

# Quantum Computers for Weather and Climate Prediction

The Good, the Bad, and the Noisy

F. Tennie and T. N. Palmer 

**ABSTRACT:** Over the past few years, quantum computers and quantum algorithms have attracted considerable interest and attention from numerous scientific disciplines. In this article, we aim to provide a nontechnical yet informative introduction to key aspects of quantum computing. We discuss whether quantum computers one day might become useful tools for numerical weather and climate prediction. Using a recently developed quantum algorithm for solving nonlinear differential equations, we integrate a simple nonlinear model. In addition to considering the advantages that quantum computers have to offer, we shall also discuss the challenges one faces when trying to use quantum computers for real-world problems involving “big data,” such as weather prediction.

**KEYWORDS:** Atmosphere; Forecasting; Numerical weather prediction/forecasting

<https://doi.org/10.1175/BAMS-D-22-0031.1>

Corresponding author: T. N. Palmer, [tim.palmer@physics.ox.ac.uk](mailto:tim.palmer@physics.ox.ac.uk)

In final form 9 October 2022

© 2023 American Meteorological Society. This published article is licensed under the terms of a Creative Commons Attribution 4.0 International (CC BY 4.0) License



Hardly a day goes by without the announcement of some new breakthrough in quantum technology. Quantum computers can already perform calculations that classical computers would be unable to perform in reasonable lengths of time—quantum supremacy, as it is known. Although current quantum computers have limited power, there are plans for next-generation quantum computers with millions of so-called qubits—the elements of a quantum computer.

This paper was motivated by a comment made by one of our colleagues: “As soon as we have a quantum-Fortran compiler, we’ll be able to start using these computers in meteorology.” Unfortunately, it is not as simple as this—the way quantum computers work is profoundly different from the way classical computers work. Indeed, it is so different that it is very difficult for someone who is not well versed in quantum mechanics to understand what exactly these differences are.

In this paper, we try to answer the question of whether quantum computers will ever take over from classical computers in trying to make weather forecasts, or seasonal predictions, or indeed climate change projections, using the underlying laws of physics. We do this assuming a typical meteorological audience with a good knowledge of fluid mechanics and classical computers but little knowledge of quantum computing theory.

In attempting to answer this question, we have been motivated by the title of the movie *The Good, the Bad and the Ugly*, but replacing it with the “The Good, the Bad, and the Noisy.” In quantum computing there are certainly good points and bad points. The question is which will win out overall. In this review, we have taken the opportunity to discuss some advances of our own, using quantum algorithms to solve nonlinear differential equations exponentially faster than classical algorithms.

We do not aim to provide a technical review article on the field of quantum computing. For a comprehensive introduction, the interested reader should consult the excellent book by Nielsen and Chuang (2010). A selection of reviews on various quantum computing paradigms and recent developments may be found in Nimbe et al. (2021), Bharti et al. (2022), Montanaro (2016), and Cerezo et al. (2021).

## The Good

**Key elements of quantum computation.** Everyone knows that the elements of a classical computer are its bits. Each bit has a value of 0 or 1. The elements of a quantum computer are qubits. It is easy to describe what a qubit is since it has a precise mathematical definition. However, it is difficult to describe what a qubit means physically since (it is safe to say), the physical meaning of quantum states is still a matter of debate and discussion. Some pedagogical discussions of quantum computing try to find simple analogies for qubits, in an effort to make them understandable. However, the truth of the matter is that they are not, currently at least, understandable in terms of intuitive analogs. In the following, it is probably best to follow Richard Feynman’s dictum:

Do not keep saying to yourself, if you can possibly avoid it, “But how can it be like that?” because you will get ‘down the drain’, into a blind alley from which nobody has escaped. Nobody knows how it can be like that.

So instead, we will describe a qubit mathematically. It is an element of a complex Hilbert space: a vector space equipped with an inner product where the vectors have complex amplitudes. Just as in physical space, a vector can be described by its components relative to an orthogonal basis, i.e.,

$$\mathbf{v} = a \hat{\mathbf{i}} + b \hat{\mathbf{j}} + c \hat{\mathbf{k}}, \quad (1)$$

where  $a$ ,  $b$ , and  $c$  are real numbers, so a vector in a two-dimensional complex Hilbert space can be written

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, \quad (2)$$

where  $\alpha$  and  $\beta$  are complex numbers. In this representation, the orthogonal basis vectors  $|0\rangle$  and  $|1\rangle$  are completely arbitrary, just as  $\hat{\mathbf{i}}$ ,  $\hat{\mathbf{j}}$ , and  $\hat{\mathbf{k}}$  are arbitrary. With two complex numbers, it would seem that this vector has 4 degrees of freedom. However, in quantum mechanics, the vectors are normalized, and there is an undefined global phase factor, meaning that in fact there are actually only 2 degrees of freedom. In appendix A, we provide further details on qubits, how they compare to classical bits, and on the Dirac notation used in Eq. (2).

The very simplest quantum computer (too simple to be useful) would take as its input state the vector  $|\psi\rangle$  in Eq. (2) with some initial values  $\alpha(t_i)$  and  $\beta(t_i)$ . There would then be a quantum circuit where certain rotations of the vector are performed in hardware. At the end of these rotations, the vector is still in the state (2) but with final values  $\alpha(t_f)$  and  $\beta(t_f)$ . These rotations are effected by the governing equation of quantum mechanics, the linear Schrödinger equation. In terms of the complex Hilbert space, the operations that map initial vectors to final vectors can be described by unitary matrices.

There is one final yet utterly critical part of our simple quantum computer. One has to “measure” the qubit. Again, while there is an experimentally well-defined procedure for performing such measurements, no one is quite sure what is going on at a theoretical level when a measurement takes place. For example, 2020 Nobel Laureate Roger Penrose has proposed that the process of measurement occurs when the effects of self gravitation of quantum systems are no longer negligible.

Again, we ignore such deep matters and simply describe the recipe which all students of quantum mechanics know. To perform the measurement, you first choose a vector basis relative to which the measurement takes place. If  $|0\rangle$  and  $|1\rangle$  describe this measurement basis, then, after measurement, the qubit will be found to be in the  $|0\rangle$  state with probability  $|\alpha(t_f)|^2$  and in the  $|1\rangle$  state with probability  $|\beta(t_f)|^2$ . This appearance of probability led Einstein to complain that God does not play dice. However, most physicists today believe that quantum uncertainty is fundamentally encapsulated in the laws of quantum physics, making it different in character to chaotic uncertainty (whose origin is our inability to know about flaps of butterflies wings). One of the authors believes that scientific orthodoxy will ultimately be overturned on this matter (Palmer 2022). However, this is not the topic of this paper.

In practice, it can be arranged so that a bell rings, or a light flashes, when the state evolves to  $|0\rangle$ . We can estimate the modulus of the complex amplitudes by carrying out multiple measurements of the state and counting the number of times the state evolved to  $|0\rangle$ .

The power of a quantum computer comes from being able to “entangle” qubits together. We form multiqubit states by what are called tensor products of Hilbert vectors. For example, the Hilbert state of a two-qubit system can be written

$$|\psi\rangle = \gamma_1|00\rangle + \gamma_2|01\rangle + \gamma_3|10\rangle + \gamma_4|11\rangle. \quad (3)$$

Here there are now four basis states, and on measurement the system evolves to one of these basis states with a probability given by the squared modulus of the associated complex amplitude. Figure 1 illustrates and contrasts a two-bit classical and a two-qubit quantum circuit.

Now we can already start to glimpse the reason why quantum computers are powerful. Two-qubit states are associated with four complex amplitudes, i.e., eight real numbers (less two for normalization and global phase). It is easy to generalize (3). A  $n$ -qubit state can be written

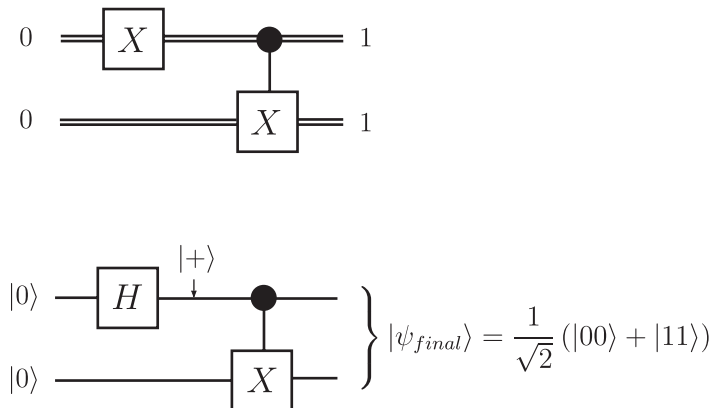
$$|\psi\rangle = \alpha_1 \left| \underbrace{00\dots 00}_n \right\rangle + \alpha_2 \left| \underbrace{00\dots 01}_n \right\rangle + \alpha_3 \left| \underbrace{00\dots 10}_n \right\rangle + \dots + \alpha_{2^n} \left| \underbrace{11\dots 1}_n \right\rangle. \quad (4)$$

In this way,  $n$ -qubit states are associated with  $2^n$  complex numbers, i.e.,  $2^{n+1}$  real numbers (less 2 for normalization and global phase). This exponential increase in the dimension of the Hilbert space with qubit number  $n$  has some remarkable consequences.

Consider a modern numerical weather prediction (NWP) model, with say 1 billion variables. These variables represent the components of wind velocity, temperature, humidity, and other variables, on a grid that covers Earth and extends up into the stratosphere. How many qubits are needed to encode the state of such an NWP model into the complex amplitudes of a multiqubit Hilbert state? A simple estimate is to take the log to base 2 of 1 billion. The answer is 30. By modern quantum computer standards, a 30-qubit quantum computer is not an especially large computer.

Hence, it appears possible to initialize a 30-qubit quantum computer with the initial state of a billion-variable NWP model. So far, so good! The next question is whether it is possible to encode the dynamical equations of an NWP model (e.g., the Navier–Stokes equation) into the unitary transformations that a quantum state evolves through before measurement. It may seem there is a fundamental obstacle here—that the Schrödinger equation is linear, while the Navier–Stokes and other equations in NWP are nonlinear. However, in fact that obstacle can be overcome. This is an area of research in which the authors, in collaboration with a group at the Massachusetts Institute of Technology, are actively engaged. In the subsection below we give a brief summary of this research. As with much in quantum computing, it is impossible to provide a schematic outline without having to give some technical details. The reader not interested in such details can skip to the end of this section.

In addition to the quantum computing paradigm described above which consists of a controlled unitary evolution of quantum states followed by measurements of the qubits, a range of other approaches have been suggested. A type of quantum machine referred to as an annealer, such as the much-reported



**Fig. 1. (top)** Classical circuit processing two bits initialized in the state “0.” The X gate flips the upper bit, and the controlled X gate subsequently flips the lower bit. Throughout, each bit is in a definite state of either “0” or “1.” **(bottom)** Simple quantum circuit. The Hadamard (H) gate transforms the upper qubit into the state  $|+\rangle = 1/\sqrt{2}(|0\rangle + |1\rangle)$ . The controlled X gate then entangles both qubits. The final state corresponds to  $\gamma_1 = \gamma_4 = 1/\sqrt{2}$  and  $\gamma_2 = \gamma_3 = 0$  in Eq. (3). Note that the information “stored” in the amplitudes is nonlocal and cannot be attributed to an individual qubit unlike in the classical case.

D-Wave machine, seeks to encode computational tasks as minimization problems of a target function (McGeoch 2020). By carefully tailoring the interaction between the internal qubits, the solution of the minimization is given by the lowest energy state of the machine. It is yet unclear whether this approach offers a quantum advantage and whether it is computationally universal.

Hybrid quantum computing represents yet another quantum computing paradigm, where the quantum hardware is used to evaluate a parameterized target function. The overall task is to find the parameter values which minimize the target function. After each evaluation of the target function on the quantum chip, a classical computer determines a new set of parameters for which the target function will be evaluated until a local optimum has been determined. As with the D-Wave machine, it has yet to be shown that this scheme can offer a scalable quantum advantage.

**Nonlinear differential equation quantum solver.** Our way to solve nonlinear differential equations is to encode multiple copies of a quantum state and let these copies interact with each other—via what in quantum theory is known as an interaction Hamiltonian (cf. Lloyd et al. 2020). We then study the evolution of one of the copies. It turns out that the equation of motion for a single copy is itself nonlinear; the nonlinearity originates as the leading-order term in the expression for the interaction between copies. It can be understood and described as an effective mean field that arises from averaging the degrees of freedom of all but one copy. In this framework, however, one finds that higher-order terms give rise to mean-field errors. These errors tend to accumulate during the course of a computation, which will be demonstrated in an example further below. In a recent paper, Lloyd et al. (2020) utilized mean-field nonlinearities to create a quantum nonlinear solver that offers an exponential resource gain over classical solvers. We give a brief description of this quantum algorithm, based on the ideas above, in appendix B.

For the rest of this section we are going to apply the Lloyd et al. algorithm to a simple nonlinear differential equation. Consider the flow of a single dynamical variable subject to  $\dot{x}(t) = x - \alpha x^3$ . For any finite initial value  $x_0$ , the system will converge to a fixed value in a sigmoid-shape trajectory in the  $t$ - $x$  plane. We have emulated the execution of the quantum solver on a classical computer because the quantum resources currently available are not yet sufficient to run the solver (see discussion below). The results are shown in Fig. 2.

In the left panel, the output of the quantum solver is shown for various different initial values. It is benchmarked by a classical forward Euler integration and the analytic solution. We observe that the output of the quantum solver is in good agreement with the classical output up to about 15 integration steps. This is no mere coincidence. In carrying out the integration,  $N = 15$  identical system copies have been used to run the quantum solver. As mentioned above, the inclusion of nonlinear terms leads to

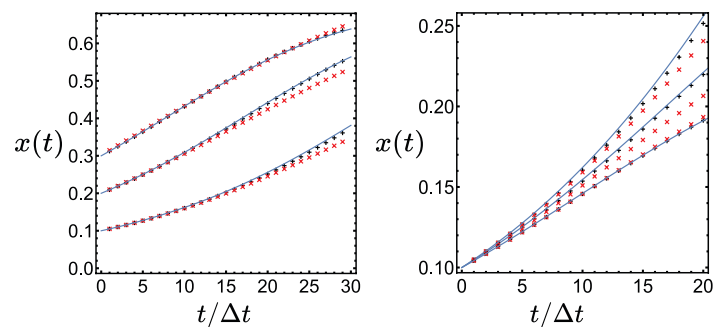


Fig. 2. Results of integration of  $\dot{x} = x - \alpha x^3$ . Red diagonal crosses mark the output of the quantum solver; by comparison the forward Euler method executed on a classical computer results in numerical values marked by black horizontal/vertical crosses. The blue solid line represents the analytic solution. (left) Integration of the model for different initial conditions  $x_0 = 0.1, 0.2, 0.3$  for a relative coupling strength  $\alpha = 2$ . (right) Integration of the model for different coupling strengths  $\alpha = 2, 8, 16$  (top to bottom branch) and initial value  $x_0 = 0.1$ . In both cases the step size was chosen as  $\Delta t = 0.05$  and the quantum solver was initialized with  $N = 15$  identical copies of the state.

additional mean-field errors which build up during each integration step. As those errors usually are suppressed by a factor of  $1/N$ , the errors limit the range of the integration to  $\mathcal{O}(N)$  steps. Although this might appear to be a drawback of the quantum solver, the linear scaling actually represents a major improvement over former quantum algorithms used to integrate nonlinear systems, which required  $\mathcal{O}(\exp[N])$  copies (cf. Leyton and Osborne 2008). In the second figure, we can see that the quantum solver is capable of handling various different relative sizes of the nonlinear terms. Generally, however, we expect that the number of required system copies increases with the relative strength of the nonlinearities of the system. Unstable systems with positive Lyapunov exponents might require an exponentially increasing number of copies.

Classical NWP utilizes ensemble methods (cf. Palmer 2022). It is therefore instructive to compare the results of integrating our one dimensional toy model using the quantum solver to a classical ensemble-average solution. As shown in Fig. 3, we find that the quantum solver output is closer to the ensemble average than to a single classical forward Euler solution. If generally true, this would be a remarkable result. Further research will be required to establish whether this observation holds true for arbitrary dynamical systems.

An obvious next step would be to integrate the famous Lorenz (1963) equations with our quantum algorithm. However, there is a difficulty on classical emulators of quantum computers that we are currently working to overcome when the time derivatives are expressed as polynomials of even degree. At this point, we have determined that the problem can be solved by supplying the single copy state vector with additional “dummy” variables. The number of these dummy variables will scale linearly with the number of integration steps  $T$ . Although this does not present a problem for a quantum computer per se, it does render the emulation of the quantum solver on a classical computer unfeasible.

One might therefore ask at which point quantum hardware will become able to run the quantum solver algorithm. Let us assume that we want to integrate the Lorenz system for  $T = 100$  integration steps. This corresponds to inverting a matrix of size

$$N \approx (\#\text{dynamical variables} \cdot T)^T \approx 10^{200}. \quad (5)$$

The state vector of size  $N$  can be encoded in  $\log_2[10^{200}] \approx 655$  qubits. In addition, there is a number of ancilla qubits linked to the precision of the matrix inversion which can be estimated to be about 100. Thus, we expect a required circuit width of a few hundred error corrected qubits to be sufficient for addressing the Lorenz system. Currently

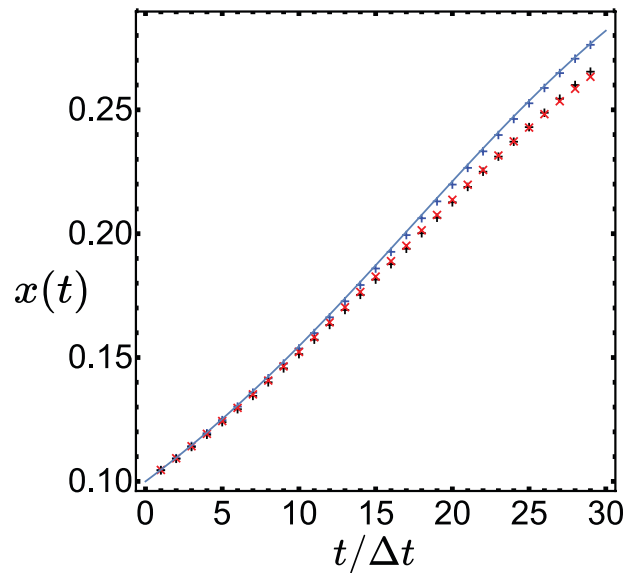


Fig. 3. Integration of  $\dot{x} = x - 8x^3$ . Comparison of a classical forward Euler solution (blue horizontal/vertical crosses) with an ensemble-averaged solution (black horizontal/vertical crosses) and the quantum solver output (red diagonal crosses). The solid blue line represents a Runge–Kutta method based solution. The step size was chosen as  $\Delta t = 0.05$  and the quantum solver was initialized with  $N = 10$  identical copies.

available quantum hardware such as IBMQ offers about 100 noisy qubits with plans to achieve about 1000 in the coming years. (For a discussion on the impact of noise, section “The Noisy.”)

So far, it thus sounds plausible to expect implementation of the quantum solver in a few years’ time. Another important quantity, however, is the circuit depth, i.e., the number of required elementary gates. Here, it is more difficult to predict the required resources. Scherer et al. (2017) predicted a circuit depth of  $10^{29}$  gates to invert a matrix of size  $N \approx 3 \times 10^9$  in the context of scattering cross-section computations. If this is correct, this poses a twofold problem. First, assuming that execution times of the individual gates are of the order of  $10^{-9}$  s, the runtime of the algorithm would be about  $10^{20}$  s. Second, in order to keep control of the errors in gate execution, a single gate would need to have a fidelity of  $\mathcal{F} \approx 1 - 10^{-29}$ . This is not realistic. Note that this gate count is closely related to properties of the matrix to be inverted. For instance, a recent publication by Perelshtein et al. (2022) reported on the inversion of a matrix of size  $N = 2^{17}$  on currently available quantum backends from IBM.

In summary, the circuit depth required for the integration of the Lorenz system crucially depends on parameters such as the connectivity of future quantum hardware. The authors of this paper believe that the benefit of their work lies in informing the design of future quantum hardware such that computational overhead will be kept to a minimum.

## The Bad

Readers may be familiar with the parlor game “20 Questions.” One person thinks of some kind of object and the rest of the group have to guess what the object is by asking questions. However, the question must be such that it can only be answered with a “yes” or a “no.” Hence, if it has been ascertained that the object is a piece of fruit, you are not allowed to ask, “What type of fruit is it?” Instead, you must ask, “Is it an orange?”

Something like these two types of question describe well the difference between classical physics and quantum physics. On a classical computer, if you want to know the value of some particular variable, such as temperature in grid box  $N$  at time step  $M$ , you simply ask the computer to write out that value. On a quantum computer, things are much more nuanced. You might start by asking the quantum computer if the temperature (at grid box  $N$  and time step  $M$ ) is greater or less than 290 K. To do this we have to add to the quantum hardware a unitary operation such that if the temperature is greater than 290 K then the state evolves to the “ $|0\rangle$ ,” so that, if this state is measured, the alarm bell, or green light, or whatever, will light with complete certainty.

Here, the fact that the state space of  $n$  qubits is exponential in  $n$  comes back to bite us. The converse of this remarkable exponential fact of quantum life is that the number of qubits that we can measure, measurements that are needed if we are to discover the results of the unitary transformations between the initial and evolved quantum states, is the logarithm of the dimension of that state space.

If we have a 30-qubit system, then essentially we can ask 30 questions about the weather around the world. Is it raining in Rio tomorrow? Is it stormy in San Francisco next week? Is the monthly outlook for Copenhagen cold?

If we had a proprietorial quantum computer and all we cared about was the weather in our backyard, then perhaps we would be satisfied with such limited information. Perhaps for climate change, all we care about is whether the global-mean temperature will rise above 2°C. However, of course, for any realistic operational weather or climate center, this is far from sufficient. We need to know the value of the many variables in an NWP model—maybe

not all billion of them, but certainly millions of values for the many variables at many parts of the world at many times.

Since the output of a quantum computation will be limited to as many bits as it processes qubits, this is an (insurmountable) bottleneck. Therefore, quantum computers are unlikely to entirely replace classical computers for NWP. They may, however, play a role performing specialized tasks within an NWP or climate model. This is consistent with the trend toward more heterogeneous computing architectures (e.g., CPU versus GPU). Quantum computers might offer an advantage in answering questions that require a resolution exceeding classical resources. However, we are a very long way from having both quantum algorithms and quantum hardware that can accomplish such specialized tasks. Indeed, our main conclusion is that quantum computers are not well suited to “big data” problems, where outputs explore the high dimensionality of the big data.

### **The Noisy**

In the first section we have seen that we can encode the state of an NWP model onto a quantum computer and that we can perhaps describe the Navier–Stokes and other nonlinear equations of NWP with the Schrödinger equation, through an interaction Hamiltonian. In this section, we jump to what seems like an “ugly” aspect of quantum computing, but argue it may not be as ugly as we might imagine at first sight.

One of the problems with quantum entanglement is that quantum systems love to get entangled with their environment. However, the environment is typically a messy, noisy system. Hence, when qubits entangle with the environment, they themselves become noisy. Quantum computer manufacturers try to minimize such environmental “decoherence” by enclosing the quantum computer in a supercold superconducting environment. However, this only works to a degree.

As a result, we have to distinguish between hypothetical noise-free “logical qubits” on the one hand, and practical noisy “physical qubits” on the other. It is possible to represent these hypothetical logical qubits if you have enough copies of the noisy physical qubits, by applying error detection and correction algorithms to the copies.

So how many copies do we need? Well the answer depends on what level of accuracy and precision we require in the first place. Furthermore, one might be able to take advantage of the fact that in NWP on classical computers, there are considerable advantages to introducing noise into computational representations of the nonlinear Navier–Stokes equation (cf. Palmer 2019). Is there a quantum analog? Due to interaction with the environment, the evolution of noisy qubits during a computation must be described by a more generalized formalism than the Schrödinger equation. While so-called density matrices are useful in understanding the outcome of several averaged runs of a noisy quantum computer, not-averaged noise encountered in a single run can be captured by the stochastic Schrödinger equation (Bouten et al. 2004). Here, the evolution of the quantum state in a single run of the quantum computer is subject to stochastic Wiener processes. Although the precise form of the stochastic terms critically depends on the design of the quantum hardware, it is plausible that stochastic terms will be injected into the stepwise integration of nonlinear systems when executed by the quantum solver. Further research will be required to establish whether potential future quantum NWP could actually benefit from this feature.

A new insight which resulted from writing this paper is that the integration of stochastic differential equations can in fact be accomplished using an extended form of the quantum solver. Here, the noise terms present at each integration step are inserted during the initial setup of the quantum computer. This means that the noise does not have to be injected at each discretized integration step, but will rather be a part of the initial quantum state.



Early studies (Epstein 1969; Fleming 1971a,b) of the state-dependent predictability of weather focused on solving the classical Liouville equation (so-called stochastic–dynamic prediction). However, in practice this proved impractical due to the large dimensionality of the state space. It is currently unclear whether noisy quantum computers would offer any advantage in solving the classical Liouville equation for such weather forecast problems—at present there is no evidence that it would.

The introduction of noise can help to reduce the dimensionality of a dynamical system. For instance, a suitable variable transformation of the deterministic three-dimensional Lorenz system yields a two-dimensional stochastic system (Palmer 2001). This offers a potential route to gauge the impact of noise in a quantum computer using near-term quantum hardware: by integrating the modified Lorenz system of reduced dimensionality on the quantum solver, the imperfections of the hardware might allow to explore chaotic behavior of the effectively stochastic system and to reproduce the Lorenz attractor.

## Conclusions

In writing this paper, the authors hope to shed light on the potential applicability of (future) quantum computing technologies in weather and climate predictions. By making use of an exponentially large state space, quantum computers offer a range of advantages when it comes to processing large sets of data. Recent developments have paved the way to integrating nonlinear systems on quantum computers (Lloyd et al. 2020). We have emulated the behavior of a nonlinear quantum solver and presented encouraging results for a number of toy models.

Unfortunately, quantum computing does not only offer benefits. In the authors' opinion, the two main challenges for quantum computers to become useful tools for researchers of disciplines involving numerical computations are the limited amount of information that can be read out after a computation, and noise. While a plausibility argument seems to indicate that a certain level of noise could even be beneficial in the integration of nonlinear dynamics, the limitations on the read-out capacity present a seemingly insurmountable problem. However, in some quantum computing paradigms such as, e.g., hybrid quantum computing, reading out the full quantum state might not be required. Unfortunately, it is yet unknown whether in these frameworks there will be a scalable quantum advantage.

With the development of more powerful quantum processors in the upcoming years, it is likely that quantum computers will become useful tools for a range of specialized applications. They, however, are unlikely to replace classical computers for NWP or climate prediction in the foreseeable future.

**Acknowledgments.** TP was supported by a Royal Society Research Professorship and FT by a Royal Society Research Professorship Enhancement Award. In addition, for this project, because of its relevance to climate modelling, TP and FT have received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (ITHACA, Grant Agreement 741112).

**Data availability statement.** Due to confidentiality agreements, supporting data can only be made available to bona fide researchers subject to a non-disclosure agreement. Details of the data and how to request access are available from [felix.tennie@physics.ox.ac.uk](mailto:felix.tennie@physics.ox.ac.uk).

## Appendix A: Bits versus qubits

Classical bits are variables assuming one of two definite values such as 0, 1 or  $\uparrow$ ,  $\downarrow$ . Each bit may be regarded as a representation of the cyclic group  $Z_2$  and can be physically implemented in any system possessing two (macroscopically) distinguishable states such as, for example, the local (macroscopic) magnetization of a tape. If we label the two states of a bit by computational basis vectors of a two-dimensional vector space, the system can either be in state

$$\begin{pmatrix} 1 \\ 0 \end{pmatrix} \sim \uparrow \text{ or } \begin{pmatrix} 0 \\ 1 \end{pmatrix} \sim \downarrow. \quad (\text{A1})$$

Multiple bits form bit strings. At any point during a computation on a deterministic classical computer those bit strings are in a definite state, e.g.,  $\bar{b} = \overline{0101100}$ .

The fundamental unit of quantum information is the qubit. It describes the state of a microscopic two-level system such as a spin-1/2-particle and can be represented by a complex vector

$$\psi = \alpha \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \beta \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \quad (\text{A2})$$

where  $\alpha, \beta \in \mathbb{C}$  and  $|\alpha|^2 + |\beta|^2 = 1$ . Commonly, this is written in Dirac's bra-ket notation as<sup>1</sup>

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle. \quad (\text{A3})$$

<sup>1</sup> Henceforth, we shall use 0, 1 and  $\uparrow$ ,  $\downarrow$  synonymously.

The set of qubits can be visualized as a sphere where  $\alpha = \cos(\theta/2)$  and  $\beta = \sin(\theta/2)e^{i\phi}$  for spherical coordinates  $\theta$  and  $\phi$  as illustrated in Fig. A1. The complex phase of  $\alpha$  may be chosen as zero since it has no physical meaning for a single qubit.

For nonvanishing  $\alpha, \beta$  the qubit is said to be a coherent superposition of the two distinguishable states associated with a classical bit. Note that this has no equivalent in the classical macroscopic viewpoint. When the qubit gets measured, i.e., evaluated, the amplitudes  $\alpha$  and  $\beta$  determine the probability of either outcome "0" or "1" by virtue of the Born rule:

$$P(\uparrow) = |\alpha|^2 \text{ and } P(\downarrow) = |\beta|^2. \quad (\text{A4})$$

Commonly, the probabilistic nature of the read-out process causes potential for confusion. However, note that a qubit is not simply a classical nondeterministic bit. Although such a nondeterministic bit has a certain probability to assume either value 0 or 1, it cannot be used to describe a system such as an electron spin. The space of all states of a nondeterministic bit may be visualized as the interval  $[0, 1]$  which is clearly not isomorphic to the hull of a sphere representing all states of

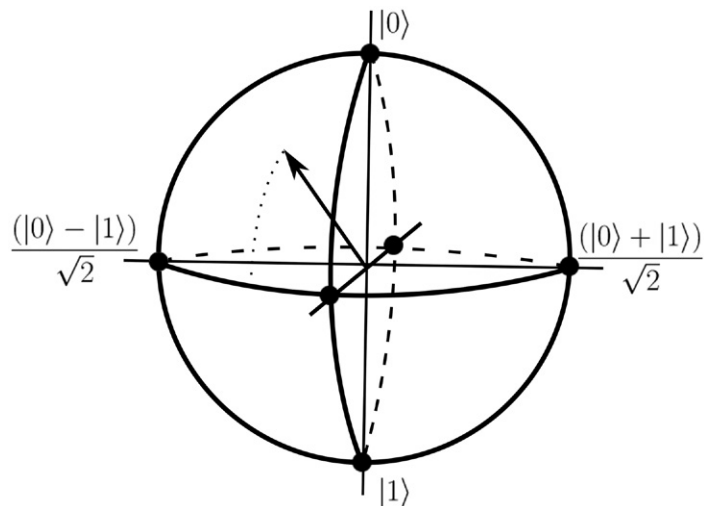


Fig. A1. Bloch sphere representing the set of all qubit states. The states  $|0\rangle, |1\rangle$  lie at Cartesian coordinates  $(0, 0, 1)$  and  $(0, 0, -1)$ , respectively. The states on the  $x$  axis have no classical analog.

a qubit. For example, a nondeterministic bit with  $P(0) = P(1) = 0.5$  cannot capture the notion of the two *physically* different states

$$|+\rangle \equiv \frac{|0\rangle + |1\rangle}{\sqrt{2}} \quad \text{and} \quad |-\rangle \equiv \frac{|0\rangle - |1\rangle}{\sqrt{2}}. \quad (\text{A5})$$

Although both states  $|+\rangle$  and  $|-\rangle$  give rise to the same probabilities for measuring 0 or 1 (i.e.,  $p = 0.5$ ), they behave very differently when manipulated further as will be seen in subsequent sections.

The general quantum state of  $n$  qubits can be described as an element in a complex vector space of dimension  $2^n$ . For instance, the general two-qubit state may be written as

$$|\Psi\rangle = c_1|0\rangle \otimes |0\rangle + c_2|1\rangle \otimes |0\rangle + c_3|0\rangle \otimes |1\rangle + c_4|1\rangle \otimes |1\rangle, \quad (\text{A6})$$

where  $\sum_i |c_i|^2 = 1$ . Note the outer products of single-qubit states forming the basis of the two-qubit state space.<sup>2</sup> Again, the probabilities of finding the two qubits in a particular state are determined by the Born rule. For instance, the joint probability of measuring the first qubit in the state 0 and the second in the state 1 is given by  $|c_3|^2$ .

<sup>2</sup> In the literature, the short form  $|ij\rangle$  is often used for  $|i\rangle \otimes |j\rangle$ , etc.

A crucial difference to classical bits can be observed when querying the state of a single qubit in a multiple qubit state. In a classical bit string, each bit has a well-defined value, e.g., the second bit of  $\bar{b} = \overline{1011}$  is in the state 0. Consider now the two-qubit state

$$|\Phi\rangle = \frac{1}{\sqrt{2}}(|0\rangle \otimes |0\rangle + |1\rangle \otimes |1\rangle). \quad (\text{A7})$$

No single-qubit states exist such that  $|\Phi\rangle$  can be written as a product state:

$$|\Phi\rangle \neq (\alpha_1|0\rangle + \beta_1|1\rangle) \otimes (\alpha_2|0\rangle + \beta_2|1\rangle) \quad (\text{A8})$$

$$= \alpha_1\alpha_2|0\rangle \otimes |0\rangle + \alpha_1\beta_2|0\rangle \otimes |1\rangle + \beta_1\alpha_2|1\rangle \otimes |0\rangle + \beta_1\beta_2|1\rangle \otimes |1\rangle. \quad (\text{A9})$$

More specifically, it is not possible to find complex numbers  $\alpha_1, \alpha_2, \beta_1, \beta_2$  such that  $\alpha_1\alpha_2 = \beta_1\beta_2 = 1/\sqrt{2}$  and  $\alpha_1\beta_2 = \alpha_2\beta_1 = 0$ . This is also known as the phenomenon of entanglement. The outcomes of measurements on both qubits of the state are perfectly correlated. Indeed, further investigation reveals that the amount of correlation exceeds the maximal classically possible amount of correlation.

## Appendix B: The quantum nonlinear solver

In this appendix we shall outline the key working mechanisms of the quantum nonlinear solver. For a detailed introduction the reader should consult Lloyd et al. (2020). Let us start by considering the integration of a linear differential equation system  $\dot{\mathbf{x}} = \hat{A}\mathbf{x}$  first, where  $\hat{A}$  is a constant operator acting on the  $d$ -dimensional vector  $\mathbf{x}$  of dynamical variables  $x_i$ . By discretizing the time domain and employing the forward Euler approach, we yield a set of linear equations:

$$\mathbf{x}[t_0 + (k+1)\Delta t] = \mathbf{x}(t_0 + k\Delta t) + \Delta t \hat{A} \mathbf{x}(t_0 + k\Delta t) \quad k = 1 \dots T. \quad (\text{A10})$$

For initial condition  $\mathbf{x}(t_0) = \mathbf{x}_{\text{ini}}$ , these equations can be cast in matrix form:

$$\begin{pmatrix} 1 & 0 & \dots & \\ -(1+\Delta t\hat{A}) & 1 & 0 & \dots \\ 0 & -(1+\Delta t\hat{A}) & 1 & 0 & \dots \\ \vdots & 0 & \ddots & \ddots & \end{pmatrix} \begin{pmatrix} \mathbf{x}_0 \\ \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \end{pmatrix} = \begin{pmatrix} \mathbf{x}_{\text{ini}} \\ 0 \\ 0 \\ \vdots \end{pmatrix}. \quad (\text{A11})$$

The history state vector  $(\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_T)$  encodes the discretized evolution of the dynamical variables. It can be derived by inverting the matrix on the left-hand side. Quantum computers can accomplish this task using the Harrow–Hassidim–Lloyd (HHL) quantum matrix inversion algorithm (cf. Harrow et al. 2009). This quantum algorithm provides an exponential speed-up over classical algorithms and therefore allows for a resource-efficient integration of linear differential equations.

To solve nonlinear differential equations, we generalize the above scheme to nonlinear forward Euler discrete steps. Consider a differential equation system  $\dot{\mathbf{x}} = \hat{f}(\mathbf{x})\mathbf{x}$ , where the operator  $\hat{f}$  is now a function of the dynamical variables  $\mathbf{x}$ . To implement  $T$  nonlinear forward Euler equations

$$\mathbf{x}[t_0 + (k+1)\Delta t] = \mathbf{x}(t_0 + k\Delta t) + \Delta t \hat{f}[\mathbf{x}(t_0 + k\Delta t)]\mathbf{x}(t_0 + k\Delta t), \quad (\text{A12})$$

we use mean-field nonlinearities. Take  $N$  identical copies of a (quantum) state  $X_k = \mathbf{x}(t_0 + k\Delta t) \otimes \dots \otimes \mathbf{x}(t_0 + k\Delta t) \equiv \mathbf{x}(t_0 + k\Delta t)^{\otimes N}$ . One can construct an operator  $\hat{F}$  acting on these  $N$  copies that upon projection on a single copy will reduce to the desired operator  $\hat{f}$  within order  $\mathcal{O}(\Delta t^2)$ . Consequently, the matrix inversion that will be carried out by the quantum computer reads:

$$\begin{pmatrix} 1 & 0 & \dots & \\ -(1+\Delta t\hat{F}) & 1 & 0 & \dots \\ 0 & -(1+\Delta t\hat{F}) & 1 & 0 & \dots \\ \vdots & 0 & \ddots & \ddots & \end{pmatrix} \begin{pmatrix} \mathbf{x}_0^{\otimes N} \\ \mathbf{x}_1^{\otimes N} \\ \mathbf{x}_2^{\otimes N} \\ \vdots \end{pmatrix} = \begin{pmatrix} \mathbf{x}_{\text{ini}}^{\otimes N} \\ 0 \\ 0 \\ \vdots \end{pmatrix}. \quad (\text{A13})$$

After inverting the matrix on a quantum computer, we yield a history state of  $N$  copies of the system state. The dynamical variables can then be evaluated by measuring individual copies.

## References

- Bharti, K., and Coauthors, 2022: Noisy intermediate-scale quantum algorithms. *Rev. Mod. Phys.*, **94**, 015004, <https://doi.org/10.1103/RevModPhys.94.015004>.
- Bouten, M., M. Guta, and H. Maassen, 2004: Stochastic Schrödinger equations. *J. Phys.*, **37A**, 3189–3209, <https://doi.org/10.1088/0305-4470/37/9/010>.
- Cerezo, M., and Coauthors, 2021: Variational quantum algorithms. *Nat. Rev. Phys.*, **3**, 625–644, <https://doi.org/10.1038/s42254-021-00348-9>.
- Epstein, E. S., 1969: Stochastic dynamic prediction. *Tellus*, **21**, 739–759, <https://doi.org/10.3402/tellusa.v21i6.10143>.
- Fleming, R. J., 1971a: On stochastic dynamic prediction: I. The energetics of uncertainty and the question of closure. *Mon. Wea. Rev.*, **99**, 851–872, [https://doi.org/10.1175/1520-0493\(1971\)099<0851:OSDP>2.3.CO;2](https://doi.org/10.1175/1520-0493(1971)099<0851:OSDP>2.3.CO;2).
- , 1971b: On stochastic dynamic prediction: II. Predictability and utility. *Mon. Wea. Rev.*, **99**, 927–938, [https://doi.org/10.1175/1520-0493\(1971\)099<0927:OSDP>2.3.CO;2](https://doi.org/10.1175/1520-0493(1971)099<0927:OSDP>2.3.CO;2).
- Harrow, A. W., A. Hassidim, and S. Lloyd, 2009: Quantum algorithm for solving linear systems of equations. *Phys. Rev. Lett.*, **15**, 150502, <https://doi.org/10.1103/PhysRevLett.103.150502>.
- Leyton, S., and T. J. Osborne, 2008: A quantum algorithm to solve nonlinear differential equations. *arXiv*, 0812.4423v1, <https://doi.org/10.48550/arXiv.0812.4423>.
- Lloyd, S., G. De Palma, C. Gokler, B. Kiani, Z.-W. Liu, M. Marvian, F. Tennie, and T. Palmer, 2020: Quantum algorithm for nonlinear differential equations. *arXiv*, 2011.06571v2, <https://doi.org/10.48550/arXiv.2011.06571>.
- Lorenz, E. N., 1963: Deterministic nonperiodic flow. *J. Atmos. Sci.*, **20**, 130–141, [https://doi.org/10.1175/1520-0469\(1963\)020<0130:DNF>2.0.CO;2](https://doi.org/10.1175/1520-0469(1963)020<0130:DNF>2.0.CO;2).
- McGeoch, C. C., 2020: Theory versus practice in annealing-based quantum computing. *Theor. Comput. Sci.*, **816**, 169–183, <https://doi.org/10.1016/j.tcs.2020.01.024>.
- Montanaro, A., 2016: Quantum algorithms: An overview. *npj Quantum Inf.*, **2**, 15023, <https://doi.org/10.1038/npjqi.2015.23>.
- Nielsen, M. A., and I. L. Chuang, 2010: *Quantum Computation and Quantum Information*. 10th ed., Cambridge University Press, 704 pp.
- Nimbe, P., B. A. Weyori, and A. F. Adekoya, 2021: Models in quantum computing: A systematic review. *Quantum Inf. Process.*, **20**, 80, <https://doi.org/10.1007/s11128-021-03021-3>.
- Palmer, T. N., 2001: A nonlinear dynamical perspective on model error: A proposal for non-local stochastic-dynamic parametrization in weather and climate prediction models. *Quart. J. Roy. Meteor. Soc.*, **127**, 279–304, <https://doi.org/10.1002/qj.49712757202>.
- , 2019: Stochastic weather and climate models. *Nat. Rev. Phys.*, **1**, 463–471, <https://doi.org/10.1038/s42254-019-0062-2>.
- , 2022: *The Primacy of Doubt*. Oxford University Press, 320 pp.
- Perelshtein, M. R., A. I. Pakhomchik, A. A. Melnikov, A. A. Novikov, A. Glatz, G. S. Paraoanu, V. M. Vinokur, and G. B. Lesovik, 2022: Solving large-scale linear systems of equations by a quantum hybrid algorithm. *Ann. Phys.*, **534**, 2200082, <https://doi.org/10.1002/andp.202200082>.
- Scherer, A., B. Valiron, S.-C. Mau, S. Alexander, E. van den Berg, and T. E. Chapuran, 2017: Concrete resource analysis of the quantum linear-system algorithm used to compute the electromagnetic scattering cross section of a 2D target. *Quantum Inf. Proc.*, **16**, 60, <https://doi.org/10.1007/s11128-016-1495-5>.