x)$ is evaluated (in NMR by applying a pulse sequence, as discussed in the sections "Bits, Bytes, and Logic Gates" and "Pulse Sequence Design"), where $x$ is the value of the first register, and the output value $f(x)$ is stored in the second register. *Since the first register is in an equal superposition of all* $|x\rangle$, *the function is evaluated for all values of* $x$ *from* 0 *to* 7 *in parallel*. For example, let $f(x) = 3$ for even $x$ and $f(x) = 1$ for odd $x$, which means the period $r$ is 2 (in real applications, we have a description of $f$ but do not know $r$ in advance). Evaluation of $f(x)$ then transforms the state of equation (5) to the state

$$\begin{aligned} &|0\rangle|3\rangle + |1\rangle|1\rangle + |2\rangle|3\rangle + |3\rangle|1\rangle \\ &\qquad + |4\rangle|3\rangle + |5\rangle|1\rangle + |6\rangle|3\rangle + |7\rangle|1\rangle \qquad (6) \\ &= (|0\rangle + |2\rangle + |4\rangle + |6\rangle)|3\rangle + (|1\rangle + |3\rangle + |5\rangle + |7\rangle)|1\rangle \qquad (7) \end{aligned}$$

We pause to point out that this state is *entangled*, which means that it cannot be written as a product of single-qubit states. The state $|00\rangle + |01\rangle + |10\rangle + |11\rangle$ is an example of an unentangled state, because it can be written as $(|0\rangle + |1\rangle)(|0\rangle + |1\rangle)$, a product of single-qubit states. In contrast, $|00\rangle + |11\rangle$ is a simple example of an entangled state.

In order to appreciate the role of the QFT, suppose we now measure the second register in equation (7) (this measurement can be left out but simplifies the explanation). The state of the first register will collapse to either

$$|0\rangle + |2\rangle + |4\rangle + |6\rangle \quad \text{or} \quad |1\rangle + |3\rangle + |5\rangle + |7\rangle \qquad (8)$$

depending on whether the measurement of register 2 gave "3" or "1." We see that all eight possible outcomes are still equally likely, so the result of a measurement does not give us any useful information. But if we apply the QFT, the first register will be transformed to

$$|0\rangle + |4\rangle \quad \text{or} \quad |0\rangle - |4\rangle \qquad (9)$$

Now a measurement of the first register does give useful information, because only multiples of $N/r$ are possible outcomes, in this example, "0" and "4." This procedure was verified successfully in 2001, when Vandersypen *et al.* demonstrated the experimental factorization of the number 15, using an NMR quantum computer.[24]

This concludes the quantum part of the computation. From the measurement result, a classical computer can efficiently calculate the inverted period $N/r$, and thus also $r$, with high probability of success using results from number theory. Now that $r$ is known, the factors of the integer $M$ can be quickly computed as well, with high probability (the probability of success can be further increased by repeating the whole procedure a few times).

We conclude with two final remarks on quantum algorithms. (i) Quantum computing cannot offer any speed-up for many common tasks, such as adding up two numbers or word processing, which can already be completed efficiently on a classical computer. (ii) There are many exponentially difficult problems which no currently available quantum algorithm could help solve faster than is possible classically. It would be somewhat disappointing from a practical viewpoint if no other applications were found; however, our understanding of the connection between physics and information and computation has already changed dramatically.

### 2.4 Quantum Error Correction

Any QC must be completed within the coherence time, in NMR $T_2$ and $T_1$, as pointed out in the section "Quantum Parallelism." $T_1$ and $T_2$ processes alter the state of the qubits and are therefore a source of errors. For many years, this requirement led to widespread pessimism about the practicality of quantum computers. In 1995, however, Peter Shor and Andrew Steane independently discovered quantum error correction[25,26] and showed that *it is possible to correct for truly random errors caused by decoherence.*

This came as quite a surprise, because quantum error correction has to overcome three important obstacles: (i) the no-cloning theorem, which states that it is not possible to perfectly duplicate unknown quantum states[5]; (ii) measuring a quantum system affects its state; and (iii) errors on qubits can be arbitrary rotations in Hilbert space, or their state can even leak out of the part of Hilbert space that is used for computation. These are much more severe issues than in the case of classical computers, where information errors can only be bit flips. Quantum error correction requires many extra operations and extra qubits (ancillae), which might introduce more errors than are corrected, especially because the effect of decoherence increases with the number of entangled qubits.[27,28] Therefore, a second surprising result[29] was that *provided the error rate (probability of error per elementary operation) is below a certain threshold, and given a fresh supply of ancilla qubits in the ground state, it is possible to perform arbitrarily long QCs.*

The threshold error rate is currently estimated to be of the order of $10^{-2}$ to $10^{-4}$.[30] The actual error rate in NMRQC is approximated by $1/2JT_2$, where $2JT_2$ is roughly the number of operations that can be computed within the coherence time and $J$ is a typical coupling constant. For small molecules, the error rate is typically on the order of $0.1–1\%$. But a remarkable implication of quantum error correction is that if (i) a molecule is found that achieves the accuracy threshold and (ii) the required ancilla spins can be reset, both $T_1$ and $T_2$ could, in principle, be infinitely lengthened by applying an error correction pulse sequence.

### 2.5 Alternative Computational Models

The computation that we have described here uses the so-called network model, which very closely resembles the models commonly used to describe classical computers. Within the field of quantum computing, however, several other computational models have been developed, which have different attractive features for theoretical or experimental aspects. They include (i) the "one-way quantum computer,"[31] where a sequence of measurements is performed on a complicated quantum state; (ii) QC by linear optics and measurements[32] where measurements and feed-forward techniques create effective interactions between single photons; (iii) adiabatic QC[33]. Here, the solution of the problem corresponds to the ground state of a suitable system Hamiltonian. An adiabatic transfer is performed to bring the system from a suitable initial state into the ground state of the actual Hamiltonian. While this computational model is not designed with NMR as the physical implementation in mind, it can be implemented by NMR. Examples include the simulation of quantum phase transitions[23] or an adiabatic factoring algorithm.[34]

## 3 NMR QUANTUM COMPUTERS

### 3.1 Pulse Sequence Design

The translation of abstract quantum algorithms or function evaluations into actual pulse sequences may appear obscure at first sight. However, systematic techniques[35,36] exist to make pulse sequence design relatively straightforward. The starting point is that *each quantum algorithm can be described by a sequence of transformations under unitary operators.*

Such unitary transformations represent rotations in Hilbert space (a multidimensional extension of the Bloch sphere). Examples of unitary transformations include evolution during RF pulses and free evolution under the system Hamiltonian; relaxation processes give rise to nonunitary transformations. Once the desired unitary operators have been identified, *arbitrary unitary operators can be translated into sequences of single-qubit rotations and* CNOT *gates.*

These building blocks can be readily implemented in NMR (see the section "Implementation of Computations"). Decompositions into other sets of elementary gates are also possible, and can be helpful for simplifying the pulse sequences.[37,38] In any case, it is crucial that the duration of the pulse sequence design process as well as the length of the resulting pulse sequence do not increase exponentially with the problem size.

We now point out two important distinctions between QC and conventional pulse sequences. On one hand, QC sequences must be more general: *QC sequences must perform the desired transformation for arbitrary input states.*

In contrast, conventional sequences are often designed assuming a particular input state. As a first example of this difference, the sequence of Figure 2 assumes that both spins are in Zeeman states, i.e., aligned along $\pm z$. It implements the unitary operator

$$\hat{U}_{\mathrm{INEPT}} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & i & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -i & 0 \end{pmatrix} \tag{10}$$

which is similar to but different from the unitary operator for the CNOT gate, defined as

$$\hat{U}_{\mathrm{CNOT}} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{11}$$

implemented, for example, by $90^a_z\, 90^b_{-z}\, 90^b_x\, 1/2J_{ab}\, 90^b_{-y}$.

As a second example, consider the so-called Hadamard gate, defined as

$$\hat{U}_{\mathrm{Had}} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \tag{12}$$

This gate creates a superposition state starting from a basis state: it transforms $|0\rangle$ to $|0\rangle + |1\rangle$ ($z$ to $x$ in the Bloch sphere) and $|1\rangle$ to $|0\rangle - |1\rangle$ ($-z$ to $-x$). At first sight, this transformation could be done simply via a $90_y$ pulse. However, the unitary operator for $90_y$

$$\hat{U}_{90_y} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} \tag{13}$$

is different from $\hat{U}_{\mathrm{Had}}$; e.g., applying $\hat{U}_{\mathrm{Had}}$ twice has no net effect, but applying $\hat{U}_{90_y}$ twice produces $U_{180_y}$. A possible sequence that implements $\hat{U}_{\mathrm{Had}}$ exactly is $90_y 180_x$.

On the other hand, QC sequences can be more specific: *QC sequences can be specialized for a specific molecule using full knowledge of its spectral properties.*

In contrast, conventional sequences must work for any molecule, because the spectral properties of the molecule are usually not known in advance. Exact knowledge of the chemical shifts and J-coupling constants allows one not only to greatly simplify the pulse sequences but also to achieve much more accurate unitary transformations than would otherwise be possible.

Finally, while systematic procedures exist to design a pulse sequence, there is a need to develop tools for finding the pulse sequence with the shortest duration and with the smallest number of RF pulses. Even small-scale QCs easily involve tens to hundreds of gates acting on multiple spins, and precise control of the spin dynamics is difficult to maintain throughout such long sequences of operations, as shown in the next section.

### 3.2 Implementation of Computations

The implementation of QCs with NMR can be based on single-spin rotations and CNOT gates, since any quantum algorithm can be translated into these building blocks.

Although these elementary operations appear quite easy to implement, *the requirements for precision in QC experiments are unusually high, due to the large number of pulses and the quantitative nature of the information contained in the output spectra.*

Implementation of accurate *single-spin rotations* about an axis in the $x-y$ plane is relatively easy in heteronuclear molecules; yet, it can be very demanding for homonuclear spin systems because spin selectivity requires longer pulses resulting in unwanted evolution of the spins during the pulses.[39] Clearly, some degree of homonuclearity is unavoidable when more than a handful of qubits is involved.

We therefore begin by reviewing the requirements for pulse shaping[40] (*see* ***Shaped Pulses***; ***Selective Pulses***). First, the magnetization corresponding to each of the lines in a multiplet must be rotated about exactly the same axis and over exactly the same angle, i.e., off-resonance effects due to line splitting must be removed. This requires self-refocusing shaped pulses or tailored composite pulses[41,42] (*see* ***Composite Pulses***). Second, the effect of J-couplings between unselected spins must be removed, either during the pulse or later in the pulse sequence. Third, all pulses must be universal rotors, i.e., the rotation must be independent of the initial state of the spin. Fourth, the unselected spins must not be affected by the RF irradiation. This last requirement is difficult to satisfy because of transient Bloch–Siegert effects[43], which can result in substantial (tens of degrees) phase shifts of unselected spins. However, it is possible to estimate and compensate for the Bloch–Siegert shift.[44,45] Finally, simultaneous (as opposed to consecutive) pulses at two or more nearby frequencies are desirable in order to keep pulse sequences short, but transient Bloch–Siegert shifts greatly deteriorate such simultaneous rotations.[46,47] Although a clever correction technique[48] can deliver very accurate simultaneous rotations at nearby frequencies, simultaneous pulses on well-coupled spins may still excite multiple quantum coherences.[49] As the number of qubits increases and the systems get more complex, it thus becomes imperative to develop pulses that achieve *exactly* the desired unitary transformation, even in the presence of experimental errors and interactions that cannot be controlled.[50] These techniques, which may be considered an extension of composite pulses,[41] use optimal control theory to generate unitary transformations that are as close as possible to the target operations.[51] While they are useful in weakly coupled systems, they are essential for strongly coupled systems.[52]

There are a number of hardware requirements for successful execution of QC experiments. Good RF coil homogeneity is crucial in avoiding excessive signal attenuation and related errors. Furthermore, it is desirable that one frequency source and transmitter board be available per qubit. If there are more qubits than spectrometer channels, the carrier frequency must be jumped to the appropriate frequencies throughout the pulse sequence, or phase-ramping techniques must be employed.[53] A dedicated frequency source for each qubit also makes it easy to keep track of the rotating frame of each spin and to apply all the pulses on any given spin with the correct relative phase. This removes the need to refocus chemical shift evolution,[40] which involves extra pulses. Alternatively, software rotating frames can be created by detailed bookkeeping of the time elapsed since the beginning of the pulse sequence such that the evolution of the rotating frame of any given spin with respect to the carrier reference frame can be calculated. The phases of the pulses throughout the pulse sequence, as well as the receiver phase, can then be adjusted accordingly.[44,54] Any single-spin rotation about $z$ can be realized easily by simply changing the phase of the subsequent pulses. Alternatively, $z$-rotations can be implemented using resonance offsets or composite pulses.[40]

Two strategies exist for implementing CNOT gates (both assume first-order spectra). If all the spins are mutually coupled, CNOTs can be realized via line-selective pulses, which invert specific lines within a multiplet.[12] In practice, it is usually more convenient to use pulse sequences such as the one in Figure 2.[10,12] For molecules with several coupled spins, the sequence of Figure 2 must be expanded with extra pulses to refocus the undesired J-couplings; systematic methods exist to design good refocusing schemes.[47,55,56] A CNOT between two uncoupled spins can be realized by swapping qubit states.[57,58] For example, for a CNOT between two spins $a$ and $c$ that are not mutually coupled but which are both coupled to a third spin $b$, the procedure is as follows: apply a pulse sequence that swaps the state of $a$ and $b$ (via $\mathrm{CNOT}_{ab}\ \mathrm{CNOT}_{ba}\ \mathrm{CNOT}_{ab}$), then perform a CNOT between $b$ and $c$, and then swap $a$ and $b$ again. The net result is $\mathrm{CNOT}_{ac}$; spin $b$ is unaffected. It is thus not necessary that all spins be pairwise coupled as long as the network of couplings includes all $n$ spins.

An alternative to imposing the correct evolution on all the spins at all times, both for RF pulses and for CNOT-type gates, is to allow erroneous evolutions that will later be reversed. Such techniques have been highly successful in certain standard pulse sequences (*see* ***Decoupling Methods***), but are much harder to develop for the nonintuitive and nontransparent QC pulse sequences. Nevertheless, it has been shown experimentally that a large degree of cancellation of erroneous evolutions is possible even in QC experiments: about 300 2-qubit gates involving over 1350 RF pulses have been successfully concatenated.[54] A general methodology designed to take advantage of this possibility has yet to be developed.

From this discussion, it is clear that *the selection of a suitable molecule is crucial for NMRQC.*

The desired properties are (i) sufficiently large chemical shifts for good addressability; (ii) large coupling constants, while maintaining first-order spectra, for fast 2-qubit gates (or a coupling network that matches the pattern of connectivities needed for the algorithm); and (iii) sufficiently long $T_2$s and $T_1$s in order to allow time to execute many logic gates. Furthermore, high-$\gamma$ nuclei are desirable for good sensitivity. More mundane but equally important requirements are that the molecule be stable, synthesizable, soluble, and safe.

### 3.3 State Initialization

Apart from experiments designed to produce nonthermal spin polarizations, setting up a proper initial state for the nuclear spins is a concept worth revisiting for NMR spectroscopists. Since this is a crucial step in quantum computing, this entire section is devoted to state initialization.

Most of the currently known quantum algorithms require a *pure* initial state, for example, a set of fully polarized spins, in the state $|00\ldots0\rangle$. However, nuclear spins in thermal equilibrium at room temperature are in an almost fully random state: for typical magnetic field strengths, the ground ($|0\rangle$) and excited ($|1\rangle$) state probabilities differ by only about 1 part in $10^5$. The spins are then said to be in a *mixed* (nonpure) state. The polarization could be increased using hyperpolarization techniques (*see* ***Sensitivity Enhancement Utilizing Parahydrogen***; ***Dynamic Nuclear Polarization: Applications to Liquid-State NMR Spectroscopy***) but with few exceptions, the state of the art is still very far from cooling nuclear spins into the ground state.

The conceptional breakthrough that made NMR QC possible at room temperature was the concept of *effective pure* or *pseudopure* states[10–12]: *effective pure states are mixed states that produce the same signal as a pure state to within a scaling factor.*

The signature of an effective pure state for $n$ spins is that all but one of the $2^n$ populations are equal, and that no coherences are present. The density matrix then consists of an identity component and a pure-state component. The identity density matrix is not observable in NMR, and it does not transform under unitary evolutions ($UIU^\dagger = I$). Thus the visible signal is derived exclusively from the one distinct population, which corresponds to a pure state. In product operator notation,[59] the effective pure ground state is proportional to $\hat{I}_z + \hat{S}_z + 2\hat{I}_z\hat{S}_z$ for two spins, to $\hat{I}_z + \hat{S}_z + \hat{R}_z + 2\hat{I}_z\hat{S}_z + 2\hat{I}_z\hat{R}_z + 2\hat{S}_z\hat{R}_z + 4\hat{I}_z\hat{S}_z\hat{R}_z$ for three spins, and so forth. Figure 5 shows how the effective pure state preparation is manifest in the spectrum of one of five coupled spins. A characteristic of effective pure (basis) states is that only one line survives in each multiplet.

Several methods have been proposed for preparing effective pure states starting from thermal equilibrium, including

1. *Logical labeling*[10,38] consists of applying a pulse sequence that rearranges the thermal populations such that a subset of the spins is in an effective pure state, conditioned upon the state of the remaining spins. Then the computation is carried out within this embedded subsystem.[60] For example, the Boltzman populations for the states $\{|000\rangle, |001\rangle, |010\rangle, |011\rangle, |100\rangle, |101\rangle, |110\rangle, |111\rangle\}$ for a homonuclear three-spin system deviate from the uniform background by $\{3a, a, a, -a, a, -a, -a, -3a\}$ respectively, where $a = \frac{1}{2^3}\frac{\hbar\omega}{2k_BT} \ll 1$. After rearranging the populations for the eight spin states as $\{3a, -a, -a, -a, a, a, a, -3a\}$, the last 2 qubits are in an effective pure state conditioned upon the first qubit being $|0\rangle$. As the total number of qubits $n$ in the molecule increases, the relative fraction of effective pure qubits goes to 1, but the preparation sequence becomes complex quite rapidly for large $n$ and the signal strength scales as $n/2^n$.

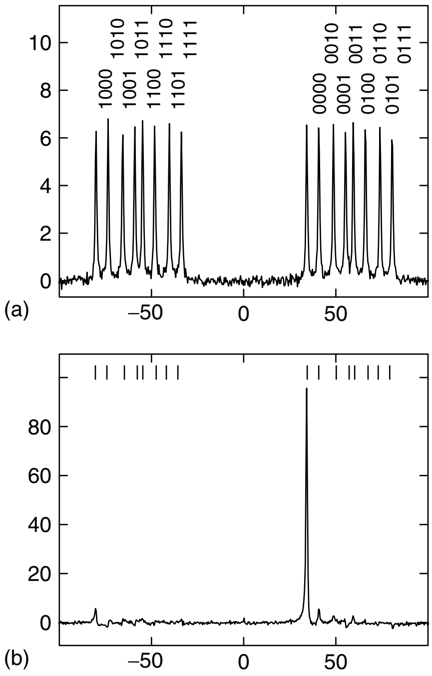

**Figure 5** (a) Spectrum of pentafluorobutadienyl cyclopentadienyldicarbonyliron complex in thermal equilibrium. (b) The same spectrum after preparing an effective pure state $|00000\rangle$. (Reproduced from Ref. 45. © American Physical Society, 2000.)

2. *Temporal averaging*[61] is similar to phase cycling (*see* ***Phase Cycling***), since it consists of adding up the spectra of multiple experiments. However, instead of changing just the phase of some pulses, each experiment starts off with a different state preparation sequence that permutes the populations. For two heteronuclear spins, adding together three experiments that yield respective population deviations $\{a, b, -b, -a\}$, $\{a, -b, -a, b\}$ and $\{a, -a, b, -b\}$ is equivalent to performing an experiment with population deviations $\{3a, -a, -a, -a\}$. For arbitary $n$, at least $(2^n - 1)/n$ experiments are needed,[45] since the effective pure state is made up of $2^n - 1$ product operator terms and the starting state, thermal equilibrium, contains $n$ terms.
3. *Spatial averaging*[12] uses a pulse sequence containing magnetic field gradients (*see* ***Field Gradients and Their Application***) to equalize all the populations but the ground-state population. Only one experiment is involved, but the preparation sequence quickly becomes unwieldy for large spin systems and the signal strength decreases exponentially with $n$.

   To date, temporal and spatial averaging have been the most popular choices for preparing effective pure states. Several hybrid schemes[44,61] have also been developed that trade off complexity of the preparation steps for the number of experiments. Nonetheless, all these state preparation schemes have in common that *creating effective pure states incurs an exponential cost either in the signal strength or in the number of experiments involved.*

Such an exponential overhead is of course not acceptable for quantum computations. The reason for this cost is that effective state preparation techniques simply select out the signal from the ground-state population present in thermal equilibrium, without enhancing it, and the fraction of the molecules in the ground state is proportional to $n/2^n$.

This exponential loss of sensitivity can, in principle, be avoided by using pure states instead of pseudopure states. This requires an increase of the spin polarization by many orders of magnitude. Possible schemes include optical pumping (*see* ***Optically Enhanced Magnetic Resonance***), dynamic nuclear polarization (*see* ***Dynamic Nuclear Polarization: Applications to Liquid-State NMR Spectroscopy***) and the use of para-hydrogen (*see* ***Sensitivity Enhancement Utilizing Parahydrogen***), possibly in combination with "algorithmic spin cooling."[62] Furthermore, not all quantum algorithms require pure input states.[63–65] Working directly with mixed states eliminates the sensitivity reduction and the complicated preparation stages associated with pseudopure input states. Depending on the algorithm, this may have to be compensated by an exponential loss of sensitivity in the output state.[64] Combined with the difficulty of addressing many qubits in a single molecule, it thus appears that liquid-state NMR may not offer a path to scalable QC.

### 3.4 Read Out

Traditionally in NMR spectroscopy, the signal from only one nuclear species is recorded. In QC, however, the concept of a single observe channel and one or more decoupler channels does not apply: *the output of a QC is the final state of one or several spins. The final states of each of the output spins must thus be read out.*

If each of the output spins ends up in $|0\rangle$ or $|1\rangle$ (or in reality in the effective pure state corresponding to $|0\rangle$ or $|1\rangle$), the answer can be read out directly by applying a pulse that rotates the spins from $\pm z$ to $\pm x$. With properly referenced receiver phase settings, the spectrum for each output spin then consists of either absorption or emission lines, indicating whether the output value of the corresponding bit is "0" or "1" (Figure 6).

If the output state is a superposition state, the situation is a bit more complicated. For a single (as opposed to an ensemble) quantum computer subject to a "hard" measurement (assumed in the section "Quantum Parallelism"), the superposition "collapses" to one of the terms in the superposition, with probabilities given by the square of the amplitude of each term. In contrast, *measurements in NMR give a (bitwise) ensemble-averaged readout.*

The output state of equation (9) serves as an example: half of the molecules in the ensemble collapse to $|0\rangle(|000\rangle)$, while the other half collapses to $|4\rangle(|100\rangle)$. In other words, spins 2 and 3 always end up in "0" so their spectral lines are absorptive; in contrast, the signal of spin 1 averages to 0 because there are equally many molecules in which spin 1 ends up in "0" as in "1." It is not clear that such bitwise averages of probabilistic output states are generally sufficient to solve the problem of interest. For Shor's period-finding algorithm, this problem can be circumvented[10] by performing the classical postprocessing steps (see the section "Quantum Algorithms") on the quantum computer using some ancillae qubits – any classical computation can also be done on a quantum computer.[5] In this way, the output state becomes the period $r$ for all the molecules in the ensemble (as opposed to an average over all the multiples of $N/r$), and the measurement result becomes deterministic.

Instead of recording the signal of each of the output spins, it is sometimes possible to use the extra information provided by the line splittings due to J couplings to derive the output state of several qubits from the spectrum of a single spin. Since each of the lines in the multiplet can be identified with specific states of the other spins (as in Figure 5), the presence or absence of each line in the multiplet gives information about the state of the other spins (Figure 6).

Finally, while the spectra of a few select spins suffice to obtain the answer to a computation, the full density matrix conveys much more information. This extra information can be used to expose the presence of errors, such as multiple-quantum coherences not visible in the single output spectra and furthermore is a useful tool for debugging pulse sequences. The procedure for reconstructing the density matrix is called *quantum state tomography*.[63,67,68] It consists of repeating the computation multiple times, each time looking at the final state of the spins after applying different sets of readout pulses that rotate different elements of the density matrix to observable positions. However, since this procedure involves on the order of $4^n$ experiments, it is practical only for experiments involving a small number of spins.

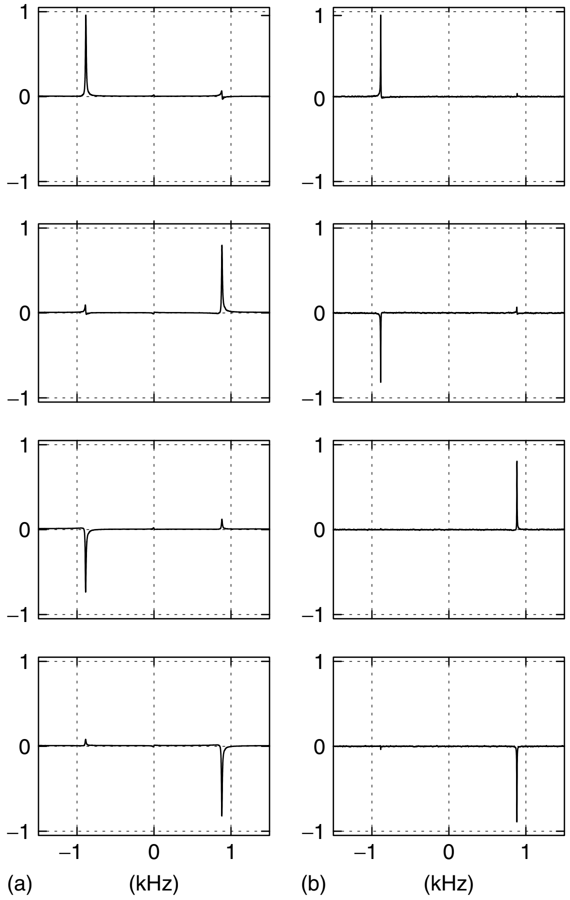

**Figure 6** Output spectra of the proton (a) and carbon (b) spins of $^{13}CHCl_3$ (dissolved in a liquid-crystal solution) for four different executions of Grover's search algorithm. Only the real part of the spectra is shown, and frequencies are relative to $\nu_H$ and $\nu_C$. A positive or negative line in the spectrum indicates that the corresponding spin was in $|0\rangle$ or $|1\rangle$ before the readout pulse. Furthermore, the position of the line in the spectrum of one spin also reveals the state of the other spin. For example, if the $^1H$ line is at $\nu_H - J_{CH}/2$, the $^{13}C$ spin is in $|0\rangle$; a $^1H$ line at $\nu_H + J_{CH}/2$ indicates the $^{13}C$ spin is in $|1\rangle$. Thus, the state of the 2 qubits for each of the four cases is (from top to bottom) $|00\rangle$, $|01\rangle$, $|10\rangle$, and $|11\rangle$. (Reproduced from Ref. 66. © American Institute of Physics, 1999.)

### 3.5 State of the Art and Outlook

To date, only very simple demonstrations of quantum algorithms have been carried out using liquid-state NMR techniques. Variations of Grover's algorithm have been demonstrated with 2[68–70] and 3 qubits,[54] the Deutsch–Jozsa algorithm with 2,[67,71] 3,[72] and 5[73] qubits, the period-finding algorithm with 5 qubits[45] and Shors algorithm using a 7-qubit molecule.[24] Nuclear spin systems with up to 12 qubits[74] have been used for demonstrating all the basic building blocks of quantum computers, including quantum simulations,[21,75,76] and quantum error detection[77] and correction.[78] While these experiments indeed demonstrate the principles of quantum information processing, they all involve far fewer qubits than would be needed to solve a problem beyond the reach of classical machines.

Despite the rapid progress in recent years, scaling liquid-state NMRQC to tens or hundreds of qubits may be impractical for several reasons, although none of them appear fundamental. In particular, as the number of qubits increases, (i) the strength of the signal selected with effective pure state techniques decreases exponentially[10,11,79]; (ii) the chemical shift separations unavoidably become smaller; (iii) J-couplings become smaller or even unresolved. While these obstacles are not fundamental, the solutions make the pulse sequences much longer. This would require increasingly longer $T_2$s and $T_1$s for larger molecules, while in practice the $T_2$s and $T_1$s tend to become shorter (*see* ***Relaxation: An Introduction***).

NMRQC has also brought up new theoretical issues.

1. Since only ensemble-averaged results are available because of the large number of molecules in a sample tube, some information is lost that would be available in an idealized quantum computer such as a single molecule at 0 K. For the known quantum algorithms, this information can be retrieved by performing classical postprocessing steps on the quantum computer (see the section “Quantum Algorithms”).
2. Since the density matrix of nuclear spins at room temperature is very close to the identity matrix, it is not possible to produce genuinely entangled states between the nuclear spins in small thermally polarized molecules in liquid solution.[80] This observation has sparked a stimulating debate about the “quantumness” of NMR, because it implies that each of the states produced in NMRQC experiments so far is classical. However, all attempts to describe the dynamics of a set of coupled spins by an efficient classical model have been unsuccessful. It is thus conjectured that even though the states are classical, the dynamics of the spins is truly quantum mechanical,[81] a proposition that will appear obvious to most NMR spectroscopists. In fact, it has also been the starting point of this introduction to NMR quantum computing.

## 4 SUMMARY AND CONCLUSIONS

In many respects, liquid-state NMR provides an ideal test bed for elementary QCs. The degree of control over the evolution of multiple coupled qubits – the result of 50 years of technology development – the long relaxation times of nuclear spins and a set of new insights[10,11] made it possible to perform certain computations in *fewer steps* than is possible using any classical machine. This is in itself a remarkable achievement.

It is unlikely that liquid-state NMR could ever be used to solve problems *faster* than any classical machine, but it has already inspired many other NMR-based proposals for quantum computing. Liquid-crystal solvents have been used to partly reintroduce dipole–dipole couplings (*see* ***Liquid Crystals: General Considerations***) to speed up the gate time and increase the number of gates possible within the coherence time.[20,52,66] New molecular architectures based on liquid-crystal solutions are now being investigated. Solid-state NMR[82] near 0 K not only circumvents the state initialization problem but also poses new questions in terms of addressability and coherence times (*see* ***Internal Spin Interactions and Rotations in Solids***). Several approaches to solve these issues have been proposed.[83,84] Another proposal that received much attention consists of doing NMR on impurity atoms placed in a linear or two-dimensional array, with chemical shifts and couplings controlled by electrodes placed on top of and in between the impurity atoms.[85]

Furthermore, there is a plethora of very different experimental approaches to building quantum computers.[6,86] The first system that was considered for implementing quantum computing and the second that was experimentally implemented are trapped atomic ions.[87] Quantum registers based on trapped ions have been demonstrated with up to 8 qubits.[88] Photons are among the most attractive quantum objects, in particular since they are also the qubits of choice for quantum communication. The main obstacle for using them in quantum computing is that they do not naturally interact with each other. Nevertheless, it is possible to create effective interactions, by combining quantum measurements with feed-forward controls.[32] In the long run, the most scalable approaches may be those based on solid-state technology, such as electron spins in quantum dots[89] or superconducting microstructures.[90,91]

It is clear that none of these proposals will be easy to implement – they all require substantial and innovative development of technology. The success of any approach will depend on the ratio of the coherence time to the gate duration, i.e., how many gates can be completed within the coherence time, and on the achievable degree of quantum control.

Many of the problems in quantum control are similar for different experimental systems, and we see that many of the ideas, concepts, and solutions developed in liquid-state NMR experiments are now adopted in a variety of other quantum computer implementations. In addition, we hope that some of the techniques developed within the context of QC may find more general application in NMR.

The possible payoff for successful quantum computing is tremendous: to solve problems beyond the reach of any classical computer. It is not clear at this point whether quantum computers will fulfill this promise, but, in any case, quantum computing has already provided an exciting new perspective on NMR and, more broadly, on the connection between physics, information, and computation.

## 5 REFERENCES

1. A. G. Anderson, R. L. Garwin, E. L. Hahn, J. W. Horton, G. L. Tucker, and R. M. Walker, *J. Appl. Phys.*, 1955, **26**, 1324.
2. R. Landauer, *Phys. Today*, 1991, **44**, (5), 22.
3. C. Bennett and D. P. DiVincenzo, *Nature*, 2000, **404**, 247.
4. A. Ekert and R. Jozsa, *Rev. Mod. Phys.*, 1996, **68**, 733.
5. M. A. Nielsen and I. L. Chuang, *‘Quantum Computation and Quantum Information’*, Cambridge University Press: Cambridge, England, 2000.
6. J. Stolze and D. Suter, *‘Quantum Computing: A Short Course from Theory to Experiment’*, Wiley-VCH: Berlin, 2008.

7. J. A. Jones, *Fortschr. Phys.*, 2000, **48**, 909.
8. L. M. K. Vandersypen and I. L. Chuang, *Rev. Mod. Phys.*, 2004, **76**, 1037.
9. N. Gershenfeld and I. L. Chuang, *Sci. Am.*, 1998, **278**(6), 66–66.
10. N. Gershenfeld and I. L. Chuang, *Science*, 1997, **275**, 350.
11. D. G. Cory, M. D. Price, and T. F. Havel, *Phys. D*, 1998, **81**, 2152.
12. D. G. Cory, A. F. Fahmy, and T. F. Havel, *Proc. Natl. Acad. Sci. U.S.A.*, 1997, **94**, 1634.
13. R. P. Feynman, *Int. J. Theor. Phys.*, 1982, **21**, 467.
14. D. Deutsch, *Proc. R. Soc. Lond. A*, 1885, **400**, 97.
15. P. Shor, In *Proceedings of the 35th Annual Symposium on the Foundations of Computer Science*, IEEE Computer Society Press, Los Alamitos, CA, 1994, 124.
16. D. Deutsch and R. Jozsa, *Proc. R. Soc. Lond. A*, 1992, **439**, 553.
17. L. Grover, *Phys. Rev. Lett.*, 1997, **79**, 4709.
18. S. Lloyd, *Science*, 1996, **273**, 1073.
19. I. Buluta and F. Nori, *Nature*, 2009, **326**, 108.
20. J. Zhang, F. M. Cucchietti, C. M. Chandrashekar, M. Laforest, C. A. Ryan, M. Ditty, A. Hubbard, J. K. Gamble, and R. Laflamme, *Phys. Rev. A*, 2009, **79**, 012305.
21. X.-H. Peng and D. Suter, *Front. Phys. China*, 2010, **5**, 1.
22. M. Greiner, O. Mandel, T. Esslinger, T. W. Hänsch, and I. Bloch, *Nature*, 2002, **415**, 39.
23. X. Peng, J. Zhang, J. Du, and D. Suter, *Phys. Rev. Lett.*, 2009, **103**, 140501.
24. L. M. K. Vandersypen, M. Steffen, G. Breyta, C. S. Yannoni, M. H. Sherwood, and I. L. Chuang, *Nature*, 2001, **414**, 883.
25. P. W. Shor, *Phys. Rev. A*, 1995, **52**, 2493.
26. A. Steane, *Phys. Rev. Lett.*, 1996, **77**, 793.
27. H. G. Krojanski and D. Suter, *Phys. Rev. Lett.*, 2004, **93**, 090501.
28. H. G. Krojanski and D. Suter, *Phys. Rev. A*, 2006, **74**, 062319.
29. J. Preskill, *Proc. Roy. Soc. Lond. A*, 1998, **454**, 385.
30. E. Knill, *Nature*, 2005, **434**, 39.
31. R. Raussendorf and H. J. Briegel, *Phys. Rev. Lett.*, 2001, **86**, 5188.
32. P. Kok, *et al. Rev. Mod. Phys.*, 2007, **79**, 135.
33. E. Farhi, *et al. Science*, 2001, **282**, 472.
34. X. Peng, *et al. Phys. Rev. Lett.*, 2008, **101**, 220405.
35. A. Barenco, *et al. Phys. Rev. A*, 1995, **52**, 3457.
36. M. D. Price, S. S. Somaroo, A. E. Dunlop, T. F. Havel, and D. G. Cory, *Phys. Rev. A*, 1999, **60**, 2777.
37. J. A. Jones, R. H. Hansen, and M. Mosca, *J. Magn. Reson.*, 1998, **135**, 353.
38. L. M. K. Vandersypen, C. S. Yannoni, M. H. Sherwood, and I. L. Chuang, *Phys. Rev. Lett.*, 1999, **83**, 3085.
39. N. Linden, H. Barjat, R. J. Carbajo, and R. Freeman, *Chem. Phys. Lett.*, 1999, **305**, 28.
40. R. Freeman, *'Spin Choreography'*, Spektrum: Oxford, England, 1997.
41. M. H. Levitt, *Prog. NMR Spectrosc.*, 1986, **18**, 61.
42. H. K. Cummins and J. A. Jones, *New J. Phys.*, 2000, **2**, 6.1.
43. L. Emsley and G. Bodenhausen, *Chem. Phys. Lett.*, 1990, **168**, 297.
44. E. Knill, R. Laflamme, R. Martinez, and C.-H. Tseng, *Nature*, 2000, **404**, 368.
45. L. Vandersypen, M. Steffen, G. Breyta, C. Yannoni, R. Cleve, and I. Chuang, *Phys. Rev. Lett.*, 2000, **85**, 5452.
46. E. Kupče and R. Freeman, *J. Magn. Reson. A*, 1995, **112**, 261.
47. N. Linden, E. Kupče, and R. Freeman, *Chem. Phys. Lett.*, 1999, **311**, 321.
48. M. Steffen, L. M. K. Vandersypen, and I. L. Chuang, *J. Magn. Reson.*, 2000, **146**, 369.
49. E. Kupče, J. M. Nuzillard, V. S. Dimitrov, and R. Freeman, *J. Magn. Reson. A*, 1994, **107**, 246.
50. C. A. Ryan, M. Laforest, and R. Laflamme, *New J. Phys.*, 2009, **11**, 013034.
51. N. Khaneja, T. Reiss, C. Kehlet, T. Schulte-Herbrüggen, and S. J. Glaser, *J. Magn. Reson.*, 2005, **172**, 296.
52. T. S. Mahesh and D. Suter, *Phys. Rev. A*, 2006, **74**, 062312.
53. S. L. Patt, *J. Magn. Reson.*, 1992, **96**, 94.
54. L. M. K. Vandersypen, M. Steffen, M. H. Sherwood, C. S. Yannoni, G. Breyta, and I. L. Chuang, *Appl. Phys. Lett.*, 2000, **76**, 648.
55. D. W. Leung, I. L. Chuang, F. Yamaguchi, and Y. Yamamoto, *Phys. Rev. A*, 2000, **61**, 042310.
56. J. A. Jones and E. Knill, *J. Magn. Reson.*, 1999, **141**, 322.
57. S. Lloyd, *Science*, 1993, **261**, 1569.
58. D. Collins, *et al. Phys. Rev. A*, 2000, **62**, 022304.
59. O. W. Sorenson, *et al. Prog. NMR Spectrosc.*, 1983, **16**, 163.
60. D. Suter, A. Pines, and M. Mehring, *Phys. Rev. Lett.*, 1986, **57**, 242.
61. E. Knill, I. L. Chuang, and R. Laflamme, *Phys. Rev. A*, 1998, **81**, 5672.
62. L. J. Schulman and U. Vazirani, *'Proceedings of the 31st Annual Symposium on Theory of Computer Science'*, Atlanta, GA, 1998, 322.
63. I. L. Chuang, L. M. K. Vandersypen, X. Zhou, D. W. Leung, and S. Lloyd, *Nature*, 1998, **393**, 143.
64. R. Brueschweiler, *Phys. Rev. Lett.*, 2000, **85**, 4815.
65. R. Stadelhofer, D. Suter, and W. Banzhaf, *Phys. Rev. A*, 2005, **71**, 032345.
66. C. S. Yannoni, *et al. Appl. Phys. Lett.*, 1999, **75**, 3563.
67. I. L. Chuang, N. Gershenfeld, M. Kubinec, and D. Leung, *Proc. R. Soc. Lond. A*, 1998, **454**, 447.
68. I. L. Chuang, N. Gershenfeld, and M. Kubinec, *Phys. Rev. Lett.*, 1998, **80**, 3408.
69. J. A. Jones, M. Mosca, and R. H. Hansen, *Nature*, 1998, **393**, 344.
70. J. A. Jones and M. Mosca, *Phys. Rev. Lett.*, 1999, **83**, 1050.
71. J. A. Jones and M. Mosca, *J. Chem. Phys.*, 1998, **109**, 1648.
72. N. Linden, H. Barjat, and R. Freeman, *Chem. Phys. Lett.*, 1998, **296**, 61.
73. R. Marx, A. F. Fahmy, J. M. Myers, W. Bermel, and S. J. Glaser, *Phys. Rev. A*, 2000, **62**, 012310.
74. C. Negrevergne, *et al. Phys. Rev. Lett.*, 2006, **96**, 170501.
75. S. Somaroo, C.-H. Tseng, T. F. Havel, R. Laflamme, and D. G. Cory, *Phys. Rev. Lett.*, 1999, **82**, 5381.
76. C.-H. Tseng, *et al. Phys. Rev. A*, 2000, **61**, 012302.
77. D. Leung, *et al. Phys. Rev. A*, 1999, **60**, 1924.
78. D. G. Cory, *et al. Phys. Rev. Lett.*, 1998, **81**, 2152.
79. W. S. Warren, *Science*, 1997, **277**, 1688.
80. S. L. Braunstein, *et al. Phys. Rev. Lett.*, 1999, **83**, 1054.
81. R. Schack and C. Caves, *Phys. Rev. A*, 1999, **60**, 4354.
82. J. Baugh, *et al. Phys. Rev. A*, 2006, **73**, 022305.
83. D. G. Cory, *et al. Fortschr. Phys.*, 2000, **48**, 875.
84. F. Yamaguchi and Y. Yamamoto, *Appl. Phys. A*, 1999, **68**, 1.
85. B. E. Kane, *Nature*, 1998, **393**, 133.
86. Special Issue, *Fortschr. Phys.*, 2000, **48**, 9.
87. H. Häffner, C. F. Roos, and R. Blatt, *Phys. Rep.*, 2008, **469**, 155.
88. H. Häffner, *et al. Nature*, 2005, **438**, 643.
89. R. Hanson, *et al. Rev. Mod. Phys.*, 2007, **79**, 1217.
90. Y. Makhlin, *et al. Rev. Mod. Phys.*, 2001, **73**, 357.
91. L. DiCarlo, *et al. Nature*, 2009, **460**, 240.

## Biographical Sketches

Lieven Vandersypen received a PhD in Electrical Engineering from Stanford University (2001) for work on experimental NMR quantum computation. Most of the experiments were performed at the IBM Almaden Research Center in San Jose, CA, under the supervision of I.L. Chuang. He then moved to TU Delft, where he is currently Antoni van Leeuwenhoek Professor and works on experimental quantum computation using electron spins in semiconductor quantum dots, as well as on nanoelectronic devices in graphene.

Prof. Vandersypen has received the 2008 Nicholas Kurti European Science prize and the 2008 IUPAP Young Scientist Prize for Semiconductor Physics, and is a member of the Young Academy of the Royal Netherlands Academy of Arts and Sciences.

Isaac Chuang is a Professor of Physics and a Professor of Electrical Engineering and Computer Science at the Massachusetts Institute of Technology. He is a pioneer in the field of quantum information science. His experimental realization of two, three, five, and seven quantum bit quantum computers using nuclear spins in molecules provided the first laboratory demonstrations of many important quantum algorithms, including Shor's quantum factoring algorithm. The error correction, algorithmic cooling, and entanglement manipulation techniques he developed provide new ways to obtain complete quantum control over light and matter, and lay a foundation for possible large-scale quantum information processing systems. Prof. Chuang came to MIT in 2000 from IBM, where he was a research staff member. He received his doctorate in Electrical Engineering from Stanford University, where he was a Hertz Foundation Fellow. Prof. Chuang also holds two bachelors and one masters degrees in Physics and Electrical Engineering from MIT, and was a postdoctoral fellow at Los Alamos National Laboratory and the University of California at Berkeley. He is the author, together with Michael Nielsen, of the textbook *Quantum Computation and Quantum Information*.

Dieter Suter received a PhD in Chemistry from ETH Zürich for work on magnetic resonance. He continued to work in NMR for two years at UC Berkeley and then moved to the laboratory of quantum optics of ETH Zürich, where he worked mostly with atomic vapors and optically detected magnetic resonance. Since 1995, he is professor for condensed matter physics at the Technical University of Dortmund. His current research includes magnetic resonance, laser spectroscopy, and quantum information processing.