

ARTICLE OPEN

Enhancing quantum control by bootstrapping a quantum processor of 12 qubits

Dawei Lu^{1,2}, Keren Li^{2,3}, Jun Li^{2,4}, Hemant Katiyar², Annie Jihyun Park^{2,5}, Guanru Feng², Tao Xin^{2,3}, Hang Li^{2,3}, Guilu Long³, Aharon Brodutch^{2,6}, Jonathan Baugh², Bei Zeng^{2,7} and Raymond Laflamme^{2,8}

Accurate and efficient control of quantum systems is one of the central challenges for quantum information processing. Current state-of-the-art experiments rarely go beyond 10 qubits and in most cases demonstrate only limited control. Here we demonstrate control of a 12-qubit system, and show that the system can be employed as a quantum processor to optimize its own control sequence by using measurement-based feedback control (MQFC). The final product is a control sequence for a complex 12-qubit task: preparation of a 12-coherent state. The control sequence is about 10% more accurate than the one generated by the standard (classical) technique, showing that MQFC can correct for unknown imperfections. Apart from demonstrating a high level of control over a relatively large system, our results show that even at the 12-qubit level, a quantum processor can be a useful lab instrument. As an extension of our work, we propose a method for combining the MQFC technique with a twirling protocol, to optimize the control sequence that produces a desired Clifford gate.

npj Quantum Information (2017)3:45 ; doi:10.1038/s41534-017-0045-z

INTRODUCTION

Quantum computers promise to outperform their classical counterparts in many applications.^{1–6} A primary obstacle in building large-scale quantum computers is the inadequacy of classical computers for the task of optimizing the experimental control field.⁷ Standard classical optimization algorithms are impractical in the long run since they have a running time that grows exponentially with the number of quantum bits (qubits).⁸ In theory, a complex quantum circuit can be decomposed into elementary gates that work on a restricted number of qubits (usually one or two) and should be readily implemented in experiment.⁹ In reality however, the control fields are never localized and the qubits interact and evolve even in the absence of the control fields. Consequently, the implementation of each elementary gate may require a control sequence that takes into account a subsystem involving many more than one or two qubits. Moreover, the number of elementary gates required for a quantum algorithm grows polynomially with the system size and the errors accumulate with each successive gate. Therefore, an effective and efficient way to optimize the control field and minimize errors is a key ingredient for scaling up quantum information processing devices.¹⁰

Here we consider the task of optimizing a control field that will drive the quantum system from a fixed input state ρ_i to a desired target state ρ_f . This problem is important in quantum information processing, as numerous tasks, such as algorithmic cooling in ensemble quantum computing,^{11,12} magic state preparation in fault-tolerant quantum computing¹³ and encoding in quantum key distribution,¹⁴ all rely on steering states regardless of the

propagator. The gradient ascent pulse engineering (GRAPE) algorithm¹⁵ is the current state-of-the-art algorithm to (classically) optimize the control field in quantum state engineering problems. It is widely used in NMR,¹⁶ electron spin resonance,¹⁷ nitrogen-vacancy centers in diamond,^{18,19} superconducting circuits,^{20,21} and ion traps.^{22,23} The GRAPE method exploits the gradient of a fidelity function to update the control field iteratively.

GRAPE has two major drawbacks that are indeed common to all classical optimization algorithms: its running time is exponential in the size of the n -qubit system, and its accuracy depends on the precision of experimentally obtained parameters describing the quantum system (e.g., the system Hamiltonian). Basically, it is a gradient-based iterative algorithm. At each iteration k , the algorithm computes the evolution of the system under the previous pulse, and produces a final state $\bar{\rho}$ and a fitness function $f = \text{tr}(\bar{\rho}\rho_f)$. It then computes the current gradient g for the use of updating the pulse. Classically, the computation involves the matrix exponential and multiplication in the 2^n -dimensional Hilbert space and hence takes an exponential (in the number of qubits n) amount of time. For instance, a cluster of 128 AMD Opteron 850 CPU (2.4 GHz) can only handle a problem size of about ten qubits using GRAPE.⁸

Recently, Li et al.²⁴ and later Rebenrost et al.²⁵ showed that a quantum processor can be used to calculate f and g efficiently. A technique called measurement-based quantum feedback control (MQFC) enables direct measurement of f and g (see Fig. 1), allowing the quantum processor to optimize its own pulses. MQFC addresses both the issues of scalability and control inaccuracies due to imperfect system characterization.^{26,27} Moreover, this

¹Department of Physics, Southern University of Science and Technology, Shenzhen 518055, China; ²Institute for Quantum Computing and Department of Physics and Astronomy, University of Waterloo, Waterloo N2L 3G1 ON, Canada; ³State Key Laboratory of Low-Dimensional Quantum Physics and Department of Physics, Tsinghua University, Beijing 100084, China; ⁴Beijing Computational Science Research Center, Beijing 100193, China; ⁵Max-Planck-Institut für Quantenoptik, D-85748 Garching, Germany; ⁶Department of Physics and Department of Electrical and Computer Engineering, Center for Quantum Information and Quantum Control, University of Toronto, Toronto M5S 3H6 ON, Canada; ⁷Department of Mathematics and Statistics, University of Guelph, Guelph N1G 2W1 ON, Canada and ⁸Perimeter Institute for Theoretical Physics, Waterloo N2L 2Y5 ON, Canada
Correspondence: Dawei Lu (ludw@sustc.edu.cn) or Bei Zeng (zengb@uoguelph.ca)

Received: 23 August 2017 Revised: 14 September 2017 Accepted: 18 September 2017

Published online: 23 October 2017

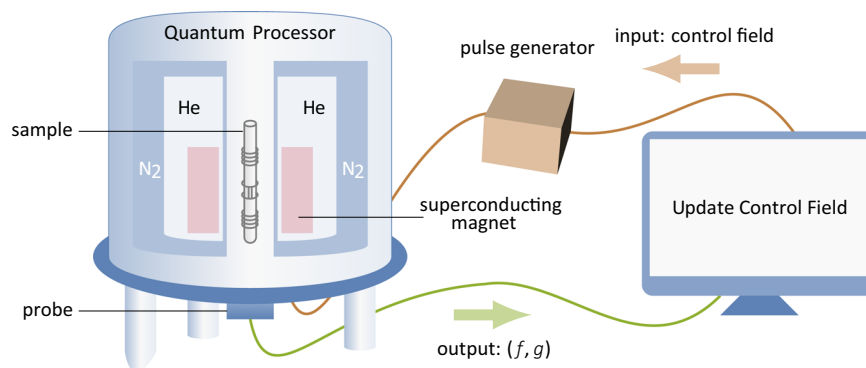


Fig. 1 MQFC process for optimizing a control field. Starting from an initial guess, a shaped pulse is created from the pulse generator and then applied to the sample. The fidelity function f of the control pulse and its gradient g are directly measured on the quantum processor, where g is used for updating the control field till that sufficiently high fidelity f has been achieved

technique is transferrable to any implementation in which control fields steer the system evolution and measurement in a standard basis is possible. In this work, we implement MQFC on a 12-qubit NMR quantum processor, and in particular demonstrate for the first time that MQFC enhances the control precision by about 10% due to its self-feedback property. Furthermore, by creating the 12-coherent state we demonstrate the capability of our quantum processor to function as a universal 12-qubit quantum processor with high-fidelity individual controls. This is also one of the largest quantum processors with individual-control to date.

RESULTS

In this paper, we refer to unnormalized deviation density matrices (without the identity term) as ‘states’, which is a standard convention in ensemble quantum computing. To distinguish from the Hamiltonian, we use capital X , Y , and Z to denote states and σ_x , σ_y and σ_z to denote Hamiltonians, while they both refer to the same set of Pauli matrices.

Quantum processor

In our NMR quantum processor, the liquid-state sample is per- ^{13}C labeled (1S,4S,5S)-7,7-dichloro-6-oxo-2-thiabicyclo[3.2.0]heptane-4-carboxylic acid dissolved in d6-acetone, which forms a 12-qubit register. The 12 qubits are denoted by nuclear spins C_1 to C_7 (^{13}C -labeled) as qubits 1 to 7, and H_1 to H_5 as qubits 8 to 12 in the molecule shown by Fig. 2a. When placed in a static z-magnetic field, it has a system Hamiltonian

$$\mathcal{H}_s = -\pi \sum_{i=1}^{12} \nu_0^i \sigma_z^i + \frac{\pi}{2} \sum_{i=1}^{12} \sum_{i=1}^{12} J_{ij} \sigma_z^i \sigma_z^j, \quad (1)$$

where ν_0^i is the Larmor frequency of the i th qubit, J_{ij} is the coupling between qubits i and j , and σ_z^i is the Pauli-z operator of the i th qubit. The values of these parameters can be found in Appendix C (See Supplementary information).

The control Hamiltonian is due to the transverse control field applied in the x - y plane, which is often digitized into M slices with slice length Δt . In each slice, there are four constant control parameters, leading to a control Hamiltonian in the form of

$$\mathcal{H}_c[m] = B_x^C[m] \sum_{i=1}^7 \sigma_x^i + B_y^C[m] \sum_{i=1}^7 \sigma_y^i + B_x^H[m] \sum_{j=8}^{12} \sigma_x^j + B_y^H[m] \sum_{j=8}^{12} \sigma_y^j, \quad (2)$$

where, for example, $B_x^C[m]$ means the x -component of the m th slice of control field in the ^{13}C channel.

The dynamics of the NMR system is governed by \mathcal{H}_s and \mathcal{H}_c simultaneously, with the propagator

$$U_1^M = U_M U_{M-1} \cdots U_1, \quad (3)$$

where

$$U_m = e^{-i(\mathcal{H}_s + \mathcal{H}_c[m])\Delta t}. \quad (4)$$

The essence of NMR quantum information processing is to optimize a control field, i.e., find a sequence of $B_{x,y}[m]$, such that one can precisely realize a quantum gate or drive the system to a target state according to Eq. (3).

Fundamentals of the GRAPE algorithm

To implement a particular target gate or state we need to find an optimal $B_{x,y}[m]$. One of the most prominent optimization algorithms to date is the GRAPE algorithm¹⁵ which was developed for the design of optimal control pulses in NMR spectroscopy. Here, we explain the basic principle of GRAPE by considering the problem of state engineering in the absence of relaxation.

Suppose the initial state of the spin system is ρ_i , and the target output state is ρ_f . After applying a M -slice trial control pulse, the system will evolve to

$$\tilde{\rho} = U_1^M(\rho_i) = U_1^M \rho_i U_1^{M\dagger}. \quad (5)$$

The fitness function defined as $f = \text{tr}(\rho_f \tilde{\rho})$ serves as a metric for the control fidelity, with the form

$$f = \text{tr}(\rho_f \tilde{\rho}) = \text{tr}(U_1^M(\rho_i) \cdot \rho_f). \quad (6)$$

Obviously, f is a function of $2M$ variables, and to find its optimum we calculate the gradient function to the first order

$$g_{x,y}[m] = \frac{\partial f}{\partial B_{x,y}[m]} \approx \sum_{k=1}^n \text{tr} \left(-i\Delta t \cdot U_{m+1}^M \left[\sigma_{x,y}^k, U_1^m(\rho_i) \right] U_{m+1}^{M\dagger} \cdot \rho_f \right), \quad (7)$$

where $[\sigma_{x,y}^k, U_1^m(\rho_i)]$ is the commutator between $\sigma_{x,y}^k$ and $U_1^m(\rho_i)$. We may increase the fitness function f by using the gradient iteration rule

$$B_{x,y}[m] \leftarrow B_{x,y}[m] + \epsilon \cdot g_{x,y}[m], \quad (8)$$

where ϵ is a suitably chosen step size.

The GRAPE algorithm proceeds as follows on a classical computer:

1. start from an initial guess control $B_{x,y}[m]$;
2. calculate $\tilde{\rho}$ according to Eq. (5);
3. evaluate fitness function $f = \text{tr}(\rho_f \tilde{\rho})$;
4. if f does not reach our preset value, evaluate gradient function g according to Eq. (7);

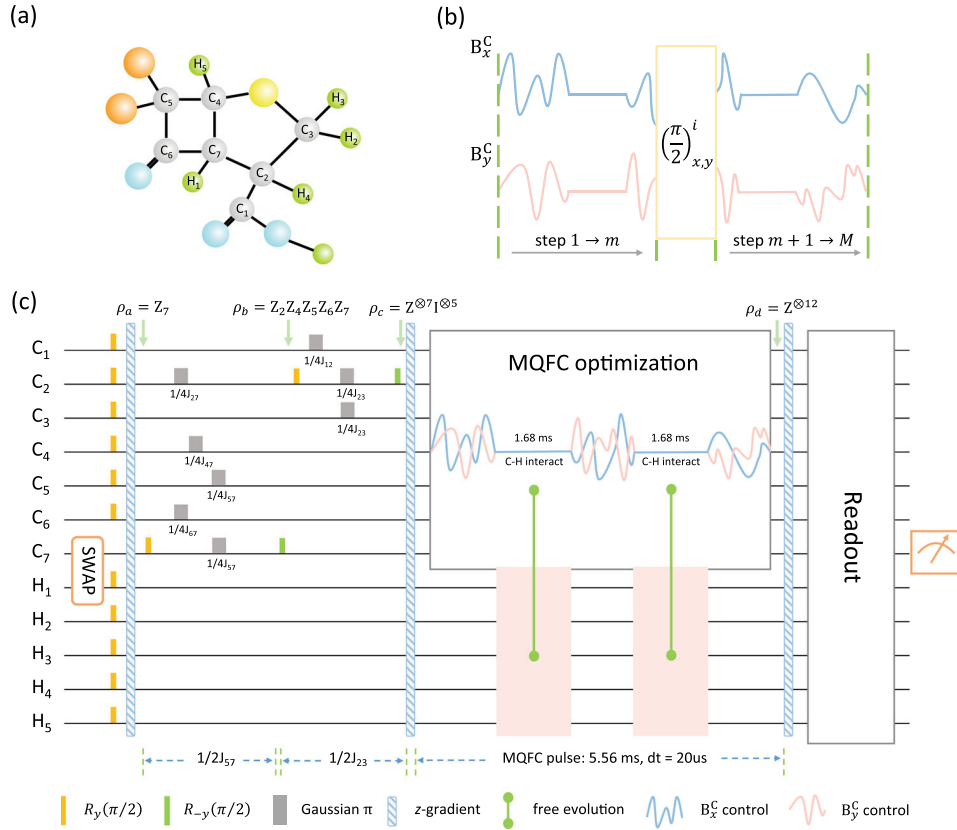


Fig. 2 MQFC scheme in creating 12-coherence. **a** Molecular structure of the 12-qubit quantum processor. **b** Schematic of measuring the m -th step gradient $g_{x,y}[m]$. A $\pi/2$ rotation about $x(y)$ -axis for qubit i is inserted between the m -th and $(m+1)$ -th slices. **c** Quantum circuit that evolves the system from the thermal equilibrium to 12-coherence, where MQFC is applied on 7-coherence $Z^{\otimes 7} I^{\otimes 5}$

5. update control variables according to Eq. (8), then go to step 2.

MQFC optimization

The GRAPE algorithm requires the calculation of U_1^M , i.e., the dynamics of the system. This step is inefficient on a classical computer when the size of the system is large. In contrast, the scheme of MQFC optimization provides an alternative way which enables direct measurement of f and g in the experimental manner, or explicitly, via the quantum evolution and measurement of the quantum processor.

Without loss of generality, let us discuss the scenario of ensemble quantum computing. e.g., NMR quantum computing, where the state is usually written as a traceless deviation density matrix and a single-shot measurement is sufficient to get the expected value of an observable. For other systems that use the computational basis or projective measurement, the following procedure needs to be slightly modified and more repetitions may be required to get the estimate of f and g .

Measuring f is straightforward. For an n -qubit system, the total number of elements in the Pauli basis is $4^n - 1$ (without the identity term). If the target state ρ_f has some decomposition, say, $\rho_f = \sum_{\gamma=1}^{\mathcal{G}} x_{\gamma} P_{\gamma}$ with respect to the Pauli basis, then the fitness function is

$$f = \text{tr}(\hat{\rho} \rho_f) = \sum_{\gamma=1}^{\mathcal{G}} x_{\gamma} \text{tr}(\hat{\rho} P_{\gamma}). \quad (9)$$

Here, $1 \leq \mathcal{G} \leq 4^n$ denotes the number of nonzero components, P_{γ} is the γ -th element of the Pauli basis, and x_{γ} is its corresponding coefficient.

Therefore, \mathcal{G} experiments are required to estimate f . In the γ -th experiment, we just need to apply the control field to the initial state ρ , and measure the expectation value $\langle P_{\gamma} \rangle$ of $\hat{\rho}$. For a generic ρ_f that contains all $\mathcal{G} = 4^n - 1$ Pauli terms, measuring f in experiment is equivalent to carrying out full state tomography, and is thus inefficient. However, many tasks require the creation of a simple target state where \mathcal{G} is quite small. For instance, if we aim to prepare the 12-coherent state $\rho_f = Z^{\otimes 12}$, one measurement is sufficient to obtain f .

Measuring g requires us to realize the commutator $[\sigma_{x,y}^k, \cdot]$ inside Eq. (7). In fact,²⁴

$$[\sigma_{x,y}^k, \rho] = i(\mathcal{R}_{x,y}^k(\rho) - \overline{\mathcal{R}}_{x,y}^k(\rho)), \quad (10)$$

in which $\mathcal{R}_{x,y}^k$ and $\overline{\mathcal{R}}_{x,y}^k$ mean a $\pi/2$ rotation and $-\pi/2$ about x or y axis on the k -th qubit, respectively. By substituting Eq. (10) into Eq. (7), we get

$$g_{x,y}[m] = \Delta t \sum_{k=1}^n \text{tr} \left\{ \left(U_{m+1}^M \mathcal{R}_{x,y}^k U_1^m \right) (\rho_i) \cdot \rho_f \right\} - \Delta t \sum_{k=1}^n \text{tr} \left\{ \left(U_{m+1}^M \overline{\mathcal{R}}_{x,y}^k U_1^m \right) (\rho_i) \cdot \rho_f \right\}. \quad (11)$$

The terms on the right-hand side are very similar to the measurement of f in Eq. (6), and the only difference is the local $\pm\pi/2$ pulse inserted between slices m and $m+1$. Explicitly, the m -th component of $g_{x,y}$ is a weighted sum of $4n\mathcal{G}$ measurement quantities, where 4 comes from the $\pm\pi/2$ pulses about the x and y axes, n from the sum over all the qubits, and \mathcal{G} from the measurement of f . In each experiment, compared to the way of measuring f , we just need to insert a local $\pi/2$ pulse after the m -th

slice evolution. Provided that all the qubits are well individually addressed, high fidelities are attainable in implementing these local $\pi/2$ rotations.

In summary, we need $4nGM$ experiments in total to perform the gradient measurement, which is linear in the number of qubits.

Experimental MQFC optimization

Now we turn to the experiment where the MQFC optimization is used to create the 12-coherent state in the 12-qubit quantum processor. First, let us clarify that all other pulses except the MQFC pulse throughout our experiments are local rotations, which are generated from a subsystem-based gradient ascent pulse engineering (SSGRAPE) approach.¹⁶ It is a technical improvement of the original GRAPE for our particular implementation, but does not address its poor scalability issue (see Appendix D, See [Supplementary information](#)). What makes the MQFC scheme remarkable is that, it does not involve the computationally expensive classical simulation of the 2^{12} -dimensional quantum dynamics in the course of optimization.

For our optimization task, GRAPE is a powerful tool, but handling 12 qubits is near the limit of capability for a typical laptop computer. In contrast, MQFC is capable of overcoming this difficulty in certain cases. Taking our experiment as an example, MQFC is able to solve the problem of finding a control field that evolves single-coherence $Z^{\otimes 11}$ into 12-coherence $Z^{\otimes 12}$ in a time that scales linearly with the number of qubits. The entire experimental procedure is depicted in Fig. 2c, with a step-by-step description in Appendix E (See [Supplementary information](#)).

First, we prepare 7-coherence $Z^{\otimes 7}|^{\otimes 5}$ on the seven ^{13}C spins, using the sequence in Fig. 2c before the *MQFC optimization* box. This procedure, benchmarked in our previous work,²⁸ is mainly done with the aid of SSGRAPE. Subsequently, we create $Z^{\otimes 12}$ via MQFC on the quantum processor, which is the main focus of this work. We attempt to optimize a control field, namely a shaped radio frequency (r.f.) pulse, to evolve the system from the input $\rho_i = Z^{\otimes 7}|^{\otimes 5}$ to the output $\rho_f = Z^{\otimes 12}$. Our control field, as shown in the *MQFC optimization* box, is comprised of three sub-pulses to realize local rotations, and two free evolutions to let ^{13}C qubits interact with ^1H qubits for the purpose of generating higher coherence. The whole control field is digitized into $M = 278$ slices with $\Delta t = 20$ μs width, while 110 slices are for three sub-pulses and 168 slices remain zero to realize the two 1.68 ms free evolutions (Appendix E, See [Supplementary information](#)). The total dynamics of the pulse is given by U_1^M in Eq. (3).

The fitness function is defined as $f = \text{tr}(\rho_f \tilde{\rho})$, a metric for the control fidelity, where $\tilde{\rho} = U_1^M(\rho_i)$ is the experimental state and $\rho_f = Z^{\otimes 12}$ is the target. In our experiment, only one measurement of the expectation value of $\langle Z^{\otimes 12} \rangle$ suffices to attain f after each iteration. If f does not hit our preset value with the current control field, we navigate the control field along its gradient g . In fact, to measure $g_x[m]$ (the same for $g_y[m]$) which is the gradient of slice m , we just need three steps: insert a local $\pm\pi/2$ pulse on every qubit about x -axis between slice m and $m+1$; apply this new control field to the initial state ρ_i and measure f (see Fig. 2b); compute $g_x[m]$ by directly combining these $\pm\pi/2$ -inserted results via Eq. (11). As long as accurate local $\pm\pi/2$ pulses are available for each qubit, g can be measured on a quantum processor. In experiment, we have designed a 1 ms $\pi/2$ pulse on every ^{13}C nucleus with the simulated fidelity over 99.7% (Appendix D, See [Supplementary information](#)). Having the gradient, we can update the control field and continue the MQFC procedure until a desired f is attained.

Direct observation of 12-coherence

After the preparation of the 12-coherent state, the next step is to observe it. In NMR spectroscopy, multiple coherence is hard to be observed directly in a one-dimensional spectrum, i.e., by flipping

the target spin to the x - y plane while others remain in Z . If all coupling between the target spin and other spins can be resolved, such observation is feasible. For example, in a two-qubit system, we can flip spin one to X to observe ZZ . In fact, XZ can be written as

$$XZ = X \otimes |0\rangle\langle 0| - X \otimes |1\rangle\langle 1|. \quad (12)$$

The first term $X \otimes |0\rangle\langle 0|$ leads to a positive peak at $\nu_1 - J_{12}/2$ in the spectrum, as the J -coupling term shifts the frequency of qubit 1 by $-J_{12}/2$. Analogously, the second term $X \otimes |1\rangle\langle 1|$ leads to a negative (due to the minus sign before the term) peak at $\nu_1 + J_{12}/2$. Generally, these two peaks can be resolved in the spectrum as long as J is large enough to separate them in frequencies. However, to observe multiple coherence, this requirement is of great challenge, since all J -couplings between the target spin and other spins should be sufficiently large to prevent the annihilations of positive and negative peaks. As a result, two-dimensional spectra and special techniques are usually employed to observe multiple coherence in conventional NMR spectroscopy.

For the purpose of NMR quantum computing, it is certainly better if one can read out multiple coherence directly in a one-dimensional spectrum, as one-dimensional spectrum reflects the state information more intuitively and reduces experimental running time remarkably compared to the two-dimensional spectroscopy. In our 12-qubit processor, although there are a few couplings as small as 0.01 Hz (Appendix C, See [Supplementary information](#)), a direct observation of 12-coherence $Z^{\otimes 12}$ is still available on C_7 . Figure 3a exhibits a strong agreement between experimental observation 12-coherence with merely 32 scans and the simulation, after rescaling the experimental result by 1.21 times to compensate for decoherence. To the best of our knowledge, our experiment is the first direct observation of multiple coherence beyond ten spins, and provides a valid evidence that our 12-qubit processor possesses excellent individual controllability and the potential to be a universal 12-qubit quantum processor.

Readout sequence

Although the direct observation of 12-coherence with 32 scans in Fig. 3a demonstrates our control precision, it is not suitable for the many experimental runs during the optimization since 32 scans leads to a great time cost. One solution is to decouple the five ^1H spins to boost the signal-to-noise ratio (SNR) by $2^5 = 32$ times, which exactly compensates for the required scan number. We have designed a readout pulse sequence to realize it as shown in Fig. 4.

The local pulses in the readout sequence are computed by SSGRAPE, and the sequence is implemented before every measurement. The phase correction is a z -rotation to neutralize the unwanted chemical shift rotation during the free evolution. If the state is $Z^{\otimes 12}$, the five ^1H spins will be evolved to the identity state after the readout sequence, and the decoupling of ^1H leads to the C_7 spectrum as shown in Fig. 3b, which is measured with a single scan. We then use spectrum fitting to obtain the signal's amplitude and phase, and thus the value of $\langle Z^{\otimes 12} \rangle$.

This readout sequence induces errors in terms of decoherence and pulse imperfections. For the former one, through our simulation we find that it leads to about 30% signal loss, which is reasonable since multi-coherence is exceptionally vulnerable to decoherence. Therefore, this factor is taken into account for all the measurement results, that is, the measured values are rescaled by about 1.3. With respect to the pulse imperfection, it consists of two parts: the imperfection of the sequence itself, i.e., some approximations when we design this simple readout sequence, and the infidelities in implementing the pulses. In total, 3.5% error arises in simulation. We use this value as the uncertainty of the experimental value of $\langle Z^{\otimes 12} \rangle$, namely, the error bars in Fig. 3c.

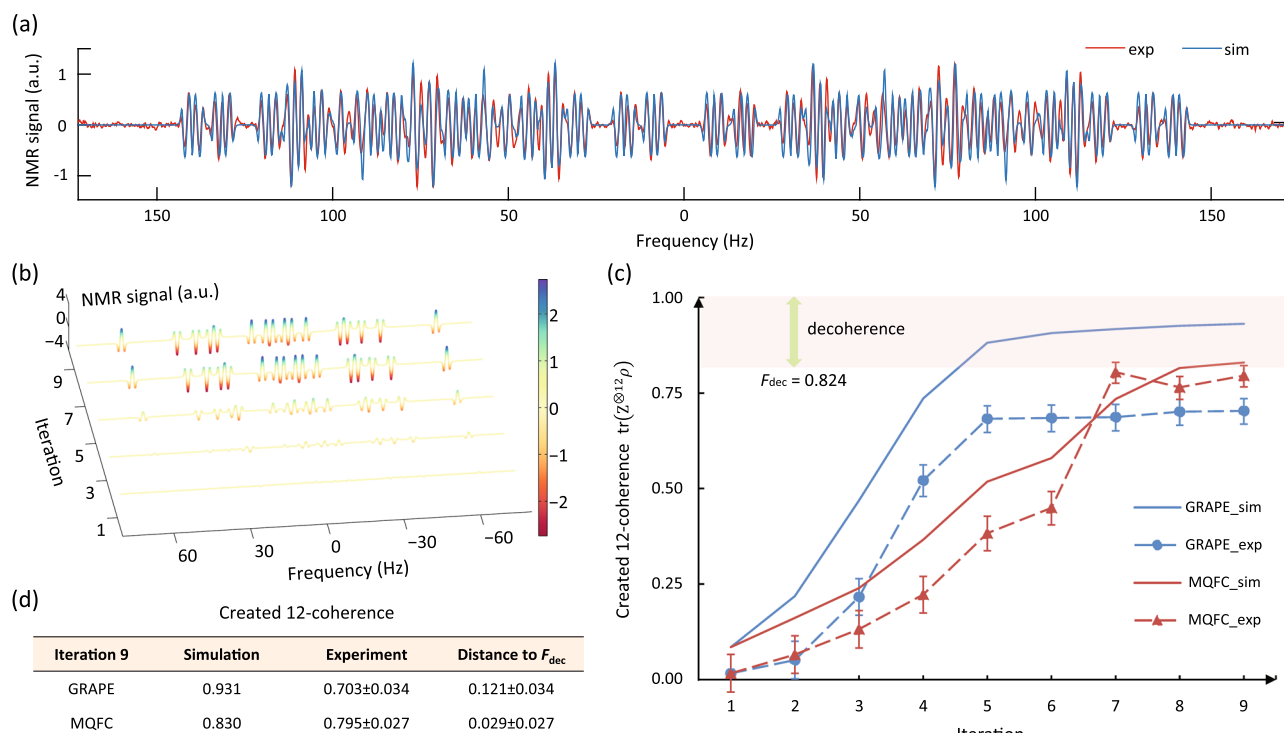


Fig. 3 Experimentally created 12-coherence using MQFC. **a** Direct observation of the created 12-coherence in one-dimensional NMR spectrum (red), where C_7 is the probe qubit. Simulated spectrum (blue) is also plotted. The experimental result is rescaled by 1.21 times to compensate for the decoherence effect for better visualization. **b** Spectra of 12-coherence after each odd iteration during the MQFC optimization. Unlike the direct observation, a readout technique is applied to gain a higher resolution. A color scale indicates peak intensities. The height of the peaks is proportional to the value of created 12-coherence. **c** Comparison between GRAPE (blue) and MQFC (red) optimizations, both in simulation (solid; without decoherence accounted) and experiment (dashed). F_{dec} is the numerical simulation of decoherence during the 12-coherence creation. Compared to the GRAPE algorithm, MQFC optimization is worse in simulation, but better in experiment. The error bars are plotted by the infidelity of the readout pulse. **d** Results at iteration 9. The experimental 12-coherence reaches 0.795 using MQFC which approaches the $F_{dec} = 0.824$ bound, while GRAPE only leads to 0.703 (i.e., 0.121 lower than F_{dec}) in experiment

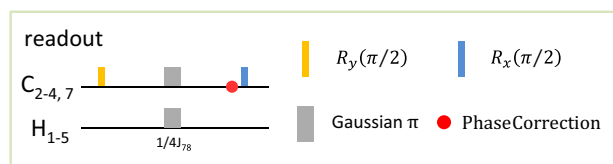


Fig. 4 Readout sequence to boost the SNR of the C_7 spectrum. It transforms the ^1H spins from Z to identity and thus enables the decoupling of ^1H channel. The phase correction compensates for the chemical shift evolutions, after which all relevant spins are along the y-axis. In principle, this technique improves the SNR by a factor of 32, and makes the measurement of f or g practical using one scan

Experimental results

Figure 3b shows the spectrum of $\tilde{\rho}$ after the readout stage for each odd iteration. The peak intensities correspond to the value of $f = \text{tr}(Z^{\otimes 12}\tilde{\rho})$, which clearly shows that MQFC increases f during the optimization. This demonstrates that MQFC is a practical technique for designing control fields in large quantum systems.

Our experiment also exhibits MQFC's ability of correcting unknown experimental errors. To demonstrate this improvement, we implement another group of 12-coherence-creating experiments, where all experimental settings are the same except that the pulse is generated from the classical SSGRAPE method other than the MQFC approach. We then compare these two groups of experiments. Figure 3c illustrates the result of SSGRAPE and MQFC pulses both in simulation and experiment. Focusing on the final result at iteration 9 in Fig. 3d, in experiment SSGRAPE finally creates a 12-coherence with $f = 0.703 \pm 0.034$, whereas MQFC

pulse creates $f = 0.795 \pm 0.027$. This experimental improvement (nearly 10%) disagrees with simulation, as in simulation MQFC (0.830) is even worse than SSGRAPE (0.931).

Considering that MQFC is a feedback-control process, some incomplete knowledge of the experimental quantum process, such as the nonlinearity of the pulse generator or imprecision of the molecular Hamiltonian, may be inherently corrected during the optimization. Indeed, the experiment clearly suggests that MQFC is advantageous in terms of correcting errors from unknown sources. Furthermore, we simulate the decoherence effect during the procedure, and find that the upper bound of $\text{tr}(Z^{\otimes 12}\tilde{\rho})$ in the presence of dephasing noise is about 0.824 (see Methods). Note that our MQFC result finally reaches 0.795, which is very close to this bound, demonstrating that our control of this 12-qubit processor is close to the theoretical prediction after accounting for decoherence.

DISCUSSION

Scalability

One major concern about control methods is their scalability with the number of qubits n . Our MQFC protocol involves a single experiment to measure f and $4nM$ experiments to measure g for each iteration, where n is the number of qubits. Assuming each experiment takes τ_{exp} time, the MQFC in total consumes $T_{it} = (4nM + 1)\tau_{\text{exp}}$ for each iteration. For comparison, one has to deal with massive $2^n \times 2^n$ matrix multiplications and exponentials using GRAPE on a classical computer. The speed-up comes from the fact that MQFC utilizes the evolution of the quantum system instead of computing the system's dynamics when evaluating f and g .

For other potential problems when scaling up the GRAPE technique, MQFC confronts similar difficulties, such as how to effectively represent a generic target state, how to choose a good initial guess, how to determine the pulse parameters before optimization, and how many iterations are needed to reach a satisfactory fidelity. Unfortunately, experimental observation of running time vs. number of qubits is not likely in NMR, since changing the number of qubits would usually require a different sample with different characteristics. So we cannot experimentally compare the scaling of MQFC vs. GRAPE, instead we must be satisfied with the fact that MQFC performs well at the 12-qubit level and should theoretically scale better than GRAPE under standard assumptions. See Appendix A (See [Supplementary information](#)) in for details.

One may also ask if there could be other classical algorithms that scale as well (or better than) MQFC. This question remains open, but it seems very unlikely—the gradient calculation is based on the dynamics as shown in Eq. (3), i.e., the expected classical algorithm needs to simulate the dynamics of an NMR system in an efficient way. Even when boiling down to our particular state engineering task, as far as we can tell, there is no employed numerical method^{29–31} to simplify such an optimization, despite extensive work on the subject since the early days of experimental quantum computing. Moreover, MQFC can correct unknown errors to some extent, while open-loop algorithms should require knowledge about the noise spectrum in advance, which is usually impractical for large quantum systems. In this sense, another potential application of MQFC is to demonstrate the quantum computing supremacy,³² where initial endeavors have been made in other systems, for example in a recent five-photon boson sampling experiment.³³

Optimizing clifford gates

While our experiment focuses on state engineering, MQFC can also be used for other quantum optimization tasks. As an example, we consider optimizing the pulse sequence for a generic Clifford gate. It is possible to use twirling to estimate the average gate fidelity of a Clifford gate efficiently.²⁸ The twirling protocol is based on finding the fidelity between experimental states following the pulse sequence and the corresponding desired states following the ideal gate. In principle this should be done for a complete set of initial states, but a randomized protocol can be used to approximate the gate fidelity with a constant number of experiments. The MQFC protocol can be modified to extract the desired fidelities and optimize the pulse sequence accordingly (details in Appendix B, See [Supplementary information](#)). Note that right after our work, a five-qubit implementation of a different quantum algorithm for gate optimization was reported.³⁴

Comparison with previous work

MQFC was originally introduced in Ref. 24 where it was implemented on a 7-qubit NMR processor. There are two significant improvements in our work. First, our work clearly demonstrates the superiority of MQFC in correcting unknown errors with around 10% fidelity boost compared to the best classical optimization result, while in the 7-qubit experiment no improvement was observed. The reason could be that the characterization of a 7-qubit system is much more accurate than a 12-qubit one, indicating that MQFC should be more powerful when dealing with large systems as the knowledge of larger systems are more likely to be incomplete. Second, our 12-qubit experiment lies at the cutting edge of present experimental quantum computing, and the capability of individual controls at this qubit number is state-of-the-art. As a comparison, in a recent work,³⁵ the 10-qubit entanglement in a superconducting circuit is created with fidelity 0.668 using *global* control. Moreover, we

demonstrated that at the 12-qubit level, the algorithm is already fast enough to justify its use as a tool in the lab.

In summary, we have created a 12-coherence state on an NMR quantum processor using MQFC. Our experimental procedure and result, in particular the direct observation of 12-coherence with one qubit as the probe, signify the capability of our quantum processor to serve as a universal 12-qubit quantum processor with high-fidelity individual controls on each qubit. In terms of control field optimization, our experiment demonstrates two superiorities in efficiency and experimental performance of MQFC beyond its classical counterpart. MQFC requires a running time that scales linearly with the number of qubits, and yields about 10% improvement compared to the best result via classical optimization. This optimization approach could be exceptionally useful in a large system with incomplete characterization, and is readily transferrable to other systems such as superconducting circuits or nitrogen-vacancy centers in diamond. We expect that, as experiments involving more than 10 qubits become more common, quantum feedback methods such as MQFC will become standard tools in quantum computing labs.

METHODS

To numerically simulate the decoherence effect in our 12-qubit system, we first make the following assumptions: the environment is Markovian; only the T_2 dephasing mechanism is taken into account since T_1 effect is negligible in our circuit; the dephasing noise is independent between all qubits; the dissipator and the total Hamiltonian commute in each pulse slice as $\Delta t = 20 \mu\text{s}$ is small. With these assumptions, we solve the master equation in two steps for each Δt : evolve the system by the propagator in Eq. (3), and subsequently apply the dephasing noise for Δt which is an exponential decay of off-diagonal elements in the density matrix. The typical length of simulating our 12-qubit experiment in the presence of dephasing noise is in the magnitude of days on a desktop computer. The simulation shows that at most $F_{\text{dec}} = 0.824$ of $Z^{\otimes 12}$ can be achieved with the 5.56 ms MQFC pulse applied on $Z^{\otimes 7}I^{\otimes 5}$, which is reasonable as high-order coherence is very vulnerable to the dephasing noise. Alternatively speaking, the upper bound of the MQFC experimental result is 0.824, since the optimization procedure does not include the function of robustness against dephasing noise yet.

Data availability

The data sets generated and/or analyzed during the current study are available from the corresponding author on reasonable request.

ACKNOWLEDGEMENTS

We thank Anthony P. Krismanich, Ahmad Ghavami, and Gary I. Dmitrienko for synthesising the NMR sample. This research was supported by CIFAR, NSERC and Industry of Canada. K.L., H.L., and G.L. acknowledge National Natural Science Foundation of China under Grant Nos. 11175094 and 91221205.

AUTHOR CONTRIBUTIONS

D.L., J.L. and R.L. conceived the experiments. D.L. and K.L. performed the experiment and analyzed the data. D.L., H.K., A.P., G.F., and H.L. tested the sample and developed the control techniques. J.L. and A.B. provided theoretical support. G.L., J.B., B.Z., and R. L. supervised the project. D.L., K.L., and J.L. wrote the manuscript with feedback from all authors. D.L., K.L., and J.L. contributed equally to this work.

ADDITIONAL INFORMATION

Supplementary information accompanies the paper on the *npj Quantum Information* website (<https://doi.org/10.1038/s41534-017-0045-z>).

Competing interests: The authors declare that they have no competing financial interests.

Publisher's note: Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

REFERENCES

- Shor, P. W. Algorithms for quantum computation: discrete logarithms and factoring. In *Proc. 35th Annual Symposium on Foundations of Computer Science* 124–134 (IEEE Computer Society Press, Santa Fe, NM, USA, 1994).
- Grover, L. K. A fast quantum mechanical algorithm for database search. In *Proc. 28th Annual ACM Symposium on Theory of Computing* 212–219 (ACM, New York, 1996).
- Farhi, E. et al. A quantum adiabatic evolution algorithm applied to random instances of an np-complete problem. *Science*. **292**, 472–475 (2001).
- Buluta, I. & Nori, F. Quantum simulators. *Science*. **326**, 108–111 (2009).
- Georgescu, I., Ashhab, S. & Nori, F. Quantum simulation. *Rev. Mod. Phys.* **86**, 153 (2014).
- Harrow, A. W., Hassidim, A. & Lloyd, S. Quantum algorithm for linear systems of equations. *Phys. Rev. Lett.* **103**, 150502 (2009).
- Ladd, T. D. et al. Quantum computers. *Nature*. **464**, 45–53 (2010).
- Gratl, T., Spörl, A., Huckle, T., Glaser, S. J. & Schulte-Herbrüggen, T. Parallelising matrix operations on clusters for an optimal control-based quantum compiler. In *European Conference on Parallel Processing* 751–762 (Springer, Berlin, Heidelberg, 2006). https://link.springer.com/chapter/10.1007/11823285_78.
- Nielsen, M. A. & Chuang, I. L. *Quantum computation and quantum information* (Cambridge University Press, 2010). <https://www.cambridge.org/core/books/quantum-computation-and-quantum-information/01E10196D0A682A6AEFFEA52D53BE9AE>.
- Brif, C., Chakrabarti, R. & Rabitz, H. Control of quantum phenomena: past, present and future. *New J. Phys.* **12**, 075008 (2010).
- Boykin, P. O., Mor, T., Roychowdhury, V., Vatan, F. & Vrijen, R. Algorithmic cooling and scalable nmr quantum computers. *Proc. Natl. Acad. Sci. USA*. **99**, 3388–3393 (2002).
- Baugh, J., Moussa, O., Ryan, C. A., Nayak, A. & Laflamme, R. Experimental implementation of heat-bath algorithmic cooling using solid-state nuclear magnetic resonance. *Nature*. **438**, 470–473 (2005).
- Souza, A. M., Zhang, J., Ryan, C. A. & Laflamme, R. Experimental magic state distillation for fault-tolerant quantum computing. *Nat. Commun.* **2**, 169 (2011).
- Bennett, C. H. Quantum cryptography: public key distribution and coin tossing. In *International Conference on Computer System and Signal Processing* 175–179 (IEEE, New York, 1984).
- Khaneja, N., Reiss, T., Kehlet, C., Schulte-Herbrüggen, T. & Glaser, S. J. Optimal control of coupled spin dynamics: design of nmr pulse sequences by gradient ascent algorithms. *J. Magn. Reson.* **172**, 296–305 (2005).
- Ryan, C., Negrevergne, C., Laforest, M., Knill, E. & Laflamme, R. Liquid-state nuclear magnetic resonance as a testbed for developing quantum control methods. *Phys. Rev. A*. **78**, 012328 (2008).
- Zhang, Y., Ryan, C. A., Laflamme, R. & Baugh, J. Coherent control of two nuclear spins using the anisotropic hyperfine interaction. *Phys. Rev. Lett.* **107**, 170503 (2011).
- Waldherr, G. et al. Quantum error correction in a solid-state hybrid spin register. *Nature*. **506**, 204–207 (2014).
- Dolde, F. et al. High-fidelity spin entanglement using optimal control. *Nat. Commun.* **5**, 3371 (2014).
- Motzoi, F., Gambetta, J., Reberntrost, P. & Wilhelm, F. K. Simple pulses for elimination of leakage in weakly nonlinear qubits. *Phys. Rev. Lett.* **103**, 110501 (2009).
- Egger, D. J. & Wilhelm, F. K. Optimized controlled-z gates for two superconducting qubits coupled through a resonator. *Supercond. Sci. Technol.* **27**, 014001 (2013).
- Nebendahl, V., Häffner, H. & Roos, C. Optimal control of entangling operations for trapped-ion quantum computing. *Phys. Rev. A*. **79**, 012312 (2009).
- Schindler, P. et al. Experimental repetitive quantum error correction. *Science*. **332**, 1059–1061 (2011).
- Li, J., Yang, X., Peng, X. & Sun, C.-P. Hybrid quantum-classical approach to quantum optimal control. *Phys. Rev. Lett.* **118**, 150503 (2017).
- Reberntrost, P., Schulz, M., Petruccione, F. & Lloyd, S. Quantum gradient descent and Newton's method for constrained polynomial optimization. arXiv:1612.01789 (2016).
- Vijay, R. et al. Stabilizing rabi oscillations in a superconducting qubit using quantum feedback. *Nature*. **490**, 77–80 (2012).
- Hirose, M. & Cappellaro, P. Coherent feedback control of a single qubit in diamond. *Nature*. **532**, 77–80 (2016).
- Lu, D. et al. Experimental estimation of average fidelity of a clifford gate on a 7-qubit quantum processor. *Phys. Rev. Lett.* **114**, 140505 (2015).
- Daley, A. J. Quantum trajectories and open many-body quantum systems. *Adv. Phys.* **63**, 77–149 (2014).
- Mascarenhas, E., Flayac, H. & Savona, V. Matrix-product-operator approach to the nonequilibrium steady state of driven-dissipative quantum arrays. *Phys. Rev. A*. **92**, 022116 (2015).
- Lloyd, S. & Montangero, S. Information theoretical analysis of quantum optimal control. *Phys. Rev. Lett.* **113**, 010502 (2014).
- Preskill, J. Quantum computing and the entanglement frontier. arXiv:1203.5813 (2012).
- Wang, H. et al. High-efficiency multiphoton boson sampling. *Nat. Photon.* **11**, 361–365 (2017).
- Dive, B., Pitchford, A., Mintert, F. & Burgarth, D. In situ upgrade of quantum simulators to universal computers. arXiv:1701.01723 (2017).
- Song, C. et al. 10-qubit entanglement and parallel logic operations with a superconducting circuit. *Phys. Rev. Lett.* <https://journals.aps.org/prl/accepted/da073Y7cPc31266793e1794762950ddac05b460f> (2017).



Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2017