

Perspective

A molecular computing approach to solving optimization problems via programmable microdroplet arrays

Si Yue Guo,¹ Pascal Friederich,^{1,6,11} Yudong Cao,^{2,10} Tony C. Wu,¹ Christopher J. Forman,³ Douglas Mendoza,^{1,2} Matthias Degroote,¹ Andrew Cavell,⁵ Veronica Krasecki,⁵ Riley J. Hickman,¹ Abhishek Sharma,⁴ Leroy Cronin,^{4,*} Nathan Gianneschi,³ Randall H. Goldsmith,⁵ and Alán Aspuru-Guzik^{1,7,8,9,*}

SUMMARY

The search for novel forms of computing to the dominant von Neumann model-based approach is important as it will enable different classes of problems to be solved. Molecular computers are a promising alternative to semiconductor-based computers given their potential biocompatibility and cost advantages. The vast space of chemical reactions makes molecules a tunable, scalable, and energy-efficient computational vehicle. In molecular computers, memory and processing units can be combined into a single, inherently parallelized device. Here, we present a microdroplet array molecular computer to solve combinatorial optimization problems by employing an Ising Hamiltonian to map problems heuristically to droplet-droplet interactions. The droplets represent binary digits and problems are encoded in intra- and inter-droplet reactions. We propose two implementations: first, a hybrid classical-molecular computer that enforces inter-droplet constraints in a classical computer and, second, a purely molecular computer where the problem is entirely pre-programmed in the nearest-neighbor droplet reactions.

INTRODUCTION

For science and technology to continue progressing at the current pace, we need our computational processing capabilities to keep growing. Conventional transistor-based computers follow the von Neumann architecture, where information is stored in memory units and processed in a central processing unit. Different logic gates, each acting on a single bit or multiple bits of information represented as a 0 or 1, are activated sequentially to perform an operation. However, as transistors are approaching tens of nanometers, i.e., the size of large molecules,¹ high heat dissipation, and slow transfer rates between processors and memory are bringing about the breakdown of Moore's law.

The search for alternatives to classical computers includes quantum computing, in which quantum mechanical phenomena allow information to exist as a superposition of states, not just individual binary bits.^{2,3} For classical systems, computing with molecules is an attractive avenue given the vast chemical space and the relatively low energy dissipated in chemical reactions compared with transistors.^{4–6} Proposals for computers that exploit chemical processes have taken one or both of two broad

Progress and potential

Among the alternatives to conventional silicon-based computers, chemical reactions have the potential for high information density and massive parallelization while remaining accessible and cost-effective. Previous work in this field has focused on using chemistry and biology to emulate circuit components, such as logic gates, or to explore a combinatorial space in parallel. These approaches can be limited in their scalability, ease of programming, and applicability to a wide range of computational problems. The molecular computer presented here attempts to address these challenges by drawing inspiration from the Ising model for magnetism. Ideal for tackling combinatorial optimization, the device is an array of microdroplets subjected to pre-programmed droplet-droplet interactions that encode a given problem. This molecular computer offers the opportunity to use chemistry to overcome barriers in classical computers, such as high energy consumption, the von Neumann bottleneck, and the combinatorial explosion of computational problems.



approaches: (1) using chemistry and biochemistry to emulate circuit components or cellular automata, and (2) employing a large number of molecules to explore a combinatorial space in parallel. Examples for the first include reaction-diffusion systems,⁷ Belousov-Zhabotinsky oscillatory reaction,⁸ memristive polymers,⁹ and transcription regulation for cellular signaling,^{10,11} and other chemical and biochemical analogs of logic gates.^{12,13} In the second category of parallelized computing, we find microfluidic devices,¹⁴ nanofabricated networks,^{15–17} and adaptive amoebal networks.¹⁸ DNA computers have explored both approaches, from early work that involved sifting through combinations of nucleic acid sequences to recently using DNA self-assembly to carry out an algorithmic process.^{19–21} However, each method presents challenges in programmability, scaling, and maintaining accuracy in large calculations.

Here, we propose a new strategy for solving optimization problems using compartments of chemicals, such as droplets, spatially localized in a lattice and linked by a network of chemical reactions. Each droplet represents a binary variable that can communicate with its neighboring droplets. We encode a problem in the intra- and inter-droplet interactions. As the droplets' contents evolve following the thermodynamics, the system reaches a steady state where the ensemble of droplet states corresponds to a solution of the given problem. As we explain in the following section, [optimization by simulated annealing](#), our proposed droplet-based computer relies on an alternative approach inspired by the pioneering work of Kirkpatrick et al.,²² which employs simulated annealing to solve combinatorial optimization problems. The droplet array molecular computer functions as a heuristic Ising solver and has the benefit of intrinsically combining both memory and processing units in one device.

We ultimately envision a molecular computer that operates solely using chemical processes and without the aid of classical computers. This implementation, which we term the “purely molecular computer,” uses pre-programmable chemical couplings, corresponding to a given problem, that determines neighboring droplets' interactions. As a stepping stone to achieving complete autonomy, we have first conceived a “hybrid classical-molecular computer,” where a classical computer is used to enforce conditions that are then carried out by the droplet system. Both incarnations of the droplet array computer would be inherently parallelized, easily scalable, efficient, specific purpose computing devices.

Following the [description and requirements](#) for the molecular computer, we present the [modeling and application of the hybrid classical-molecular computer](#) and a description of the [purely molecular computer](#). In [implementation](#), we explore ways to physically realize our device, and finally, in [discussion](#), we consider the advantages of this approach and examine the scalability of the system.

CONCEPT

Optimization by simulated annealing

The basic principle of the droplet array computer rests in the analogy between combinatorial optimization and statistical physics.²² Similarly to the ensemble behavior of physical systems, such as magnetic spins, an optimization problem is a complex system of interacting variables. Such interactions are captured in a cost function or Hamiltonian, where the constraints and requirements of the problem correspond to interaction energies between variables. At a given temperature, the equilibrium distribution of the configuration (ups and downs) of the spins follows a

¹Department of Chemistry, University of Toronto, 80 St. George Street, Toronto, ON M5S 3H6, Canada

²Department of Chemistry and Chemical Biology, Harvard University, 12 Oxford Street, Cambridge, MA 02138, USA

³Department of Chemistry, Northwestern University, 2145 Sheridan Road, Evanston, IL 60208, USA

⁴School of Chemistry, University of Glasgow, Joseph Black Building, University Avenue, Glasgow G12 8QQ, Scotland, UK

⁵Department of Chemistry, University of Wisconsin-Madison, 1101 University Avenue, Madison, WI 53706, USA

⁶Institute of Nanotechnology, Karlsruhe Institute of Technology, Hermann-von-Helmholtz-Platz 1, 76344 Eggenstein-Leopoldshafen, Germany

⁷Canadian Institute for Advanced Research (CIFAR) Senior Fellow, Toronto, ON M5S 1M1, Canada

⁸Canada CIFAR AI Chair Lebovic Fellow, Vector Institute, Toronto, ON M5S 1M1, Canada

⁹Department of Computer Science, University of Toronto, 40 St. George Street, Toronto, ON M5S 2E4, Canada

¹⁰Present address: Zapata Computing Inc., Boston, MA 02110, USA

¹¹Present address: Institute of Theoretical Informatics, Karlsruhe Institute of Technology, Am Fasanengarten 5, 76131 Karlsruhe, Germany

*Correspondence:

lee.cronin@glasgow.ac.uk (L.C.),
aspuru@utoronto.ca (A.A.-G.)

<https://doi.org/10.1016/j.matt.2021.03.002>

Boltzmann distribution, which favors the ground state of the Hamiltonian or the optimal (set of) solution(s) to the objective function. As the temperature lowers, the statistical weight (probability) of the ground state grows, which inspires the idea of simulated annealing to emulate the effect of such an annealing process due to cooling for finding or approximating the ground state of the Hamiltonian.

A canonical model for magnetic spins in crystalline materials is the Ising or lattice spin Hamiltonian:

$$H_{\text{Ising}} = -\mu \sum_{i=1}^N h_i s_i - J \sum_{i<j}^N s_i s_j,$$

whose terms describe a tradeoff between the cost of flipping a spin $s_i \in \{-1, 1\}$ in an external magnetic field h and the interaction energy between neighboring spins, J , and where μ is the magnetic moment. For encoding general combinatorial optimization problems, however, we allow for more tunability between pairs of coupled spins as well as the local fields on each spin, giving rise to the more general Ising Hamiltonian:

$$H_{\text{general}} = \sum_{i=1}^N \alpha_i s_i + \sum_{i<j}^N \beta_{ij} s_i s_j,$$

in which the α and β coefficients are given by the problem.

In other words, the computational problem is encoded in the coefficients α_i , representing the local field for individual variables, and β_{ij} , representing the couplings between pairs of variables. For the given values of the vector α and matrix β , the optimal configurations of s_i represent one or multiple solutions that minimize the Hamiltonian.

The model can be generalized for higher-order terms (k -local), where the Hamiltonian is:

$$H = h^{(0)} + \sum_{i=1}^N h_i^{(1)} s_i + \sum_{i<j}^N h_{ij}^{(2)} s_i s_j + \sum_{i<j<k}^N h_{ijk}^{(3)} s_i s_j s_k + \dots$$

We shall proceed with this notation, where the $h^{(k)}$ is the tensor of rank k representing the coupling between k variables.

To implement the Ising model in a programmable computer, we must be able to control the local field and couplings (one-body and two-body terms) for a set of variables. While this approach for computation has been explored for various types of devices, as we survey below, in this work we propose that chemical reactions can be used to achieve these requirements, forming the basis of a molecular computer.

The Ising model paradigm for computation has inspired algorithm and hardware development for both classical and quantum devices. In [Table 1](#), we give an overview of these optimization annealing machines and their characteristics. The classical devices were chiefly based on traditional electronic components, such as a network of coupled LC (inductor-capacitor) oscillators,^{23–25} and conventional CMOS technologies, including the Fujitsu-led application-specific digital annealer²⁶ and the Toshiba simulated bifurcation algorithm for field-programmable gate arrays.^{27,28} Quantum annealing devices were argued to be superior to classical annealing due to the availability of quantum effects, such as superposition and tunneling.^{29,30} Notably, the D-Wave quantum annealer,^{31–33} using superconducting

Table 1. A survey of classical and quantum optimization Ising machines and their capabilities along with the chemical molecular computer presented in this work and anticipated future implementations

	Algorithm/approach	Hardware	No. of bits	Connectivity	Typical time to solution	Date	Optimization problem	Refs
Classical								
Oscillator-based Ising machine	subharmonic injection locking	LC oscillators	8 and 32 oscillators	6 (Chimera)	N/A	March 2019	MAX-CUT	²⁴
Coupled nonlinear oscillators	Euler-Maruyama method for SDEs	LC oscillators	4 oscillators	full	10^{-4} s	June 2019	MAX-CUT	²⁵
Fujitsu Digital annealer	digital annealing	application-specific CMOS	1,024 variables	full	$1-10^2$ s	April 2019	spin glass	²⁶
Toshiba simulated bifurcation	simulated bifurcation	FPGAs and GPUs	2,000 variables	full	5×10^{-4} s	April 2019	MAX-CUT (NP-hard)	^{27,28}
Quantum								
Coherent Ising machine	adiabatic quantum computation, quantum annealing	optical parametric oscillators; laser pulses	2,000 spins	full	5×10^{-3} – 5×10^{-2} s	May 2019	MAX-CUT, spin glass	^{34–36}
D-WAVE 2000Q Chimera	quantum annealing	superconducting qubits	2,000 qubits	6 (Chimera)	10^2-10^4 s	January 2017	numerous applications	³²
D-WAVE “next gen” Pegasus	quantum annealing	superconducting qubits	5,000 qubits	15 (Pegasus)	N/A	April 2019	numerous applications	³³
Chemical								
Hybrid classical-molecular computer	simulated annealing, stochastic gradient descent	droplet array + classical computer	12 droplets–10,000 droplets	full	10^3-10^5 s; to be reduced with smaller droplets ^b	2019	k-SAT, lattice protein	this work
Purely molecular computer	chemical annealing	droplet array	10s of droplets–10,000 droplets	4–6 ^a	10^3-10^5 s ^b		2-local k-SAT, 2-local lattice protein, TSP	
Long-term molecular computer	chemical annealing	droplet array	~1.5 million droplets ⁴⁰	4–6 ^a	N/A			
Inkjet printer	chemical annealing	microscopic nozzle (600 dpi)	~30 million droplets	4–6 ^a	N/A			

The time to solution (TTS) is the product of the average time for one run and the average number of runs required to solve a problem. The TTS depends on the technology and performance of the device and typically scales with the size of the problem (e.g., the number of problem variables).

FPGA, field-programmable gate array; N/A, not applicable.

^aThe connectivity is based on square and hexagonal lattices (see Figure S5).

^bThe time estimate for the molecular computer is based on preliminary experiments of polymerization in microdroplets, where one run requires ~10 min³⁸ (although we expect this time to vary significantly with the choice of chemical reaction and droplet size) and a scaling of $\exp(b\sqrt{N})$ for the number of runs.³⁹

qubits, and the Coherent Ising Machine,^{34–36} implemented with laser pulses, have shown promise in the number of qubits and speed of calculation. Recently, bifurcation-based quantum annealing was proposed as an alternative to standard quantum annealing.^{37–40}

Adiabatic quantum computing and quantum annealing provide us with a method of solving for the Ising ground state, by evolving a system from the ground state of an accessible problem to that of a more difficult problem.^{29,30,41} Although a molecular computer remains a classical device, unable to avail itself of quantum tunneling across barriers in the potential energy surface relating to the problem Hamiltonian, the rationale can still be extended to simulated annealing.

In adiabatic quantum optimization, a system is initially in the ground state of a Hamiltonian H_0 . To solve for the ground state of a problem Hamiltonian H_p , the Hamiltonian H is adiabatically changed over time following a function $a(t)$:

$$H(t) = [1 - a(t)] H_0 + a(t) H_p.$$

For simulated annealing, H_0 represents kinetic or thermal energy and H_p the potential energy, corresponding to the problem we would like to solve. Just like annealing metals, as the temperature of a system decreases, the spins or variables reach an optimal configuration.

For any classical optimization algorithm that can be mapped to the classical Ising system, the time to solution to find the ground state asymptotically grows at most exponentially in n , i.e., $O(c^n)$, as it is needed to explore all possible outcomes, and finding an approximate solution grows as polynomially in n , i.e., $O(n^\gamma)$, where γ is a problem-dependent constant.⁴¹ By the Church-Turing thesis,⁴ the molecular computer should not have any exponential speedup over any classical computing devices.

Simulated annealing was found to be well suited to tackle NP and NP-hard combinatorial optimization problems.⁴² These are problems whose computational complexity increases exponentially with the number of variables, and for which no efficient (polynomial-time) algorithm is presently known. Nevertheless, since solving the Ising model was shown to be NP-hard, a good heuristic Ising model solver will be beneficial for solving real-world problems that are often NP-complete, such as the traveling salesperson problem, graph coloring, and Boolean satisfiability.⁴³ (NP-complete problems belong to both NP and NP-hard classes. They are defined as decision problems where a polynomial-time reduction exists from every other problem in NP.⁴⁴) Barahona showed that solving the states of the Ising model⁴⁵ on a two-dimensional (2D) lattice with arbitrary interactions is NP-complete. Since every instance of a problem in NP can be efficiently encoded as an instance of an NP-complete problem, Barahona's work⁴² implies that all problems in the NP complexity class can be mapped to instances of 2D Ising model, and thus can potentially be solved efficiently with a heuristic Ising solver. The work of Barahona and others,⁴⁶ summarized in Table 2, also includes the complexity of solving different Ising models.

Description and requirements

As described earlier, the proposed molecular computer consists of an array of droplets arranged in a lattice, which is shown in Figure 1. To apply the concept of adiabatic quantum optimization described above to the molecular computer, we begin with the random initialization of the droplet states. We can program the

Table 2. The complexity of the Ising ground state problem in one, two, and three dimensions

Dimension and graph	Restrictions on coupling	Solving for the Ising ground state
1 D	none	P
2 D (planar graph)	$h_i = 0$	P
	$h_i \neq 0$	NP-hard
Non-planar graph	$J_{ij} \in \{-1, 0, 1\}$	NP-hard

problem Hamiltonian in the computer by setting the local field terms h_i and couplings J_{ij} . The droplet array molecular computer enforces these parameters so that, as the system anneals, it reaches the ground state of the problem, which can be read out and interpreted. Since it is not a quantum computer, the system will not tunnel between local minima in the potential energy surface that is associated with the problem Hamiltonian. Instead, we will require multiple initializations to attain the global minimum, thus solving a given problem.

To achieve molecular computing by annealing as described above, the chemistry taking place in the droplet array must fulfill several requirements. First, the system should exhibit physical properties that can be manipulated, read, and mapped as two states, so that each droplet can correspond to a binary variable. Second, changes of the states must be reversible and controllable by the state of their neighboring droplets. Third, to implement interactions between the states that correspond to coupling terms of the Ising model, there must be connections between the droplets that allow for programmable positive and negative coupling interactions.

We propose two approaches to implementing such a system: first, a hybrid classical-molecular computer in which an array of droplets is connected to a classical computer; and second, a purely molecular computer. In [Table 1](#) we compare the two approaches and give the roadmap to scale to millions of droplets, as is currently achievable with inkjet printers at 600 dots per inch (dpi) resolution.⁴⁰

In the hybrid classical-molecular computer, measurements of each droplet's state will be passed to a classical computer software that computes the interaction based on a given problem Hamiltonian and outputs instructions for the droplet array to implement the calculated coupling, using, e.g., electrodes or optical excitations (see [implementation](#)). In the case of the purely molecular computer, the coupling between nearest-neighbor droplets is pre-programmed in physical or chemical interaction between the droplets that take the form of mass or energy exchange (see [purely molecular computer](#)).

The reversibility of the chemical reaction is essential to maintaining the symmetry of the problem. For instance, two droplets that are positively coupled to each other have a coefficient $J > 0$. The contribution to the Hamiltonian $-J s_1 s_2$ is minimized when the two droplets are in the same state, either both 0 or both 1, which means both reactions $0 \rightarrow 1$ and $1 \rightarrow 0$ must be achievable. (This symmetry is broken as interactions with other droplets are taken into account, leading to possible frustration as they all compete with each other.)

We must emphasize that the molecular computer does not have *a priori* knowledge of the solution to the problem. It merely imposes the problem Hamiltonian on the droplets via a local field and droplet couplings. As the chemical system evolves under these conditions, it will attempt to minimize its thermodynamic free energy, which is the physical

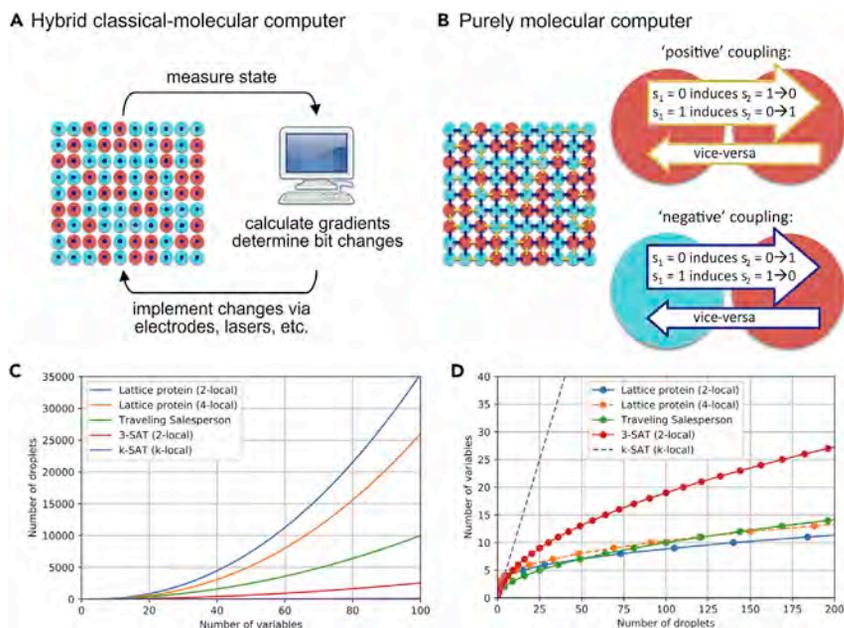


Figure 1. Schematic of the droplet array-based molecular computer

(A) A hybrid classical-molecular computer in which measurements of droplet states are sent to a computer that calculates the gradient on each droplet and directs the changes to be implemented *in situ*.

(B) A purely molecular computer where the droplets interact via nearest-neighbor couplings, either “positive” or “negative”. The couplings are programmed into the device in the initiation stage.

(C) The scaling of the number of droplets for different optimization problems: lattice protein folding (quadratic/2-local and quartic/4-local forms), traveling salesperson, Boolean satisfiability (3-SAT reduced to 2-local and k-SAT k-local).

(D) The number of variables in these problems that a given droplet array size can address.

impetus to explore distinct solution states associated with the problem Hamiltonian, imposed by the couplings. Depending on the given problem, it may have degenerate ground states corresponding to multiple optimal configurations.

We expect that the kinetics of the problem, as well as imperfections in the settings of the molecular computer, will result in trapping in local minima for particular experiments, requiring the repetition of the computation to sample from the low-energy states of the problem. The alternative approaches mentioned earlier, namely the simulated annealing, quantum annealing, and simulated bifurcation algorithms, share this challenge.^{25,27}

MODELING AND APPLICATION OF THE HYBRID CLASSICAL-MOLECULAR COMPUTER

Tuning the intra- and inter-droplet chemical reaction to satisfy the requirements described above is not a trivial task. To test chemical systems for the droplet-based computer, we propose first developing a hybrid classical-molecular computer. A hybrid device also has the benefits of non-local interactions, full connectivity between all droplets, and higher-order couplings that involve k droplets (k -local interaction).

Our proposed implementation of a hybrid classical-molecular computer uses an array of microdroplets containing various chemical compounds. Manipulation of droplets can be achieved with electrostatic fields and electrochemical transformations induced by electrodes or optical stimuli induced by light sources. The change

in the states of droplets should be detectable by optical means, which allows for a convenient readout of the states. For the sake of simplicity in this concept paper, we use an abstract nomenclature of states 0 and 1. We assume that the droplets fluctuate within this range of states as a result of external stimuli controlled by a classical computer. We discuss possible experimental schemes to build such a system in [purely molecular computer](#). One can think of the value associated with each droplet as the droplet's progress along an arbitrary reaction coordinate.

To explore the capabilities and potential application areas of the hybrid classical-molecular computer, we implemented an *in silico* model of it. The model is composed of two parts, a set of readable and writable states that represents the droplet array in the molecular computer and a computer algorithm that, given a model Hamiltonian, calculates and applies stepwise changes to these states. The model are described below in [simulation by stochastic gradient descent](#).

We applied the *in silico* model of the hybrid classical-molecular computer to two applications that we discuss in [application 1: Boolean satisfiability](#) and [application 2: lattice protein model](#). The first application is Boolean satisfiability (in particular 3-SAT), which was proven to be NP-complete.⁴⁷ The second application is a simplified protein model consisting of six types of amino acids on a 2D square lattice. To solve these problems using the molecular computer, we map them to Ising Hamiltonians, encode the coupling elements of the Ising Hamiltonian in the couplings between the droplets, and initialize the droplets in random states. Annealing of the system leads to a local optimum, from which a solution can be read out and checked on validity. Repetition of this procedure allows us to determine if a Boolean satisfiability problem has a solution that fulfills all conditions or that identifies low-energy conformations of the protein. An extensive list of mappings from NP problems to the Ising Hamiltonian is presented in Lucas.⁴⁵

Simulation by stochastic gradient descent

We modeled the annealing process of the hybrid classical-molecular computer using a stochastic gradient descent simulation. As shown in the supplemental information (sections A and B), it is possible to convert Boolean satisfiability problems, as well as the lattice protein folding model, to an Ising model that can be represented in the form of tensors $h^{(0)}, h^{(1)}, \dots, h^{(n)}$. The first term, $h^{(0)}$, is an offset to the global energy that does not influence the states of the system, $h^{(1)}$ is a vector of bias terms to the states of each droplet, $h^{(2)}$ is a matrix of two-droplet coupling terms, $h^{(3)}$ is a tensor containing three-droplet coupling terms, and so on. The 3-SAT problem can be converted to an Ising model with up to three-body interaction terms, while the lattice protein model contains up to four-body interactions. As discussed earlier, the hybrid classical-molecular computer can handle interactions of any order (requiring n droplets for n -body terms), while the purely molecular computer will be limited to nearest-neighbor two-droplet interactions due to the limited connectivity between the droplets. It is, however, possible to convert higher-order interactions, such as three- and four-droplet to two-body interactions by introducing ancillary states in the system.⁴⁸ While our model of the hybrid classical-molecular computer can process any order of many-body interactions, we used the two-body representation of the lattice protein model that includes ancillary states.

In the simulations of the hybrid classical-molecular computer, we have the possibility to run the procedure as a mixture of two extreme modes, an ensemble annealing mode and a trajectory mode. In the ensemble annealing mode, we run many short and independent annealing simulations from random initial states and analyze the statistics of all final states to search for the global optimum as well as the distribution

of local optima. In contrast, in trajectory mode, we run one instance of the molecular computer over a large number of steps and add a certain level of noise (“temperature”) on the gradients to overcome barriers between local optima. In practice, we are using a mixture of both modes to obtain enough statistics with limited computational time, i.e., we run multiple instances of independent trajectories with a medium noise level that is high enough to overcome small barriers and set a convergence criterion to end the trajectories and record the results. A systematic optimization of the hyperparameters (number and length of trajectories, noise level, convergence criterion, etc.) is beyond the scope of this work.

Application 1: Boolean satisfiability

To solve Boolean satisfiability problems, such as the 3-SAT problem, with our computational model of the hybrid classical-molecular computer, it is necessary to convert it from its conjunctive normal form to an Ising model (see [supplemental information](#) section A for a description of one mapping and section B for an example on a four-droplet system). In the case of 3-SAT problems, the Ising model will contain up to three-body coupling terms, which requires connectivity that goes beyond that of a purely molecular computer. We evaluated the satisfiability of random 3-SAT problems with N variables and M unique clauses of $k = 3$ literals and compared the results with solutions obtained using the Davis-Putnam-Logemann-Loveland (DPLL) algorithm⁴⁹ (see [Figure 2A](#)). The results are shown in [Figure 2B](#), where we observe a transition from a high probability of a problem being satisfiable when the clause density (the ratio of M/N) is small to a low probability at large M/N ratios. The results obtained with the simulation are in good agreement with the results obtained using the (exact) DPLL algorithm.

Application 2: Lattice protein model

As a second application, we applied the simulated hybrid classical-molecular computer to the 2D lattice protein model presented in Perdomo-Ortiz et al.⁵⁰ and Babbush et al.⁵¹ The proteins in this model consist of a sequence of amino acids that can fold onto a 2D square lattice. The protein conformation is represented as a sequence of turns, where each turn is encoded by two bits (00, down; 01, right; 10, left; 11, up; see [Figures 3A](#) and [3B](#) for a description and an example of a folding).

The energy of each conformation contains two terms: clashes of two amino acids are penalized with positive terms of size λ , whereas attracting interactions between specific amino acids (i and j , $|i - j| > 1$) are rewarded with negative energy terms ϵ_{ij} in cases where amino acids i and j are direct, non-diagonal neighbors (native contact). We can construct an Ising Hamiltonian with up to four-body interactions that encodes both types of interactions. This requires ancillary bits that activate and deactivate depending on the state of the bits that encode the physical conformation of the protein. The overlap parameter λ can be chosen in a way that the energy spectrum of the protein model and thus of the Ising Hamiltonian has negative or zero energies for all conformations without clash and positive energies for all other conformers. The global ground state of the Hamiltonian yields the protein conformation in which the highest possible number of native bonds are formed. Non-clashing conformers with fewer or less strong native bonds are local optima with energies smaller than zero, whereas non-clashing conformers without any native bond formed have zero energy (“unfolded”).

As described in [simulation by stochastic gradient descent](#), the simulated hybrid classical-molecular computer is capable of solving Ising models with four-body interactions, whereas any purely molecular computer will only have local connections that can encode up to two-body interactions. There, the four-body Ising model of the protein was converted to a two-body Ising model, which requires the introduction

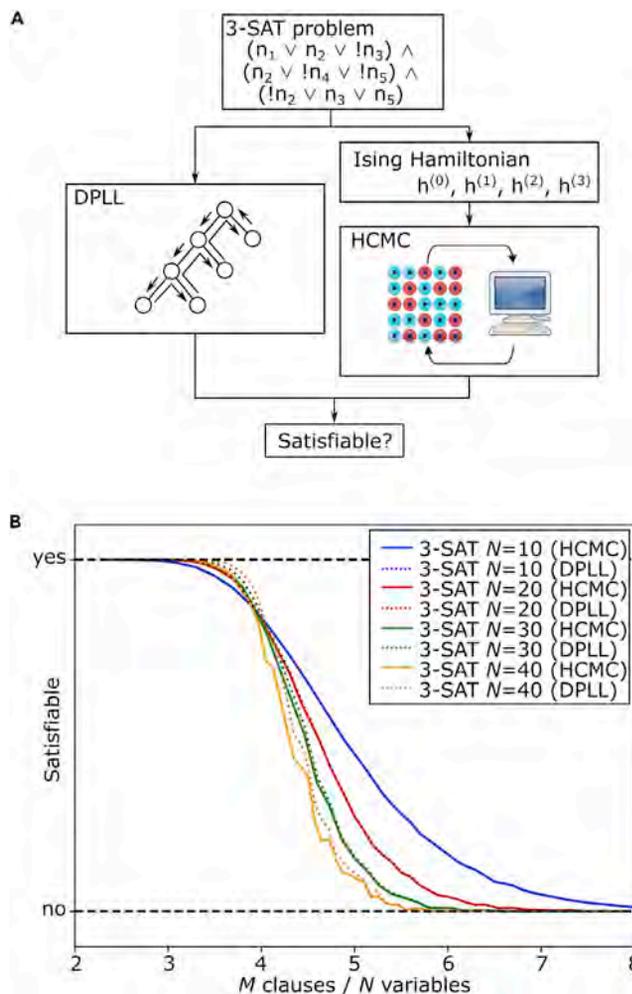


Figure 2. Comparison of the simulated hybrid classical-molecular computer to an exact algorithm for Boolean satisfiability

(A) Illustration of two methods to solve the 3-SAT problem: the exact Davis-Putnam-Logemann-Loveland (DPLL) algorithm and the simulated hybrid classical-molecular computer (HCMC).

(B) Probability of satisfiability as a function of clause density (ratio of clauses and variables) for a random 3-SAT problem, showing agreement between the HCMC (solid lines) and the solutions of the DPLL algorithm (dotted lines). Each point is the average of up to 10 000 3-SAT problems.

of further bits (“reduction bits”). This procedure is described in Davis et al.⁴⁹ The results of annealing simulations of the lattice protein model are shown in Figure 3C. Selected trajectories that ended at low-energy conformations show the energy as a function of the step number. Formation of native contacts, as well as annealing of the ancillary bits, reduce the total energy of the system. The final states shown in Figure 3C are local optima of the Hamiltonian. The barriers between these local optima are considerably higher than the energy differences between the local optima (further details are provided in section C of the supplemental information), which makes it impossible for a (stochastic) gradient descent algorithm to overcome the barrier between local and global optima. To find the global optimum and, thus, the ground state of the protein, it is necessary to run many instances of the molecular computer. No speedup can be expected compared with an algorithm that searches through all 2^K states of the protein, with K being the number of bits that encode the physical conformation of the protein ($K = 7$ in our example).

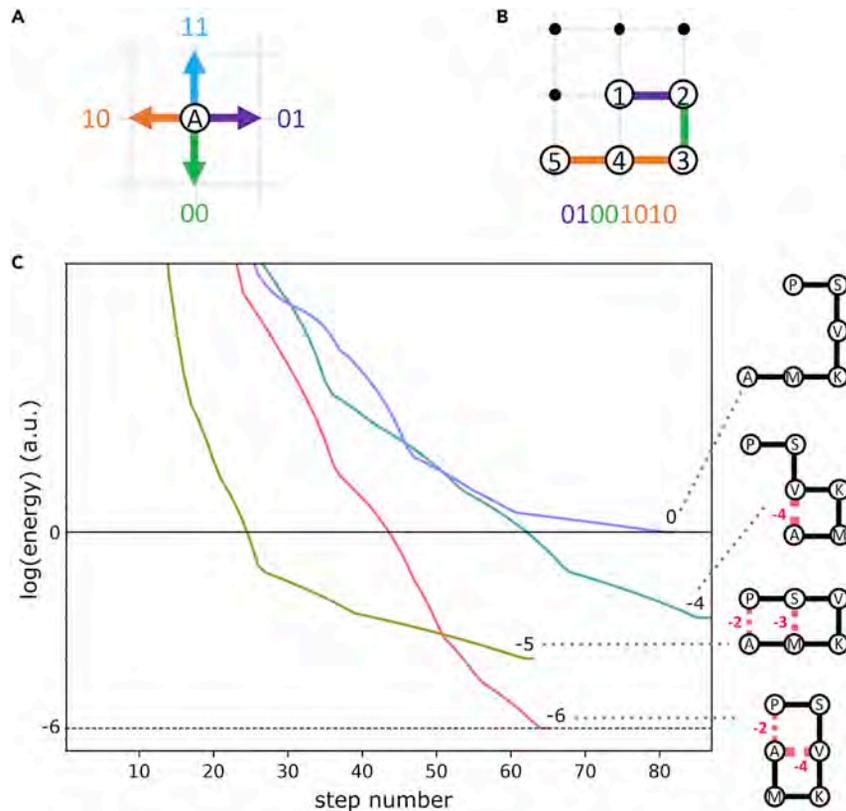


Figure 3. Lattice protein model simulated on the molecular computer

(A) Encoding of each link in the amino acid chain as two bits, (B) an example five-link chain with the corresponding sequence of bits, (C) results of a simulation for a six-link chain, where each curve corresponds to an instance of random initialization followed by annealing to a local minimum, with the corresponding folded configuration. The ground state configuration, with an energy of -6 , is shown in the lower right corner.

PURELY MOLECULAR COMPUTER

Ultimately, we envisage a purely molecular version of the droplet array computer that anneals to a ground state configuration solely by the physical and chemical interactions between droplets. In the absence of an external classical computer to enforce droplet-droplet couplings, the problem must be pre-programmed into the contents of the droplets and the interactions of neighboring droplets. These programmable couplings can take place via mass exchange (diffusion, biological membranes) or energy exchange (excitons, Förster resonance energy transfer [FRET]^{52,53}).

Such a system benefits from the complete parallelization of the problem. For a quadratic unconstrained binary optimization (QUBO) problem, a hybrid computer still requires that, at regular intervals representing a step of the optimization, a classical computer calculates up to $n(n-1)/2$ terms representing the pairwise energies of the n variables, and even more for problems with higher-order terms. A purely molecular computer would implement these couplings $O(n^2)$ times faster since all pairwise interactions occur simultaneously, in $O(1)$ time.

Rather than using a classical computer to perform stochastic gradient descent, as we did in the hybrid version of the device, the system is driven, kinetically and thermodynamically, to its ground state. The key is to select the appropriate

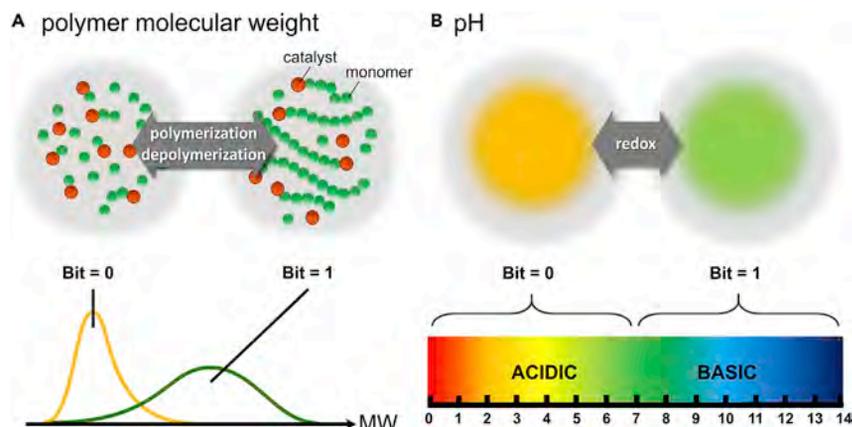


Figure 4. Chemical reactions to implement the droplet states

Two approaches to representing a droplet as a binary variable: in (A) polymerization and depolymerization giving different molecular weight distribution, and in (B) redox reactions altering the pH of the droplet.

reactions and chemicals. A few proposed mechanisms are detailed in [Implementation](#).

While the hybrid computer does not make use of the spatial arrangement of the droplets, in the purely molecular version, each droplet is connected only to those droplets with which it can physically interact. A problem must therefore be mapped to this graph using minor-embedding techniques to address the sparse connectivity.^{54,55}

In addition, this device can only implement two-body terms, which means auxiliary ancilla droplets are needed for reducing k -local problems (PUBO) to a QUBO. Such algorithms are commonly used to map problems to various quantum devices, such as the D-WAVE quantum annealer and the Rigetti quantum computer.^{39,48}

$$H = c + \sum_{i=1}^N h_i s_i + \sum_{i < j}^N J_{ij} s_i s_j$$

In the following section, we discuss practical considerations for building a physical device.

IMPLEMENTATION

Given the requirements above for the hybrid and purely molecular computers, numerous chemical systems can be considered as viable candidates. A physical system capable of implementing either the hybrid classical-molecular computer or the purely molecular computer should have properties that can be precisely defined and measured. While these measurements often lie on a continuum, they can be used to assign a binary state to a droplet using, for example, a threshold value. Since each droplet will be subjected to changing constraints from neighboring droplets (two-body terms, $h^{(2)}$) as the computation progresses, changes in these properties must be reversible so as to enable droplet states to flip between 0 and 1.

[Figure 4](#) gives two examples of inter-droplet reactions and the respective physical properties that can be represented as a binary state. In [Figure 4A](#), the contents of the droplets undergo polymerization and the resulting polymer weight distribution is used to determine the droplet state. Recent experiments have demonstrated the ability to optically monitor the extent of polymerization in microdroplets.³⁸

(Section D of the [supplemental information](#) details a simple, idealized model of a system based on polymerization within droplets.) Another versatile property to form the basis of computing bits for the hybrid classical-molecular computer is the pH of a droplet, as illustrated in [Figure 4B](#). The pH of a droplet can be reversibly adjusted using several chemical reactions, such as addition of acid or base or by tuning electric potentials to drive redox reactions that generate or consume H^+ .

Other candidate properties include the concentration of a reagent, oxidation states, and color. The state of each droplet can be read out in a non-interfering manner via optical or electrochemical means. For example, pH-sensitive dyes or ion indicators can be used to read out a fluorescence signal proportional to the pH or ion concentration in each droplet in the array.⁵⁶ Functionalized electrode arrays can be used to address individual droplets and determine their pH. Fluorescent dyes sensitive to gelation or viscosity changes can be used to indicate the presence of polymer.⁵⁷

In the hybrid classical-molecular computer, the inter-droplet couplings, whose nature and strength correspond to the problem being optimized, are enforced by an external control algorithm. In the purely molecular computer, these couplings should be pre-programmed into the inter- and intra-droplet interactions. Information between neighboring droplets can be shared via mass transfer or energy transfer. The former relies on the movement of molecules across the droplet boundaries, which can take place through passive diffusion across the interface, by electrophoretic motion, or via pores installed in the membranes to enhance the rates of transport between droplets.^{58,59} The latter energy transfer-based system could eliminate these considerations by allowing for inter-droplet communication without mass transfer, such as via FRET. (See [supplemental information](#) section E for an example showing how bits can be mapped to both a hybrid classical-molecular computer and a purely molecular computer.)

To execute either version of the molecular computer, we must achieve precise control over droplet placement. Microfluidic-based robotic platforms, as well as inkjet printers, can be used to generate droplets with precise compositions and to place them in square or hexagonal packed arrays. In this way, the droplets can be situated on electrode arrays for external control of droplet couplings or the read out of droplet states. The states of the droplets can be determined via optical or electrochemical readout taken at regular intervals using the means mentioned above. For the hybrid classical-molecular computer, this readout can be sent to a classical computer that can implement the necessary couplings. The classical computer can perturb the droplets through external stimuli (optical and electrochemical control) based on the set couplings. This control loop would continue until the problem reaches a minimum energy state. For the purely molecular computer, the couplings are pre-programmed into the droplet interactions, removing the need for a classical computer to impose these couplings.

DISCUSSION

Problem scaling analysis of the molecular computer

An essential factor in the choice of problems to solve on the microdroplet array computer is how the droplet array size scales with the number of variables in the problem. Compared with quantum annealers, which are constructed using Josephson junctions cooled to near absolute zero temperature, the cost of adding droplets to the molecular computer is negligible compared with the overall cost of the setup.

Since the molecular computer takes Ising parameters as input, any problem that can be efficiently encoded in a spin Ising system can also be efficiently implemented on a

microdroplet array computer. As far as we know, the parallelization of chemical reactions across all droplets could give the molecular computer an advantage over numerical gradient evaluations in a classical computer or GPU.

In Figures 1C and 1D, we plotted the scaling behavior of three types of problems that can be solved on either the hybrid or the purely molecular computer.⁴⁵ The problems include the two examples given in the main text, Boolean satisfiability (SAT) and lattice protein folding, as well as the traveling salesperson problem (TSP), an important logistics optimization problem in which we seek the shortest path between a set of points (e.g., Uber driver scheduling passengers' pickup and drop-off). Recall that the hybrid version of the molecular computer is fully connected and allows for higher-order expressions involving k variables, or k -local terms. The purely molecular computer, on the other hand, is limited to two-body (2-local) nearest-neighbor connections. Scaling relations must take into account the locality reduction algorithms and ancilla droplets required to overcome this limitation and solve any general problem.

On the hybrid computer, Boolean satisfiability for k variables per clause (k -SAT) scales linearly with the problem size, as $O(n)$, since each variable corresponds to one droplet and all droplets are connected to each other via the control hardware. For three or more clauses $k \geq 3$, the problem is NP-complete. To reduce a 3-SAT problem to a 2-local Ising Hamiltonian for the purely molecular computer, $(n-1)^2/4$ ancilla droplets are required.⁴⁸ For the traveling salesperson problem involving n points or nodes, n^2 droplets are needed to label the sequence of points visited, thus the number of droplets scales as $O(n^2)$.⁴⁵ Since the Ising Hamiltonian for this problem comprises only quadratic terms (2-local), it can be implemented on the purely molecular computer without locality reduction. Finally, the lattice protein problem is the most complex of our examples. To find the minimum energy 2D structure for a string of n amino acids, the 4-local Hamiltonian must encode interaction energies between neighboring amino acids and self-avoidance, scaling as $n^2 \log(n)$.^{50,51} We note that the scaling of the described problems is at most polynomial in the length of the input size. Also, it has been shown that embedding QUBO problems on a sparsely connected graph scales linearly with the number of variables in the problem.³⁹

As the number of variables in a problem increases, so does the number of droplets and the expectation that errors will accumulate as a result of noise, control, and readout. Albash et al.⁶⁰ recently examined the errors in analog Ising machines, which scale as power-law with problem size. While classical bit repetition schemes could be used for molecular computers, we can also draw inspiration from quantum annealing correction schemes, such as those employed by D-WAVE.^{61,62} One such scheme is the nested quantum annealing correction which makes C copies of each bit and increases the couplings by C^2 .⁶¹ Once the device has performed the computation, majority vote over the C copies of each bit determines its final state. While error correction schemes ensure a higher probability of reaching the ground state solution, it does involve increasing the number of bits, or droplets, in the molecular computer.

Advantages of the molecular computer

With a fully realized molecular computer, there will be many advantages compared with conventional silicon-based computers, as well as compared with many other recent proposed computing architectures. If some of the proposed mechanisms in previous sections could be realized, we believe that a molecular computer to be particularly well suited to solving combinatorial optimization problems, including some NP problems.

Comparison with classical computation: since the rules of classical physics still govern the operation of a molecular computer, we do not believe it will be more efficient than classical conventional computers in terms of time or space scaling. There are nevertheless many advantages to using chemical reactions for solving specific types of problems, such as discrete optimization. The approach described here removes the need for physically defined, atomic-sized circuits. The molecular computer benefits from the intrinsic parallelization of chemical reactions, effectively solving a system of differential equations of motion in parallel and eliminating the storage in memory of each step of the optimization process. Molecular computers also offer the possibility of reducing the energy consumption and heat dissipation of computing, two issues that affect the performance of computer chips. Inspired by biological processes, neuromorphic and biochemical computers were found to be 1,000 times more efficient than classical computers, in part due to their inherent massively parallelized processes.^{15,63,64} For the molecular computer described here, which also benefits from mass parallelization and a built-in memory, using chemical reactions for the bit interaction to replace multi-gate solutions potentially reduces the operation energy usage. A promising classical, CMOS-based alternative are digital annealers; while they use a similar approach and algorithms to those of the molecular computer described here, digital annealers would also be affected by the aforementioned issues. In molecular computers, the computation speed is, however, limited by the intrinsic speed of droplet motions and chemical reactions, nuclear rearrangements being orders of magnitude slower than electronic rearrangements in the Born-Oppenheimer approximation.

Comparison with biomolecular computation: the droplet array molecular computer also has advantages over biomolecular methods, such as DNA computing and cellular signaling, with faster operation speed, fewer errors, and significantly less complexity than biological cells. There is no need for post-processing or offline analysis of the chemical content, allowing for immediate readout. Droplet array initialization, computation through annealing, and optical readout are all fast and automatable processes. The phase space of chemical reactions, mechanisms, and reagents, is extensive. The contents of each droplet can be cost-effective, widely available, and safe materials.

Comparison with quantum computation: in recent years, quantum computing has gained much attention due to the potential for solving problems that are intractable for classical Turing machines. These include cryptography (prime factorization), quantum chemistry, as well as the optimization problems described above. However, with quantum entanglement and superposition come challenges in error correction, noise, decoherence, and qubit scaling, which must be overcome before quantum computers attain their full potential. The droplet array, on the other hand, is easily scalable through inkjet printing. The cost of adding additional droplets, representing bits, is negligible compared with the cost of the overall experimental setup.⁴⁰ Also, there is little experimental overhead to operate the molecular computer, since we can select for chemical reactions that take place at or near room temperature. Information storage would also be trivial since the output of a calculation can be printed onto a piece of paper for future readout.

CONCLUSION

In sum, we have proposed a new heuristic method for computation with programmable droplet-arrays to solve combinatorial optimization problems. The device consists of a 2D array of microdroplets that represents a set of interacting binary

variables that evolve under an Ising Hamiltonian. Each droplet corresponds to one variable, whose value is determined by measuring a specific property of the droplet. A specific problem is solved by programming the intra-droplet contents and inter-droplet interactions. As the system evolves collectively, the droplet states approach the optimal solution of the problem through a process akin to annealing in materials.

D-WAVE first adopted a quantum version of this approach; more recently, a classical digital annealer was introduced by Fujitsu. To our knowledge, this is the first proposal of a molecular computer operating in annealing mode. In its purely molecular version, the microdroplet array computer benefits from all of the advantages of computing with molecules: concurrent information processing and storage, massive parallelization of chemical reactions, energy-efficient processes, vast phase space of molecules and reactions, cost-effectiveness, and scalability of the device.

As a stepping stone to a purely molecular computer, we first developed a hybrid model where a classical computer imposes the parameters of the optimization problem, and the information processing and storage is carried out by the individual droplets in the array. Throughout numerous iterations, the classical computer takes in the set of droplet states and performs a stochastic gradient descent algorithm to search for the optimal ground state using the droplets. A simulation of the hybrid classical-molecular computer demonstrated its ability to solve two NP-hard problems, reproducing the phase transition in Boolean satisfiability (3-SAT) as a function of clause density, and identifying the ground state configuration in a lattice protein folding problem.

Our next step is to perform these calculations on a physical hybrid classical-molecular computer. The challenge of identifying suitable chemical reagents and reactions to program the microdroplet array can be facilitated with machine learning and high-throughput experimentation. We would also employ robotics and computer vision to operate the device. In due course, we shall tackle droplet miniaturization to scale up to thousands and eventually millions of droplets. In the ultimate limit of this technology, we imagine a molecular computer operated by inkjet printing a problem onto a sheet of paper; by the time the ink dries, the problem is solved and imprinted onto the sheet. Applying the same concept to a 3D printed ink may also open a novel route for large-scale assemblies of matter.

Data and code availability

The code for the hybrid classical-molecular computer simulation, applied to the Boolean satisfiability and lattice protein folding problems, is available at https://gitlab.com/pascal_friederich/molecularcomputer.

SUPPLEMENTAL INFORMATION

Supplemental information can be found online at <https://doi.org/10.1016/j.matt.2021.02.020>.

ACKNOWLEDGMENTS

We are grateful to the Defense Advanced Research Projects Agency (DARPA) for funding this project under award no. W911NF-18-2-0036 from the Molecular Informatics program. P.F. acknowledges funding by the European Union's Horizon 2020 Research and Innovation Program under the Marie Skłodowska-Curie grant agreement no. 795206 (MolDesign). L.C. thanks the ERC for an Advanced Grant (Smart-POM) and the EPSRC for funding grant EP/R009902/1. A.A.-G.

acknowledges the generous support from Anders G. Frøseth. R.J.H. is grateful to the Natural Sciences and Engineering Research Council of Canada (NSERC) for provision of the Postgraduate Scholarships-Doctoral Program (PGSD3-534584-2019). Computations were performed on the Niagara supercomputer at the SciNet HPC Consortium with support from Compute Ontario and Compute Canada.

DECLARATION OF INTERESTS

The University of Toronto has filed an application for a US patent based on the technology described in this paper, naming S.Y.G., P.F., Y.C., T.C.W., C.J.F., A.S., L.C., N.G., R.H.G., and A.A.-G. as inventors.

REFERENCES

- Zhirnov, V.V., Cavin, R.K., Hutchby, J.A., and Bourianoff, G.I. (2003). Limits to binary logic switch scaling—a gedanken model. *Proc. IEEE* 9, 1934–1939.
- Cao, Y., Romero, J., Olson, J.P., Degroote, M., Johnson, P.D., Kieferová, M., Kivlichan, I.D., Menke, T., Peropadre, B., Sawaya, N.P.D., et al. (2019). Quantum chemistry in the age of quantum computing. *Chem. Rev.* 119, 10856–10915.
- Ladd, T., Jelezko, F., Laflamme, R., Nakamura, Y., Monroe, C., and O'Brien, J.L. (2010). Quantum computers. *Nature* 464, 45–53.
- Conrad, M. (1985). On design principles for a molecular computer. *Commun. ACM* 28, 464–480.
- Waldrop, M.M., and Mitchell Waldrop, M. (2016). The chips are down for Moore's law. *Nature* 530, 144–147.
- Rozenberg, G., Bäck, T., and Kok, J.N. (2012). *Handbook of Natural Computing* (Springer).
- Parrilla-Gutierrez, J.M., Sharma, A., Tsuda, S., Cooper, G.J.T., Aragon-Camarasa, G., Donkers, K., and Cronin, L. (2020). A programmable chemical computer with memory and pattern recognition. *Nat. Commun.* 11, 1442.
- Holley, J., Jahan, I., De Lacy Costello, B., Bull, L., and Adamatzky, A. (2011). Logical and arithmetic circuits in Belousov-Zhabotinsky encapsulated disks. *Phys. Rev. E* 84, 056110.
- Zhang, B., Fan, F., Xue, W., Liu, G., Fu, Y., Zhuang, X., Xu, X., Gu, J., Li, R., and Chen, Y. (2019). Redox gated polymer memristive processing memory unit. *Nat. Commun.* 10, 736.
- Galstyan, V., Funk, L., Einav, T., and Phillips, R. (2019). Combinatorial control through allostery. *J. Phys. Chem. B* 123, 2792–2800.
- Buchler, N.E., Gerland, U., and Hwa, T. (2003). On schemes of combinatorial transcription logic. *Proc. Natl. Acad. Sci. U S A* 100, 5136–5141.
- Katz, E. (2012). *Molecular and Supramolecular Information Processing* (Wiley-VCH Verlag GmbH & Co. KGaA).
- Katz, E. (2012). *Biomolecular Information Processing* (Wiley-VCH Verlag GmbH & Co. KGaA).
- Chiu, D.T., Pezzoli, E., Wu, H., Stroock, A.D., and Whitesides, G.M. (2001). Using three-dimensional microfluidic networks for solving computationally hard problems. *Proc. Natl. Acad. Sci. U S A* 98, 2961–2966.
- Nicolau, D.V., Lard, M., Korten, T., van Delft, F.C., Persson, M., Bengtsson, E., Månsson, A., Diez, S., Linke, H., and Nicolau, D.V. (2016). Parallel computation with molecular-motor-propelled agents in nanofabricated networks. *Proc. Natl. Acad. Sci. U S A* 113, 2591–2596.
- Einarsson, J. (2016). New biological device not faster than regular computer. *Proc. Natl. Acad. Sci. U S A* 113, E3187.
- Nicolau, D.V., Lard, M., Korten, T., van Delft, F.C., Persson, M., Bengtsson, E., Månsson, A., Diez, S., Linke, H., and Nicolau, D.V. (2016). Reply to Einarsson: the computational power of parallel network exploration with many bioagents. *Proc. Natl. Acad. Sci. U S A* 113, E3188.
- Zhu, L., Kim, S.-J., Hara, M., and Aono, M. (2018). Remarkable problem-solving ability of unicellular amoeboid organism and its mechanism. *R. Soc. Open Sci.* 5, 180396.
- Woods, D., Doty, D., Myhrvold, C., Hui, J., Zhou, F., Yin, P., and Winfree, E. (2019). Diverse and robust molecular algorithms using reprogrammable DNA self-assembly. *Nature* 567, 366–372.
- Adleman, L. (1994). Molecular computation of solutions to combinatorial problems. *Science* 266, 1021–1024.
- Rutten, M.G.T.A., Martin, G.T., Vaandrager, F.W., Johannes, A.A., and Nolte, R.J.M. (2018). Encoding information into polymers. *Nat. Rev. Chem.* 2, 365–381.
- Kirkpatrick, S., Gelatt, C.D., Jr., and Vecchi, M.P. (1983). Optimization by simulated annealing. *Science* 220, 671–680.
- Wang, T., and Roychowdhury, J. (2017). Oscillator-based Ising machine. *arXiv*, 1709.08102.
- Wang, T., and Roychowdhury, J. (2019). OIM: oscillator-based Ising machines for solving combinatorial optimisation problems. In *Unconventional Computation and Natural Computation*. UCNC 2019. Lecture Notes in Computer Science, 11493, I. McQuillan and S. Seki, eds., pp. 232–256.
- Chou, J., Bramhavar, S., Ghosh, S., and Herzog, W. (2019). Analog coupled oscillator based weighted Ising machine. *Sci. Rep.* 9, 1–10.
- Aramon, M., Rosenberg, G., Valiante, E., Miyazawa, T., Tamura, H., and Katzgraber, H.G. (2019). Physics-inspired optimization for quadratic unconstrained problems using a digital annealer. *Front. Phys.* 7, 48.
- Goto, H. (2016). Bifurcation-based adiabatic quantum computation with a nonlinear oscillator network. *Sci. Rep.* 6, 1–8.
- Goto, H., Tatsumura, K., and Dixon, A.R. (2019). Combinatorial optimization by simulating adiabatic bifurcations in nonlinear Hamiltonian systems. *Sci. Adv.* 5, eaav2372.
- Farhi, E., Goldstone, J., Gutmann, S., Lapan, J., Lundgren, A., and Preda, D. (2001). A quantum adiabatic evolution algorithm applied to random instances of an NP-complete problem. *Science* 292, 472–475.
- Santoro, G.E., Martonák, R., Tosatti, E., and Car, R. (2002). Theory of quantum annealing of an Ising spin glass. *Science* 295, 2427–2430.
- Johnson, M.W., Amin, M.H.S., Gildert, S., Lanting, T., Hamze, F., Dickson, N., Harris, R., Berkley, A.J., Johansson, J., Bunyk, P., et al. (2011). Quantum annealing with manufactured spins. *Nature* 473, 194–198.
- Mandrà, S., and Katzgraber, H.G. (2018). A deceptive step towards quantum speedup detection. *Quant. Sci. Technol.* 3, 04LT01.
- Boothby, K., Bunyk, P., Raymond, J., and Roy, A. (2019). Next-generation topology of D-wave quantum processors. *arXiv*, 2003.00133.
- McMahon, P.L., Marandi, A., Haribara, Y., Hamerly, R., Langrock, C., Tamate, S., Inagaki, T., Takesue, H., Utsunomiya, S., Aihara, K., et al. (2016). A fully programmable 100-spin coherent Ising machine with all-to-all connections. *Science* 354, 614–617.
- Inagaki, T., Haribara, Y., Igarashi, K., Sonobe, T., Tamate, S., Honjo, T., Marandi, A., McMahon, P.L., Umeki, T., Enbutsu, K., et al. (2016). A coherent Ising machine for 2000-node optimization problems. *Science* 354, 603–606.
- Hamerly, R., Inagaki, T., McMahon, P.L., Venturelli, D., Marandi, A., Onodera, T., Ng, E., Langrock, C., Inaba, K., Honjo, T., et al. (2019). Experimental investigation of performance differences between coherent Ising machines and a quantum annealer. *Sci. Adv.* 5, eaau0823.

37. Takahashi, K. (2020). Bifurcation-based quantum annealing with nested spins. *arXiv*, 2003.13439.
38. Cavell, A.C., Krasecki, V.K., Li, G., Sharma, A., Sun, H., Thompson, M.P., Forman, C.J., Guo, S.Y., Hickman, R.J., Parrish, K.A., et al. (2020). Optical monitoring of polymerizations in droplets with high temporal dynamic range. *Chem. Sci.* **11**, 2647–2656.
39. Perdomo-Ortiz, A., Feldman, A., Ozaeta, A., Isakov, S.V., Zhu, Z., O’Gorman, B., Katzgraber, H.G., Diedrich, A., Neven, H., de Kleer, J., et al. (2019). Readiness of quantum optimization machines for industrial applications. *Phys. Rev. Appl.* **12**, 014004.
40. Feng, W., Ueda, E., and Levkin, P.A. (2018). Droplet microarrays: from surface patterning to high-throughput applications. *Adv. Mater.* **30**, 1706111.
41. Bapst, V., Foini, L., Krzakala, F., Semerjian, G., and Zamponi, F. (2013). The quantum adiabatic algorithm applied to random optimization problems: the quantum spin glass perspective. *Phys. Rep.* **523**, 127–205.
42. Barahona, F. (1982). On the computational complexity of Ising spin glass models. *J. Phys. A Math. Gen.* **15**, 3241–3253.
43. Hayes, B. (1997). Can’t get no satisfaction. *Am. Sci.* **85**, 108–112.
44. Karp, R.M. (1972). Reducibility among combinatorial problems. In *Complexity of Computer Computations*. The IBM Research Symposia Series, R.E. Miller, J.W. Thatcher, and J.D. Bohlinger, eds., pp. 85–103.
45. Lucas, A. (2014). Ising formulations of many NP problems. *Front. Phys.* **2**, 5.
46. Istrail, S. (2000). Statistical mechanics, three-dimensionality and NP-completeness: I. Universality of intracatability for the partition function of the Ising model across non-planar surfaces (extended abstract). in *Proceedings of the thirty-second annual ACM symposium on Theory of computing—STOC ’00* 87–96 (ACM Press).
47. Cook, S. A. (1971). The complexity of theorem-proving procedures. in *Proceedings of the third annual ACM symposium on Theory of computing—STOC ’71*.
48. Babbush, R., O’Gorman, B., and Aspuru-Guzik, A. (2013). Resource efficient gadgets for compiling adiabatic quantum optimization problems. *Ann. Phys.* **525**, 877–888.
49. Davis, M., Logemann, G., and Loveland, D. (1962). A machine program for theorem-proving. *Commun. ACM* **5**, 394–397.
50. Perdomo-Ortiz, A., Dickson, N., Drew-Brook, M., Rose, G., and Aspuru-Guzik, A. (2012). Finding low-energy conformations of lattice protein models by quantum annealing. *Sci. Rep.* **2**, 571.
51. Babbush, R., Perdomo-Ortiz, A., O’Gorman, B., Mcready, W., and Aspuru-Guzik, A. (2014). Construction of energy functions for lattice heteropolymer models: efficient encodings for constraint satisfaction programming and quantum annealing. *Adv. Chem. Phys.* **155**, 201–244.
52. Booth, M.J., Restrepo Schild, V., Downs, F.G., and Bayley, H. (2017). Functional aqueous droplet networks. *Mol. Biosyst.* **13**, 1658–1691.
53. Sawaya, N.P.D., Rappoport, D., Tabor, D.P., and Aspuru-Guzik, A. (2018). Excitonics: a set of gates for molecular exciton processing and signaling. *ACS Nano* **12**, 6410–6420.
54. Choi, V. (2008). Minor-embedding in adiabatic quantum computation: I. The parameter setting problem. *Quant. Inform. Process.* **7**, 193–209.
55. Choi, V. (2011). Minor-embedding in adiabatic quantum computation: II. Minor-universal graph design. *Quant. Inform. Process.* **10**, 343–353.
56. Han, J., and Burgess, K. (2010). Fluorescent indicators for intracellular pH. *Chem. Rev.* **110**, 2709–2728.
57. Bosch, P., Catalina, F., Corrales, T., and Peinado, C. (2005). Fluorescent probes for sensing processes in polymers. *Chemistry* **11**, 4314–4325.
58. Elani, Y., Law, R.V., and Ces, O. (2014). Vesicle-based artificial cells as chemical microreactors with spatially segregated reaction pathways. *Nat. Commun.* **5**, 5305.
59. Huebner, A., Sharma, S., Srisa-Art, M., Hollfelder, F., Edel, J.B., and deMello, A.J. (2008). Microdroplets: a sea of applications? *Lab. Chip* **8**, 1244–1254.
60. Albash, T., Martin-Mayor, V., and Hen, I. (2019). Analog errors in Ising machines. *Quant. Sci. Technol.* **4**, 02LT03.
61. Vinci, W., Albash, T., and Lidar, D.A. (2016). Nested quantum annealing correction. *NPJ Quantum Inf.* **2**, 16017.
62. Pearson, A., Mishra, A., Hen, I., and Lidar, D.A. (2019). Analog errors in quantum annealing: doom and hope. *NPJ Quantum Inf.* **5**, 107.
63. Poon, C., and Kuan, Z. (2011). Neuromorphic silicon neurons and large-scale neural networks: challenges and opportunities. *Front. Neurosci.* **5**, 108.
64. Merolla, P.A., Arthur, J.V., Alvarez-Icaza, R., Cassidy, A.S., Sawada, J., Akopyan, F., Jackson, B.L., Imam, N., Guo, C., Nakamura, Y., et al. (2014). Artificial brains. A million spiking-neuron integrated circuit with a scalable communication network and interface. *Science* **345**, 668–673.