



OPEN

Neural networks using two-component Bose-Einstein condensates

Tim Byrnes¹, Shinsuke Koyama², Kai Yan¹ & Yoshihisa Yamamoto^{1,3}

¹National Institute of Informatics, 2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 101-8430, Japan, ²The Institute of Statistical Mathematics, 10-3 Midori-cho, Tachikawa, Tokyo 190-8562, Japan, ³E. L. Ginzton Laboratory, Stanford University, Stanford, CA 94305.

The authors previously considered a method of solving optimization problems by using a system of interconnected network of two component Bose-Einstein condensates (Byrnes, Yan, Yamamoto New J. Phys. 13, 113025 (2011)). The use of bosonic particles gives a reduced time proportional to the number of bosons N for solving Ising model Hamiltonians by taking advantage of enhanced bosonic cooling rates. Here we consider the same system in terms of neural networks. We find that up to the accelerated cooling of the bosons the previously proposed system is equivalent to a stochastic continuous Hopfield network. This makes it clear that the BEC network is a physical realization of a simulated annealing algorithm, with an additional speedup due to bosonic enhancement. We discuss the BEC network in terms of neural network tasks such as learning and pattern recognition and find that the latter process may be accelerated by a factor of N .

With the exception of a small class of problems that are solvable analytically, most quantum many-body problems can only be examined using numerical means, for which exact simulations scale exponentially with the problem size. Approximate methods such as quantum Monte Carlo and Density Matrix Renormalization Group (DMRG) give accurate results for certain cases but no general algorithm exists that can be applied to an arbitrary system. In the field of quantum information technology, quantum simulation has gathered a large amount of attention as an alternative means to study such quantum many-body problems. A quantum simulator is a device where a quantum many-body problem of interest is artificially created in the lab, such that its properties can be controlled and measured directly¹. By directly using quantum mechanics in the simulation, there is no exponential overhead in keeping track of the number of states in the Hilbert space of the problem. This is in the spirit of Feynman's original motivations for quantum computing², where quantum mechanics, rather than classical mechanics, is used to simulate quantum many-body problems.

Given this general approach to quantum many-body problems, the question of whether a quantum simulation approach can be applied to a general Ising model becomes an important question. The Ising model problem consists of finding the lowest energy state of the Hamiltonian

$$H_P = \sum_{i,j=1}^M J_{ij} \sigma_i \sigma_j + \sum_{i=1}^M \lambda_i \sigma_i, \quad (1)$$

where J_{ij} is a real symmetric matrix that specifies the connections between the sites i, j , and $\sigma_i = \pm 1$ is a spin variable. The task is then to find the minimal energy spin configuration $\{\sigma_i\}$ of the Hamiltonian (1). The problem of finding the solution of the Hamiltonian (1) is in fact known to be NP-complete, since it can be trivially mapped onto the MAX-CUT problem³. Furthermore, it can in principle encode an arbitrary NP-complete problem by a polynomial time mapping procedure⁴, thus the potential application of a reliable method of solving (1) is extremely broad. Although (1) is itself a classical Hamiltonian since it does not contain any non-commuting operators, as with quantum annealing where the Hamiltonian is supplemented with an additional transverse field⁵, quantum “tricks” may be used to speed up the solution of the ground state beyond classical methods.

In a previous work we investigated a computational device which finds the solution of an Ising model by a set of interconnected Bose-Einstein condensates^{6,7}. In the approach of Ref. 6, each spin was replaced by a system of N bosons which can take one of two states. By implementing an analogous Hamiltonian to (1) and cooling the system down into the ground state, it was shown that the solution of the original Ising model problem could be found. There is a speedup compared to simply implementing (1) using single spins, because of the presence of bosonic final state stimulation within each bosonic spin. This resulted in finding the solution of (1) with a speedup of approximately N . The attractive feature of the proposal in Ref. 6 is that the computation is done simply by

SUBJECT AREAS:

ULTRACOLD GASES

INFORMATION THEORY AND
COMPUTATION

APPLIED MATHEMATICS

QUANTUM INFORMATION

Received
1 March 2013

Accepted
12 July 2013

Published
30 August 2013

Correspondence and
requests for materials
should be addressed to
T.B. (tbyrnes@nii.ac.
jp)



implementing a static Hamiltonian, so that no complicated gate sequence needs to be employed in order to find the ground state solution. Effectively, the dissipative process itself performs the computation itself, and therefore can also be considered to be a reservoir engineering approach to quantum computation. Related proposals were offered in Refs. 8–10, where instead of BECs, photons were used.

In this paper, we analyze the proposal in Ref. 6 from the point of view of neural networks, specifically the stochastic continuous Hopfield model. Recasting the proposal in this form allows for a clearer analysis of the properties of the device, where standard results can be carried over to the BEC case. It clarifies the origin of the $\sim N$ speedup of the device, which was established via a numerical approach in Ref. 6. We find that the $\sim N$ speedup originates from each element of the Hopfield network being accelerated due to bosonic stimulation, and thermal fluctuations provide the stochastic aspect to the Hopfield network. We then consider some simple applications of the BEC network for neural networking tasks.

Results

Bose-Einstein condensate networks. We first give a brief description of the proposal of Ref. 6. In order to solve (1) we consider a system such as that shown in Figure 1. Each spin σ_i in H_P is associated with a trapping site containing N bosonic particles. The bosons can occupy one of two spin states, which we label by $\sigma = \pm 1$. The set of possible states can then be written

$$|k\rangle = \frac{1}{\sqrt{k!(N-k)!}} \left(a_{i+}^\dagger\right)^k \left(a_{i-}^\dagger\right)^{N-k} |0\rangle \quad (2)$$

where $a_{i\sigma}$ is the annihilation operator for spin σ on site i . A concrete example of the two levels for the case of atomic BECs can be found in Fig. 6(a) of Ref. 11. On each site i we may define a total pseudospin (which we call spin henceforth for brevity) operator $S_i = a_{i+}^\dagger a_{i+} - a_{i-}^\dagger a_{i-}$ taking eigenvalues $S_i |k\rangle = (-N + 2k) |k\rangle$. The sites are controlled such that the system follows the bosonic Ising Hamiltonian

$$H = \sum_{i,j=1}^M J_{ij} S_i S_j + \sum_{i=1}^M \lambda_i S_i \quad (3)$$

where J_{ij} is the same matrix as in H_P which specifies the computational problem, and $s_i = S_i/N$ are normalized spin variables.

The ground state of (3) can be used to infer the ground state of (1). This follows from the fact that for extremal states $s_i = \pm 1$, the same

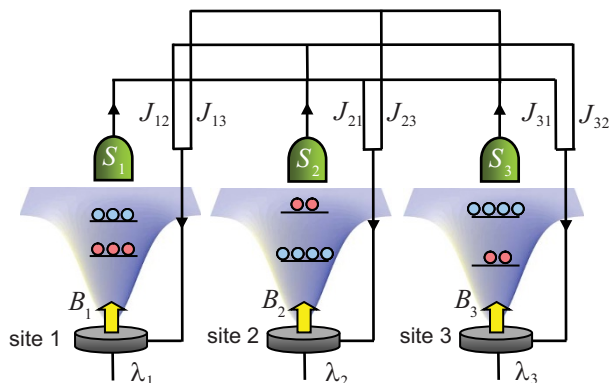


Figure 1 | Each site of the Ising Hamiltonian is encoded as a trapping site, containing N bosons. The bosons can occupy one of two states $\sigma = \pm 1$, depicted as either red or blue. The interaction between the sites may be externally induced by measuring the total spin S_i on each site i via the detectors. A local field on each site equal to $B_i = \sum_j J_{ij} S_j + \lambda_i$ is applied via the feedback circuit.

structure of the Hamiltonian gives the same ordering of energy states. However, unlike the discrete σ_i variables, s_i take a large number of values between -1 and 1 , and thus there are many more states present in (3) in comparison to (1). One may then worry that some of these states are in fact lower than the ground state. As discussed in Ref. 6, it can be shown that such states can never be lower in energy than the extremal states, and are at worst degenerate with the ground state configuration of (1). Thus one may always deduce a ground state of (1) by making the assignment $\sigma_i = s_i/|s_i|$, provided s_i is a ground state configuration.

One possible method of creating the interactions experimentally in (3) is via a measurement-feedback approach. In this approach, the total spin S_i on each site is continuously measured, and an appropriate control signal is continuously fed back into the system by applying a local dc field on another site. Given a measurement result of $\{s_j(t)\}$ across the spins, a local field

$$B_i(t) = \sum_j J_{ij} s_j(t) + \lambda_i \quad (4)$$

is applied on site i . Since the Zeeman energy due to this field is

$$H = \sum_i B_i S_i, \quad (5)$$

a simple substitution yields (3).

Starting from a random spin configuration $\{s_j(t)\}$, the system is cooled assuming that the ambient temperature T is fixed. The procedure is essentially identical to a simulated annealing procedure, the sole difference being the use of the bosonic Ising model. By varying the temperature during the cooling process such that it is time dependent $T(t)$, a strategy similar to thermal annealing may be performed, in order to escape being trapped in local minima. In practice, instead of varying the temperature, varying the overall magnitude of (3) by adjusting the strength of the magnetic field (4) is equivalent. By taking advantage of the bosonic amplification of the cooling process, it was found in Ref. 6 that an approximate factor of N was found in the cooling process.

The time evolution is modeled by an extension of the method presented by Glauber to bosons¹⁸. Given the M site Hamiltonian (3), the states are labeled

$$|k\rangle = \prod_{i=1}^M \frac{1}{\sqrt{k_i!(N-k_i)!}} \left(a_{i+}^\dagger\right)^{k_i} \left(a_{i-}^\dagger\right)^{N-k_i} |0\rangle, \quad (6)$$

where the k_i range from 0 to N , $a_{i\sigma}^\dagger$ is the creation operator for a boson on site i in the state σ , and we have defined the vector $\mathbf{k} = (k_1, k_2, \dots, k_M)$. The approach as described in Ref. 6 is to assign a probability p_k of occupation of each configuration labeled by $|k\rangle$. The system then evolves between the states $|k\rangle \rightarrow |k'\rangle$ by a probabilistic process, characterized by transition weight factors w . The weight factors are chosen such that in the long time limit, the states k obey a thermal equilibrium statistics.

Given an initial probability distribution set by the initial conditions, p_k then evolves according to

$$\frac{dp_k}{dt} = \sum_{i=1}^M \left[-w(\mathbf{k}, \delta_i) p_k + w(\mathbf{k} + \delta_i, -\delta_i) p_{\mathbf{k}+\delta_i} - w(\mathbf{k}, -\delta_i) p_k + w(\mathbf{k} - \delta_i, \delta_i) p_{\mathbf{k}-\delta_i} \right] \quad (7)$$

where δ_i is a vector of length M with its i th element equal to one and zero otherwise. The $w(\mathbf{k}, \delta_i)$ is a weight factor representing the transition $|k\rangle \rightarrow |k + \delta_i\rangle$, containing both the bosonic final state stimulation factor and a coefficient to ensure that the system evolves to the correct thermal equilibrium distribution. We have restricted the transitions to first order transitions in (7) for simplicity. The final state stimulation factor can be calculated by assuming a Hamiltonian



$H_i = \sum_{\sigma} a_{i\sigma}^\dagger a_{i-\sigma}$ and calculating the transition rate according to Fermi's golden rule (up to a constant)

$$|\langle \mathbf{k} + \boldsymbol{\delta}_i | H_i | \mathbf{k} \rangle|^2 = (k_i + 1)(N - k_i), \quad (8)$$

At thermal equilibrium, the transition rates are equal between the states $|\mathbf{k}\rangle \leftrightarrow |\mathbf{k} + \boldsymbol{\delta}_i\rangle$, which ensures that $\frac{dp_k}{dt} = 0$. The final state stimulation factors cancel for such pairs and we have

$$\frac{w(\mathbf{k}, \boldsymbol{\delta}_i)}{w(\mathbf{k} + \boldsymbol{\delta}_i, -\boldsymbol{\delta}_i)} = \frac{p_{\mathbf{k} + \boldsymbol{\delta}_i}}{p_{\mathbf{k}}}, \quad (9)$$

and similarly for $\boldsymbol{\delta}_i \rightarrow -\boldsymbol{\delta}_i$. From the probability distribution at thermal equilibrium we can calculate

$$\frac{p_{\mathbf{k} + \boldsymbol{\delta}_i}}{p_{\mathbf{k}}} = \exp\left(-2\beta \left[\sum_{j \neq i} J_{ij}(N - 2k_j) + \lambda_i \right]\right), \quad (10)$$

where $\beta = \frac{1}{k_B T}$. This gives the coefficients as

$$\begin{aligned} w(\mathbf{k}, \boldsymbol{\delta}_i) &= \alpha(1 + \gamma_i)(k_i + 1)(N - k_i) \\ w(\mathbf{k}, -\boldsymbol{\delta}_i) &= \alpha(1 - \gamma_i)k_i(N - k_i + 1) \end{aligned} \quad (11)$$

where

$$\gamma_i = \tanh\left(-\beta \left[\sum_{j \neq i} J_{ij}(N - 2k_j) + \lambda_i \right]\right), \quad (12)$$

α is a constant determining the overall transition time scale. α is set to 1 for all numerical calculations.

Equivalence to the stochastic continuous Hopfield model. We now show that the scheme detailed in the previous section is formally equivalent to the stochastic continuous Hopfield model. After defining the Hopfield model we show the equivalences between the two systems, and derive the evolution equations for the BEC network in the context of the Hopfield model.

Definition of the Hopfield model. The Hopfield model is an asynchronous model of an artificial neural network^{12,13}, where the each unit of the network changes state at random times, such that no two units change simultaneously. The Hopfield model is in the class of recurrent neural networks, where the output of the network is fed back into itself, so that given a particular initial configuration, the system undergoes a sequence of changes in configuration of the units until steady state is achieved.

In the standard (discrete, deterministic) Hopfield model, each unit takes the value $\sigma_i = \pm 1$. The units are updated according to the rule

$$\sigma_i \rightarrow \sigma'_i = \text{sgn}\left[\sum_j \mathcal{J}_{ij}\sigma_j + b_i\right] \quad (13)$$

where $\text{sgn}(x) = x/|x|$ denotes the sign function, \mathcal{J}_{ij} is a symmetric matrix with zero diagonal elements, b_i is the threshold of unit i . The units are updated to their new values σ'_i at random. Whether a given state is stable with respect to the updates can be determined by the Lyapunov (energy) function

$$E = \sum_{ij} \mathcal{J}_{ij}\sigma_i\sigma_j + \sum_i b_i\sigma_i. \quad (14)$$

The update sequence proceeds until a local minima with respect to the Lyapunov function is reached. From a physics perspective, the transition rule (13) can be viewed as a cooling process at zero temperature, where given an initial high energy spin configuration, the spins cool one by one randomly into a low energy state. It is thus equivalent to the discrete Ising model problem (1) up to a constant

energy factor originating from the diagonal elements of J_{ij} , which play no part in the dynamics of the problem.

The model can be extended to one where continuous variables $x_i \in [-1, 1]$ are used in place of the discrete ones σ_i . Such continuous Hopfield networks have similar properties to the discrete version in terms of the configuration of stable states¹⁴. The way the model is usually defined is in terms of an electric circuit model (see for e.g. Ref. 13). On a single unit i , the time evolution of the circuit obeys

$$C_i \frac{dv_i}{dt} = -\frac{v_i}{R_i} + b_i + \sum_j \mathcal{J}_{ij}x_j \quad (15)$$

where C_i is the capacitance, R_i is the resistance, v_i is the voltage, and x_i is the output of the circuit after operation of the nonlinear amplifier (or activation function). The activation function restricts the output to a limited region such that output is always in the range $[-1, 1]$. A typical choice is

$$x_i = \phi(v_i) = \tanh(v_i). \quad (16)$$

The corresponding energy function for the dynamics is

$$E = \sum_{ij} \mathcal{J}_{ij}x_ix_j + \sum_i b_ix_i + \sum_i \frac{1}{R_i} \int \phi^{-1}(x)dx. \quad (17)$$

From the energy function it may then be shown that the system is guaranteed to converge to a local minima of the energy. The last term in (17) gives a modification of the energy landscape which shifts the position of the minima away from the extrema $x = \pm 1$. For a sufficiently high-gain amplifier (corresponding to a modification of the activation function to $\phi(v_i) \rightarrow \phi(Gv_i)$ with $G > 1$), this term is known to have a negligible effect on the overall dynamics¹⁴. Due to the energy structure of the continuous model being equivalent to the discrete model¹³, a solution to the continuous model then gives a one-to-one correspondence to the discrete model.

Equations of motion on a single site. In order to see the equivalence with the Hopfield model, let us first examine the dynamics on a single site, and set $M = 1$. In this case the probability distribution of the states evolve as

$$\begin{aligned} \frac{dp_k}{dt} &= -w(k, 1)p_k + w(k+1, -1)p_{k+1} \\ &\quad -w(k, -1)p_k + w(k-1, 1)p_{k-1}. \end{aligned} \quad (18)$$

where in this case

$$\begin{aligned} w(k, 1) &= \alpha(1 + \gamma)(k+1)(N - k) \\ w(k, -1) &= \alpha(1 - \gamma)k(N - k + 1) \\ \gamma &= \tanh(-\beta\lambda), \end{aligned}$$

since there is only one site. Multiplying the whole equation by k and summing over k given an equation for the mean value

$$\frac{d\langle k \rangle}{dt} = \alpha(1 + \gamma)\langle (k+1)(N - k) \rangle - \alpha(1 - \gamma)\langle k(N - k + 1) \rangle. \quad (19)$$

Making the approximation that $\langle k^2 \rangle \approx \langle k \rangle^2$, and changing variables to $S = -N + 2k$ the equation can be recast into the form

$$\frac{1}{\alpha} \frac{ds}{dt} = -N\gamma s^2 - 2s + \gamma(2 + N) \quad (20)$$

where we have used the normalized variable $s = \langle S \rangle / N$. This approximation is accurate under physically relevant probability distributions generated by (18). For example, a uniform distribution $p_k = 1/N$ gives $\langle k^2 \rangle \approx N^2/4$, in comparison to the exact result $\langle k^2 \rangle \approx N^2/3$. The approximation improves as the distribution becomes more peaked. An explicit solution for this may be found to be



$$s(t) = A \tanh(\alpha \gamma A N t + K_0) - \frac{1}{N \gamma} \quad (21)$$

where $A = \sqrt{1 + \frac{2}{N} + \frac{1}{N^2 \gamma^2}}$ is a constant that is order unity at low temperatures such that $N \gamma \gg 1$. K_0 is fixed by the initial conditions, for which we typically assume $s(t=0) = 0$. We see that the spin approaches its steady state value with a timescale $\sim 1/\alpha |\gamma| N$. At zero temperature where $\gamma = \pm 1$ (depending upon the direction of the applied external field), the time scale is enhanced by a factor of N .

The physical origin of the speedup is due to bosonic final state stimulation. When N indistinguishable particles occupy a quantum state, the rate of population transfer is enhanced by a factor of $\sim N$. This effect is most familiar from the theory of lasing, where photon emission occurs preferentially into the lasing mode¹⁵. In our case, the cooling rate of the bosons on a single site is accelerated by a factor of N , due to the bosonic factors (8) in the transition rates (11). This accelerated cooling is what provides the speedup of the network as a whole⁶.

Let us compare this to the Hopfield network for a single site. In this case the equation of motion reads

$$C \frac{dv}{dt} = -\frac{v}{R} + b. \quad (22)$$

Changing variables to the output of the nonlinear amplifier, we obtain

$$C \frac{dx}{dt} = \left(\frac{d\phi^{-1}}{dx} \right)^{-1} \left[-\frac{\phi^{-1}(x)}{R} + b \right]. \quad (23)$$

This is the counterpart of the equation (20) for the Hopfield case. The solution of this

$$x(t) = \phi(bR + \exp[-t/CR + K'_0]), \quad (24)$$

where K'_0 is set by the initial conditions.

Writing the equations in this form makes the correspondence between the BEC system and the Hopfield network clear, which we summarize in Table I. The output x of the Hopfield network corresponds to the normalized spin variable s . The magnetic field B applied on each site for the BEC system then corresponds to the voltage on each Hopfield unit before the non-linear amplifier. The steady state values are determined by setting the right hand side of (20) and (23) to zero. The value of the steady state depends upon the ratio of the field (λ or b) applied and the temperature $k_B T = 1/\beta$ or the conductance $1/R$ respectively. The overall time constant is controlled by $1/\alpha$ or C in each case. In the Hopfield network, there is obviously no bosonic final state stimulation, so the speedup proportional to N is absent. While the exact equation of motion for the two cases (20) and (23) differ in their details, it is clear that the qualitatively there is a similar structure and behavior to the dynamics. While in the Hopfield network, the overall time constant is determined by the capacitance of the circuit, the fundamental timescale of the BEC network is determined by the cooling rate of the bosons on each site.

Table I | Equivalences between the network of BECs proposed in Ref. 6 and the Hopfield model

Quantity	BEC network	Hopfield model
Site variable	$s_i = \langle S_i \rangle / N$	x_i
Site field variable	B_i	v_i
Ising matrix	J_{ij}	\mathcal{J}_{ij}
Local field	λ_i	b_i
Time constant	$1/\alpha$	C
Steady-state control parameter	$1/k_B T$	R_i
Activation function	Φ	ϕ

The analogue of the activation function $\phi(v)$ may be derived as follows. Considering the activation function as a rule that converts the internal magnetic field B to the spin variable s , we may derive the average spin at thermal equilibrium using the Boltzmann distribution taking into account of bosonic statistics. From the partition function

$$p_k^{\text{eq}} = (1 - \exp(-2\beta B)) \exp(-2\beta B k), \quad (25)$$

and therefore

$$\begin{aligned} s^{\text{eq}} &= \sum_k p_k^{\text{eq}} (-1 + 2k/N) \\ &= \Phi(z) \equiv \frac{(1 - e^{2z(2+N)}) + (2+N)(e^{2z(N+1)} - e^{2z})}{(1 - e^{2z})(1 - e^{2z(1+N)})} \end{aligned} \quad (26)$$

where $z = \beta B$ here. The function above has a dependence that has similar behavior to

$$\Phi(z) \approx \tanh\left(-\frac{z(N+2)}{3}\right) \quad (27)$$

which makes it clear that it plays a similar role to that of the activation function ϕ in the Hopfield network.

Equations of motion for interconnected BECs. We may now generalize to the multi-site case. Multiplying (7) by k_i and summing over all k gives the equations

$$\begin{aligned} \frac{d\langle k_i \rangle}{dt} &= \alpha \langle (1 + \gamma_i(\mathbf{k}))(k_i + 1)(N - k_i) \rangle \\ &\quad - \alpha \langle (1 - \gamma_i(\mathbf{k}))k_i(N - k_i + 1) \rangle. \end{aligned} \quad (28)$$

where we have written $\gamma_i \rightarrow \gamma_i(\mathbf{k})$ to remind ourselves that this is not a constant in this case. Making the approximation that $\langle k^m \rangle \approx \langle k \rangle^m$, and making the change of variables to $s_i = \langle S_i \rangle / N$, we obtain

$$\frac{1}{\alpha} \frac{ds_i}{dt} = -N \gamma_i(s) s_i^2 - 2s_i + \gamma_i(s)(2 + N). \quad (29)$$

The sole difference to the single site case here is that the equilibrium values of s_i are now dependent on the spins of all the other sites s . The dynamics on each site is the same as the single site case, and thus evolves in time as (21), considering the other spins to be approximately fixed. This basic structure is precisely the same dynamics that determine the equation of motion of the Hopfield network (15).

Although it is not possible to solve the set of equations (29) analytically, we may see in this formulation why the whole system should have a speedup of $\sim N$, as found in Ref. 6. Considering an asynchronous update procedure (in fact this is exactly what was performed to simulate the dynamics of the system in the Monte Carlo procedure⁷), all but one “active” site is fixed in spins. The active site then evolves in time according to the evolution of (21). This has a speedup of $\sim N|\gamma|$ in the evolution of the spin, thus to make an incremental change δs in the active spin takes a time reduced by $N|\gamma|$ compared to the $N = 1$ case. The spin is then fixed, and then another site is chosen at random and this is updated. Since each step takes a reduced time of $N|\gamma|$, the whole evolution proceeds at an accelerated rate. For sufficiently low temperatures, $\gamma \approx \pm 1$, and therefore the speedup is approximately $\sim N$.

Stochastic Hopfield network. Up to this point, the equations of motion (29) have been entirely deterministic. The role of the temperature was to merely shift the equilibrium values of the spins, as determined by (26), and did not contribute to any stochastic evolution of the system. In the BEC network there are thermal fluctuations which cause the system to behave in a random manner. Therefore in order to fully capture the dynamics of the BEC network we must include the contribution of the random thermal process. Such



stochastic versions of Hopfield networks, and their generalization to the Boltzmann machine (by having additional hidden units) are defined by modifying the update rule (13) to include probabilistic effects. Considering the discrete Hopfield model first, the algorithm consists of selecting a particular spin σ_i and making an update according to

$$\begin{aligned}\sigma_i &\rightarrow -\sigma_i & \text{if } \Delta E < 0 \\ \sigma_i &\rightarrow -\sigma_i & \text{if } \Delta E > 0 \text{ with probability } e^{-\Delta E/k_B T} \\ \sigma_i &\rightarrow \sigma_i & \text{if } \Delta E > 0 \text{ and otherwise.}\end{aligned}\quad (30)$$

ΔE is the energy difference between the state with the flipped spin and no flipped spin¹⁶.

The evolution equations (7) for the BEC network can be converted into a set of stochastic update rules which give the same time dependence when an ensemble average is taken. The stochastic formulation also allows for a convenient method of numerically simulating the system, which was discussed in detail in Ref. 7. We briefly describe the procedure as applied to the current formulation of the BEC network. The simulation is started from a random initial value of $\mathbf{k} = (k_1, k_2, \dots, k_M)$ in (7), and we update the system by repeating the stochastic transition process following the kinetic Monte Carlo method¹⁷. In each update we calculate the transition weight $w(\mathbf{k}, \delta_i)$ in (7) for all the possible transitions. The transition is then made with a probability in proportion to the transition weight $w(\mathbf{k}, \delta_i)$. The time increment is then calculated according to

$$\Delta t = -\ln(r)/W_{\text{tot}} \quad (31)$$

where $r \in (0, 1]$ is a randomly generated number and

$$W_{\text{tot}} = \sum_i (w(\mathbf{k}, \delta_i) + w(\mathbf{k}, -\delta_i)) \quad (32)$$

This procedure is repeated for many trajectories so that ensemble averages of quantities such as the average spin can be taken.

This procedure produces exactly the same time dynamics as (7), hence for quantities such as the equilibration time this procedure must be followed. However, if only the behavior at equilibrium is required, the update procedure can be replaced by the Metropolis algorithm. The update procedure is then as follows. Start from a random initial value of $\mathbf{k} = (k_1, k_2, \dots, k_M)$. Then make an update according to

$$\begin{aligned}k_i &\rightarrow k_i \pm 1 & \text{if } \Delta E < 0 \\ k_i &\rightarrow k_i \pm 1 & \text{if } \Delta E > 0 \text{ with probability } e^{-\Delta E/k_B T} \\ k_i &\rightarrow k_i & \text{if } \Delta E > 0 \text{ and otherwise.}\end{aligned}\quad (33)$$

where the energy difference in this case is

$$\Delta E = \pm 2 \left[\sum_{j \neq i} J_{ij} (N - 2k_j) + \lambda_i \right]. \quad (34)$$

The exponential factor is precisely the same as that determining the weight factors in (10). The only difference in this case is that the bosonic stimulation factors are not present in this case, which are important only for determining the transition rates, and not the equilibrium values. The above Metropolis transition rule (10) is identical to the stochastic Hopfield network, up to the difference that each site contains energy levels between $k_i = 0, \dots, N$. It is therefore evident in this context that the two systems are equivalent in their dynamics.

We may also derive the equivalent stochastic differential equations for the average spin by adding noise terms to (29) (see Methods)

$$\begin{aligned}\frac{1}{\alpha} \frac{ds_i}{dt} &= -N\gamma_i(s)s_i^2 - 2s_i + \gamma_i(s)(2+N) \\ &+ \sqrt{\frac{2}{\alpha} \left[(1+s_i)(1-s_i) + \frac{2}{N} (1-\gamma_i(s)s_i) \right]} \xi_i(t).\end{aligned}\quad (35)$$

The above equation is identical to (29) up to the Gaussian noise term. This allows for the system to escape local minima in the energy landscape. The steady state evolutions then approach the correct thermal equilibrium averages as defined by the Boltzmann distribution. Combined with an annealing schedule, this may be used to find the ground state of the Ising Hamiltonian. As found previously, due to the factors of N in (35) originating from bosonically enhanced cooling, the annealing rates may be made N times faster, allowing for an accelerated method of finding the solution to Ising model problems, as claimed in Ref. 6.

Learning and pattern completion. Our discussion to this point has focused on the problem of finding the ground state given a particular problem instance encoded by the J_{ij} parameters. This may be given for example by translating a MAX-CUT problem into the corresponding J_{ij} matrix. The task is then to try and find the optimal spin configuration for this problem instance. However, given the correspondence of the BEC network to the continuous Hopfield model, it is clear that there should be other applications beyond this problem solving scenario.

One common application of neural networks is learning and memory retrieval. In this application the network is initially exposed to several patterns that the user would like to retrieve later. In this step the J_{ij} matrix is modified according to a prescribed learning algorithm. Thus in contrast to the previous application J_{ij} is the quantity to be determined in the learning step. Once the learning process is complete, the network is put in retrieval mode and the J_{ij} parameters are fixed. The network now operates in the same way as the problem solving scenario considered previously. Given an initial spin configuration that is similar to the stored configurations, the network then retrieves the stored patterns, demonstrating pattern completion.

In this section, we discuss the application of learning and memory retrieval in BEC networks. The first step of learning can be straightforwardly adapted from standard methods¹³. To illustrate the technique, we consider the Hebbian learning, one of the simplest unsupervised learning algorithms. The performance of the pattern completion process is then assessed by examining the time taken to retrieve the patterns. This serves to both illustrate the equivalence of the BEC network to the continuous Hopfield model, and show the potential benefits of using many bosons in the system.

Learning. The simplest example of unsupervised learning is the Hebbian learning rule. Using the associations in Table I it is straightforward to write down the corresponding rule in the case of the BEC network. We follow the presentation given in Ref. 13 (sec. 2.5) for the case of continuous activation functions, since in the BEC system the measured spin is a continuous quantity. We assume that the BEC network starts with the system shown in Figure 1 with the Ising matrix set to

$$J_{ij} = 0. \quad (36)$$

The learning algorithm then proceeds as follows. We apply various magnetic field configurations $\lambda_i^{(n)}$ where n labels the various pattern configurations that the network is exposed to during the learning process. Starting with the first field configuration $n = 1$, we apply this field and wait until the spins reach their equilibrium value, which will be given (for the first iteration)

$$s_i = \Phi(\beta \lambda_i^{(1)}) \quad (37)$$

We then update the Ising matrix according to

$$J_{ij} \rightarrow J_{ij} + cs_i \lambda_j^{(n)} \quad (38)$$

where c is the learning constant which determines the speed of the learning process. We then make subsequent applications of the field



$\lambda_i^{(n)}$, measure the field s_i in each case, then make the replacement (38). For $n \geq 2$, the spins do not simply take the values of (37) since J_{ij} will in general be non-zero. The process is continued until the learning examples are exhausted, or the same set can be recycled. Other learning algorithms may be derived in a similar way using the associations in Table I.

We now discuss in more detail the experimental configuration to realize the Hebbian learning rule for the BEC network. In the configuration of Figure 1, we assume that the J_{ij} matrices can be changed as desired, i.e. they are computer controlled. We also assume that the spins s_i can be read out in addition to the existing feedback circuit. The experimental procedure then is as follows. 1) Set $J_{ij} = 0$; 2) Apply the field $B_i = \lambda_i^{(1)}$ to the sites and wait for equilibration; 3) Read the spins s_i and apply the update rule (38); 4) Apply the field $B_i = \sum_j J_{ij} s_j(t) + \lambda_i^{(n)}$ and wait for equilibration; 5) Go to step 3 until the patterns are exhausted. This builds up the J_{ij} matrix which can be used in the next step for pattern recognition.

Pattern completion. As a simple example of the use of the BEC network for pattern completion, we test the set of equations given in (29) using an Ising matrix trained using the Hebbian algorithm of the previous section. We numerically evolve a set of 16×16 equations forming a two dimensional grid at zero temperature from the initial conditions as shown in Figure 2. In Figures 2a and 2b we start from fragments of the learned patterns while in Figure 2c we start from a randomly chosen spin configuration. We see that in all cases the spins evolve towards the learned configurations, with the BEC network completing the patterns as desired. For the random initial configuration, the spins evolve towards whichever configuration happens to be closer to the learned patterns.

The time scaling behaviour is shown in Figure 3. In Figure 3a we plot the normalized Hamming distance

$$D = \frac{1}{2M} \sum_i |s_i(t) - s_i^{(n)}| \quad (39)$$

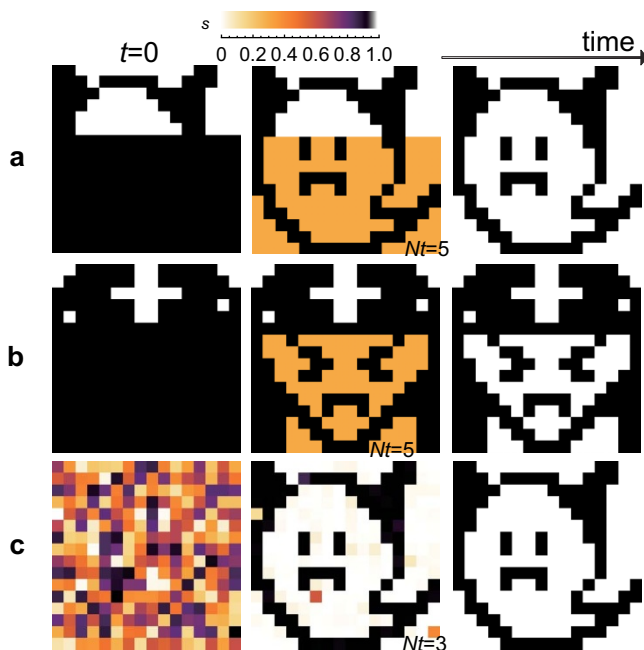


Figure 2 | Pattern completion for a BEC network prepared with Hebbian learning for the two characters in (a) and (b). The BEC network is evolved in time starting from the initial conditions as shown and evolve towards their steady state configurations. Parameters used are $\alpha = 1$, $N = 10^4$, $k_B T = 0$. Pixel characters provided courtesy of Akari Moffat (blablahospital).

between the evolved spin configuration and the learned spin configurations $s_i^{(n)}$. We see that the general behavior is that the time for the pattern completion scales as $\sim 1/N$, which can again be attributed to bosonic stimulated cooling. There is a logarithmic correction to this behavior, where there is an initial stiffness of the spins to move towards the steady state configuration. In Figure 3b we show the scaling of the time to reach a particular Hamming distance ϵ with respect to N . We see that for large N all curves converge to the dominant $\sim 1/N$ behaviour. This shows that BEC networks can equally well be used to perform tasks such as pattern completion, with the additional benefit of a reduced time in proportion to N .

Discussion

We have analyzed the BEC network proposed in Ref. 6 in terms of the theory of neural networks and found that it is equivalent to a stochastic continuous Hopfield model. In contrast to the continuous Hopfield model where the overall timescale of the evolution is determined by the capacitance within each unit, in the BEC network the timescale is determined via the rate of cooling. Due to bosonic stimulated cooling, the rate of cooling may be accelerated in proportion to the number of bosons N on each site, which in turn accelerates the cooling rate of the entire system. The bosonic stimulated cooling makes the time evolution equations (29) on a single site not precisely the same as its Hopfield model counterpart (15), but the difference merely gives a modification of the dynamics as the system heads towards equilibrium, the overall behaviour of the system as a whole remains the same. In particular, tasks such as pattern completion may be performed using the BEC network, in the same way as the Hopfield model.

In this context, it would appear that using a BEC network, rather than a physical implementation of a Hopfield network, is nothing but a more complicated way of implementing what could be done equally well by either standard electronics or optical means¹². Specifically, one could imagine using simply Hopfield circuits with small capacitances such that the timescale of the circuit is as small as desired. Other variations of optical implementations of Hopfield models allow for fast operation speeds. While for the zero temperature case this may be true, the BEC system does have the advantage that the random fluctuations following Boltzmann statistics is already built-in, and do not require additional circuitry to simulate. Another possible advantage is that the speedups can be made systematically faster by simply increasing the number of bosons. At the level of the model that we present in this paper there is no fundamental limitation to the speedups that can be achieved. However, in practice other issues are likely to be present, so any speedup must be taken advantage of within the practical limitations that the experimental device imposes.

A possible issue for the physical realization is whether the connections between each Ising site require response times of the order of the cooling time on each site. Apart from a simple slowdown due to bottlenecks in the transmission, such delay times in the information between each site can introduce instabilities in the system causing

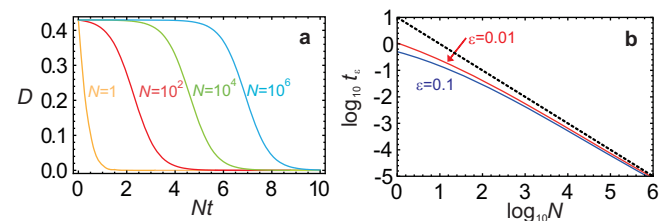


Figure 3 | (a) Hamming distance D versus time for the pattern recognition task given in Figure 2a for various boson numbers N as shown. (b) Times t_e necessary for reaching a Hamming distance of ϵ as a function of N . Dotted line corresponds to $1/N$.



divergent behavior. We leave as future work whether the proposals in Refs. 8–10 can be treated with the same analysis.

Methods

First consider a single site, and start with the probability distribution (18). Assuming that $N \gg 1$, introduce the variables $z = k/N$, $\epsilon = 1/N$, and the density $q(z, t) = pk/\epsilon$ of z at time t , the master equation is rewritten

$$\frac{\partial q(z, t)}{\partial t} = -w(z, \epsilon)q(z, t) + w(z + \epsilon, -\epsilon)q(z + \epsilon, t) - w(z, -\epsilon)q(z, t) + w(z, \epsilon)q(z, t). \quad (40)$$

Expanding $w(z \pm \epsilon, \mp \epsilon)$ and $q(z \pm \epsilon, t)$ up to second order in ϵ , we obtain

$$\begin{aligned} \frac{\partial q(z, t)}{\partial t} &= -\frac{\partial}{\partial z}[(w(z, \epsilon) - w(z, -\epsilon))\epsilon q(z, t)] \\ &\quad + \frac{1}{2} \frac{\partial^2}{\partial z^2}[(w(z, \epsilon) + w(z, -\epsilon))\epsilon^2 q(z, t)] + O(\epsilon^3) \\ &= -\frac{\partial}{\partial z}[A_\epsilon(z)q(z, t)] + \frac{1}{2} \frac{\partial^2}{\partial z^2}[B_\epsilon(z)q(z, t)] + O(\epsilon^3), \end{aligned} \quad (41)$$

where

$$\begin{aligned} A_\epsilon(z) &= (w(z, \epsilon) - w(z, -\epsilon))\epsilon \\ B_\epsilon(z) &= (w(z, \epsilon) + w(z, -\epsilon))\epsilon^2. \end{aligned} \quad (42)$$

Using the diffusion approximation such that the transition rates are on the order of $w(z, \pm \epsilon) \sim 1/dt$, $(w(z, \epsilon) - w(z, -\epsilon)) \sim \epsilon/dt$, and $\epsilon^2/dt \sim O(1)$, and taking the limits of $\epsilon \rightarrow 0$, we obtain the Fokker-Planck equation

$$\frac{\partial q(z, t)}{\partial t} = -\frac{\partial}{\partial z}[A(z)q(z, t)] + \frac{1}{2} \frac{\partial^2}{\partial z^2}[B(z)q(z, t)], \quad (43)$$

where $A(z) = \lim_{\epsilon \rightarrow 0} A_\epsilon(z)$ and $B(z) = \lim_{\epsilon \rightarrow 0} B_\epsilon(z)$. The corresponding stochastic differential equation is given by

$$\frac{dz}{dt} = A(z) + \sqrt{B(z)}\xi(t) \quad (44)$$

where $\xi(t)$ is Gaussian white noise with $\langle \xi(t) \rangle = 0$ and $\langle \xi(t)\xi(t') \rangle = \delta(t - t')$. Changing variables to $s = -1 + 2z$, the coefficients for our case are

$$\begin{aligned} A(z) &= \frac{\alpha}{2}[-N\gamma s^2 - 2s + \gamma(2 + N)] \\ B(z) &= \frac{\alpha}{2}\left[(1 + s)(1 - s) + \frac{2}{N}(1 - \gamma s)\right]. \end{aligned} \quad (45)$$

The stochastic differential equation including noise is then obtained as

$$\frac{1}{\alpha} \frac{ds}{dt} = -N\gamma s^2 - 2s + \gamma(2 + N) + \sqrt{\frac{2}{\alpha}\left[(1 + s)(1 - s) + \frac{2}{N}(1 - \gamma s)\right]}\xi(t). \quad (46)$$

A straightforward generalization to the multi-site case gives the following evolution equations

$$\frac{1}{\alpha} \frac{ds_i}{dt} = -N\gamma_i(s_i)s_i^2 - 2s_i + \gamma_i(s_i)(2 + N) + \sqrt{\frac{2}{\alpha}\left[(1 + s_i)(1 - s_i) + \frac{2}{N}(1 - \gamma_i(s_i)s_i)\right]}\xi_i(t). \quad (47)$$

1. Buluta, I. & Nori, F. Quantum Simulators. *Science* **326**, 108 (2009).
2. Feynman, R. Simulating physics with computers. *Int. J. Theor. Phys.* **21**, 467 (1982).
3. Mezard, M., Parisi, G. & Virasoro, M. A. *Spin Glass Theory and Beyond* (World Scientific, 1987).
4. Ausiello, G. *et al. Complexity and approximation* (Springer, 1999).
5. Das, A. & Chakrabarti, B. K. Quantum annealing and analog quantum computation. *Rev. Mod. Phys.* **80**, 001061 (2008).
6. Byrnes, T., Yan, K. & Yamamoto, Y. Accelerated optimization problem search using Bose-Einstein condensation. *New J. Phys.* **13**, 113025 (2011).
7. Yan, K., Byrnes, T. & Yamamoto, Y. Kinetic Monte Carlo study of accelerated optimization problem search using Bose-Einstein condensates. *Prog. Inf.* **8**, 39 (2011).
8. Utsunomiya, S., Takata, K. & Yamamoto, Y. Mapping of Ising models onto injection-locked laser systems. *Opt. Exp.* **19**, 18091 (2011).
9. Takata, K., Utsunomiya, S. & Yamamoto, Y. Transient time of an Ising machine based on injection-locked laser network. *New J. Phys.* **14**, 013052 (2012).
10. Yamamoto, Y., Takata, K. & Utsunomiya, S. Quantum computing vs. coherent computing. *New Generation Computing* **30**, 327 (2012).
11. Treutlein, P. *et al.* Quantum information processing in optical lattices and magnetic microtraps. *Fortschr. Phys.* **54**, 702 (2006).
12. Rojas, R. *Neural networks: A systematic introduction* (Springer, 1996).
13. Zurada, J. *Introduction to artificial neural systems* (West Publishing, 1992).
14. Hopfield, J. Neurons with graded response have collective computational properties like those of two-state neurons. *Proc. Natl. Acad. Sci.* **81**, 3088 (1984).
15. Silfvast, W. T. *Laser Fundamentals* (Cambridge University Press, 2004).
16. Duda, R. O., Hart, P. E. & Stork, D. G. *Pattern Classification* (Wiley, 2001).
17. Voter, A. F. *Introduction to the Kinetic Monte Carlo Method*, Springer, NATO Publishing Unit, In Press.
18. Glauber, R. J. Time-dependent statistics of the Ising model. *J. Math. Phys.* **4**, 294 (1963).

Acknowledgements

This work is supported by the Transdisciplinary Research Integration Center, Special Coordination Funds for Promoting Science and Technology, Navy/SPAWAR Grant N66001-09-1-2024, Project for Developing Innovation Systems of MEXT, the JSPS through its FIRST program, the Okawa Foundation, and NTT.

Author contributions

T.B. conceived the work, made the calculations, and wrote the paper. S.K. and K.Y. assisted in the calculations. Y.Y. oversaw the research.

Additional information

Competing financial interests: The authors declare no competing financial interests.

How to cite this article: Byrnes, T., Koyama, S., Yan, K. & Yamamoto, Y. Neural networks using two-component Bose-Einstein condensates. *Sci. Rep.* **3**, 2531; DOI:10.1038/srep02531 (2013).



This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported license. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-sa/3.0>