Science Advances
AAAS

advances.sciencemag.org/cgi/content/full/5/5/eaau0823/DC1

Supplementary Materials for

Experimental investigation of performance differences between coherent Ising machines and a quantum annealer

Ryan Hamerly*, Takahiro Inagaki*, Peter L. McMahon*, Davide Venturelli, Alireza Marandi, Tatsuhiro Onodera, Edwin Ng, Carsten Langrock, Kensuke Inaba, Toshimori Honjo, Koji Enbutsu, Takeshi Umeki, Ryoichi Kasahara, Shoko Utsunomiya, Satoshi Kako, Ken-ichi Kawarabayashi, Robert L. Byer, Martin M. Fejer, Hideo Mabuchi, Dirk Englund, Eleanor Rieffel, Hiroki Takesue, Yoshihisa Yamamoto

*Corresponding author. Email: rhamerly@mit.edu (R.H.); inagaki.takahiro@lab.ntt.co.jp (T.I.); pmcmahon@stanford.edu (P.L.M.)

Published 24 May 2019, *Sci. Adv.* **5**, eaau0823 (2019)
DOI: 10.1126/sciadv.aau0823

This PDF file includes:

Table S2. Problem-dependent constants α and β used in the relation $N_0 = \alpha + \beta \log_{10}(T/\mu s)$ for the success probability exponential $P = e^{-(N/N_0)^2}$.
References (*61*–*64*)

Section S1. D-Wave embeddings and J_c optimization

Native clique embeddings (*41*) are used for all SK problems, MAX-CUT problems on graphs with edge density 0.5, and MAX-CUT problems on varying-density graphs (Figs. 2C, 3B and 4A respectively in main text). The code to generate the embeddings is available on GitHub (*61*). Once an embedding is chosen, the embedding parameter J_c (ferromagnetic coupling between qubits in a chain) is tuned to maximize performance. In no cases does the optimal J_c depend on the annealing time.

Fig. S1 shows that the optimal J_c scales roughly as $N^{1/2}$ for SK problems and $N^{3/2}$ for MAX-CUT problems of edge density 0.5. In particular, the relations $J_c = 1.1N^{1/2}$ (SK) and $J_c = 0.047N^{3/2}$ (MAX-CUT) were used in Figs. 2C, 3B.

For graphs with variable edge density, it was shown in Fig. 4B that the optimal J_c scales as d for fixed N, with $J_c = 0.5d = 9.5x$ for $N = 20$ shown in the figure ($x = d/(N-1)$ is the edge density). Extrapolating this using the $N^{3/2}$ relation above (which holds for constant $x = \frac{1}{2}$), we

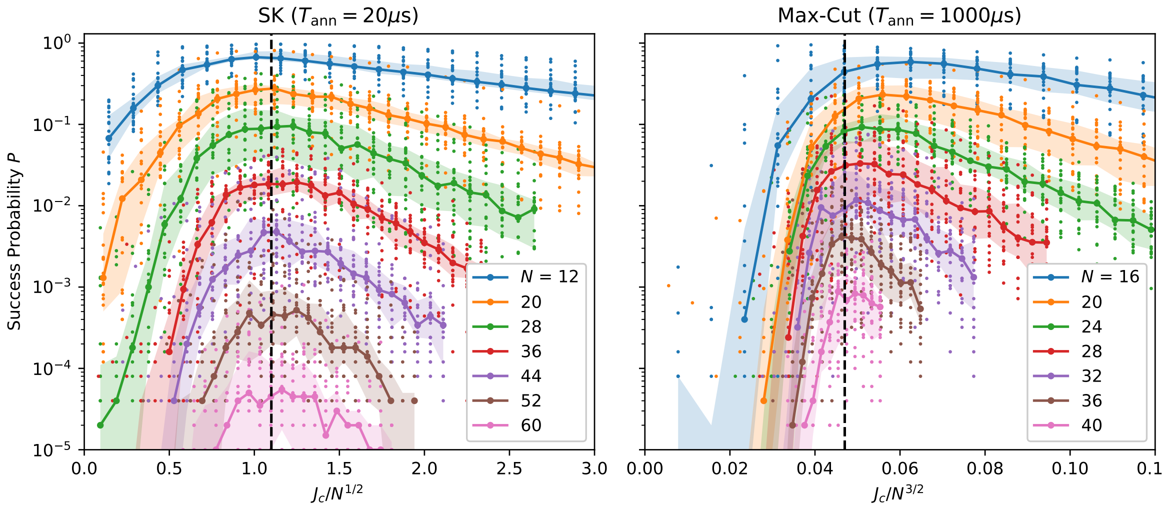

Fig. S1. D-Wave success probability for SK problems and MAX-CUT problems of edge density 0.5 as a function of problem size N and embedding parameter J_c.

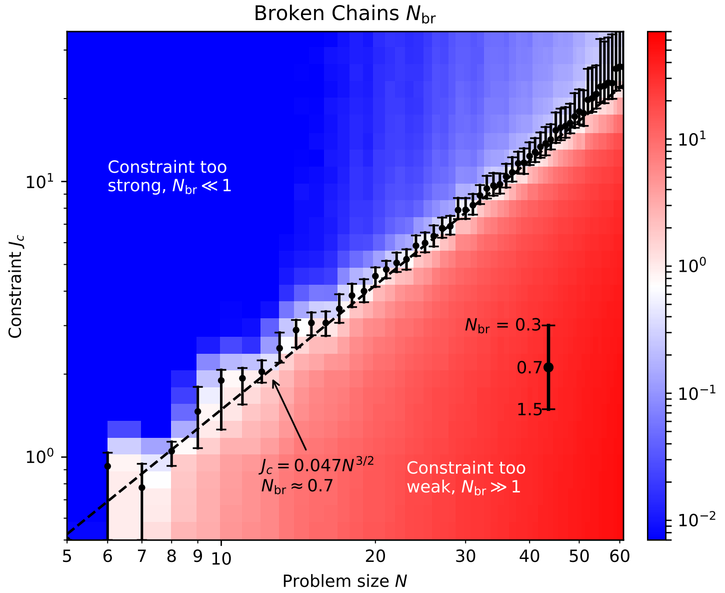

Fig. S2. MAX-CUT on graphs with an edge density of 0.5. Broken chains as a function of problem size N and embedding parameter J_c.

used $J_c = 9.5(N/20)^{3/2}x$, which is very close to the $J_c = 0.047N^{3/2}$ used for edge-density 0.5 graphs. The relation was also tested for $N = 30$ variable edge-density graphs and found to give the optimal J_c.

Fig. 3A of the main text suggests that the success probability is maximized when the number of broken chains is $N_{\mathrm{br}} \approx 0.7$. Plotting N_{br} as a function of N and J_c in fig. S2, we see that $N_{\mathrm{br}} \approx 0.7$ for a narrow range of J_c centered around the line $J_c = 0.047N^{3/2}$. For a wide range of N, this value of J_c also roughly maximizes the success probability (fig. S1).

The fact that dense MAX-CUT problems are optimally embedded when $N_{\mathrm{br}} = O(1)$ is an example of the general principle that J_c must neither be too strong nor too weak for a problem. If J_c is too small so that $N_{\mathrm{br}} \gg 1$, the constraint is not enforced effectively and thus the embedded problem can have a ground state that is different from the logical problem. Once $N_{\mathrm{br}} \lesssim 1$, increasing J_c further will not improve the computation significantly because all of the

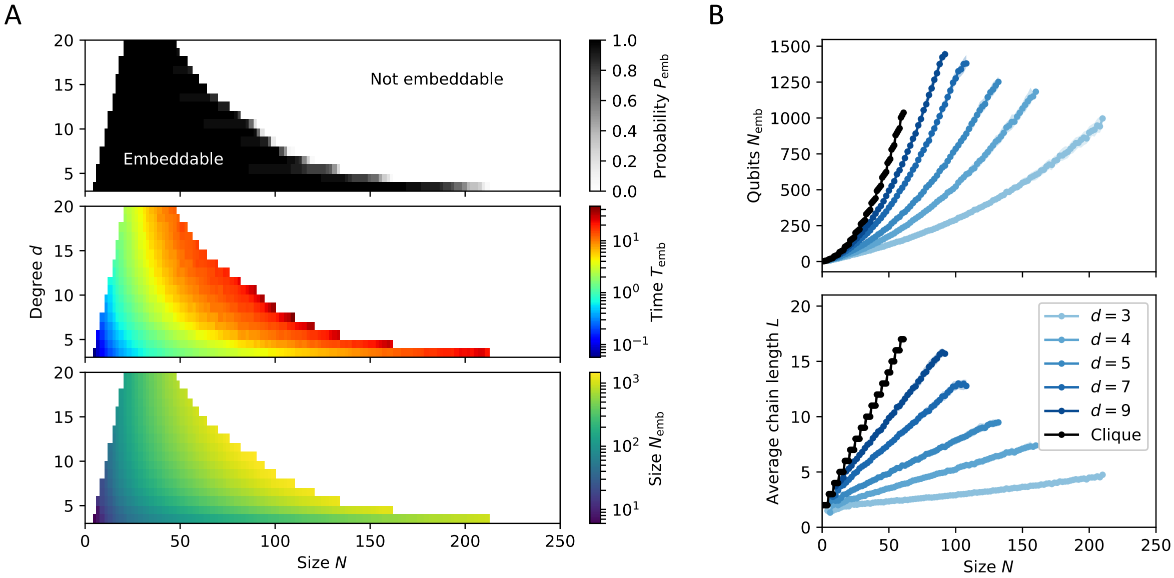

Fig. S3. Properties of heuristic embeddings for fixed-degree graphs. (**A**) Probability of finding an embedding using the heuristic, average time required to find an embedding, and number of physical qubits as a function of graph parameters (N, d) for fixed-degree graphs. (**B**) Number of qubits and average embedding chain length as functions of N.

constraints are already satisfied with high probability. Rather, it degrades performance because J_c maxes out the physical coupling on the chip so that logical couplings are scaled down as J_c^{-1}, which will correspondingly reduce the spectral gap of the (physical) Hamiltonian, and can also cause problems due to the finite bit precision and hardware imperfections of the D-Wave system.

For the sparse graphs, embeddings are found using the heuristic of Cai et al. (*21*), which is available as part of the D-Wave API toolkit. For each sparse graph instance, we attempt to generate 10 embeddings using the heuristic with a time-out of 60 seconds. The probability of finding an embedding is shown in fig. S3A (the $d = 3$ case is in agreement with (*21, Fig. 7*)). The time required to find an embedding (on average) and the number of physical qubits N_{emb} are also plotted in fig. S3A.

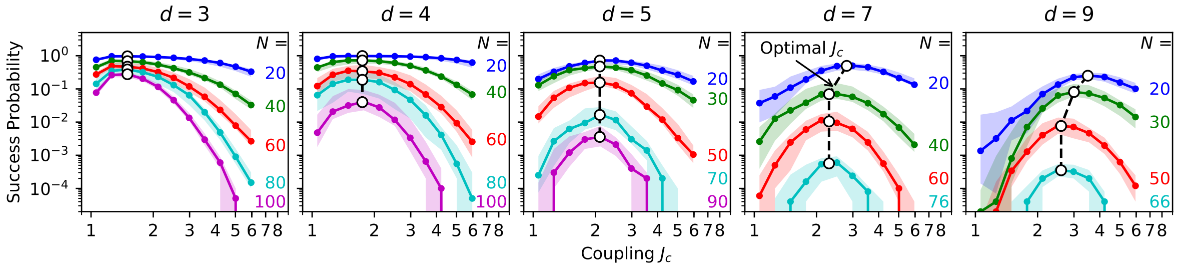

Fig. S4. Choice of optimal coupling for sparse graphs using the heuristic embedding.

Fig. S3B shows the number of physical qubits for graphs of degree $d = 3, 4, 5, 7, 9$ embedded using the heuristic, as well as the average chain length $L = N_{\rm emb}/N$. This is compared against the clique embeddings described above.

Because the heuristic embeddings differ markedly from clique embeddings, we do not use the formula $J_c = 9.5(N/20)^{3/2}x$ derived above. Rather, the optimal J_c is found by hand, running the quantum annealer for a range of N, d and J_c (fig. S4). We find that the optimal J_c is independent of N for sufficiently large N, while it increases slightly for small N for $d = 7, 9$. We interpolate using the curves of fig. S4 to find the embedding parameter used in the main text (Fig. 3C).

Section S2. CIM data and post-selection

The CIM is based on an OPO network, which is sensitive to optical phase fluctuations. During the course of operation, the phase of the injection beam will drift. This drift is slow compared to experimental timescales, but can become large if a calculation is run thousands of times.

To filter out out-of-phase computations (which always lead to the wrong answer), each CIM includes a phase-checking mechanism, albeit somewhat different for the NTT and the Stanford CIMs. We summarize both here.

In the NTT system, phase stability and calibration is implemented with a phase-check graph: the 2,048 spins in the CIM are partitioned into a 16-spin (unused) header, a 32-spin bipartite graph for phase checking, and a "frame" of 2,000 spins for the desired problem. Since $N \ll 2000$ for the problems in this paper, we can solve up to $\lfloor 2000/N \rfloor \approx 2000/N$ problems in parallel per frame. The coupling matrix J_{ij} has a block-diagonal structure (fig. S5A).

The couplings of the bipartite graph for phase-check are randomly set to $+1$ or -1 and the value of the phase-check Hamiltonian $H_{\rm PC} = \frac{1}{2}\sum_{ij} J_{ij}\sigma_i\sigma_j$ is computed after each run. If the optical phase is incorrect, we find $H_{\rm PC} > 0$ because the system couplings are reversed and the machine is trying to minimize $-H_{\rm PC}$. The top plot of fig. S5B shows the phase-check $H_{\rm PC}$ value (normalized to the maximum) as a function of time. $H_{\rm PC}$ drops sharply to a negative

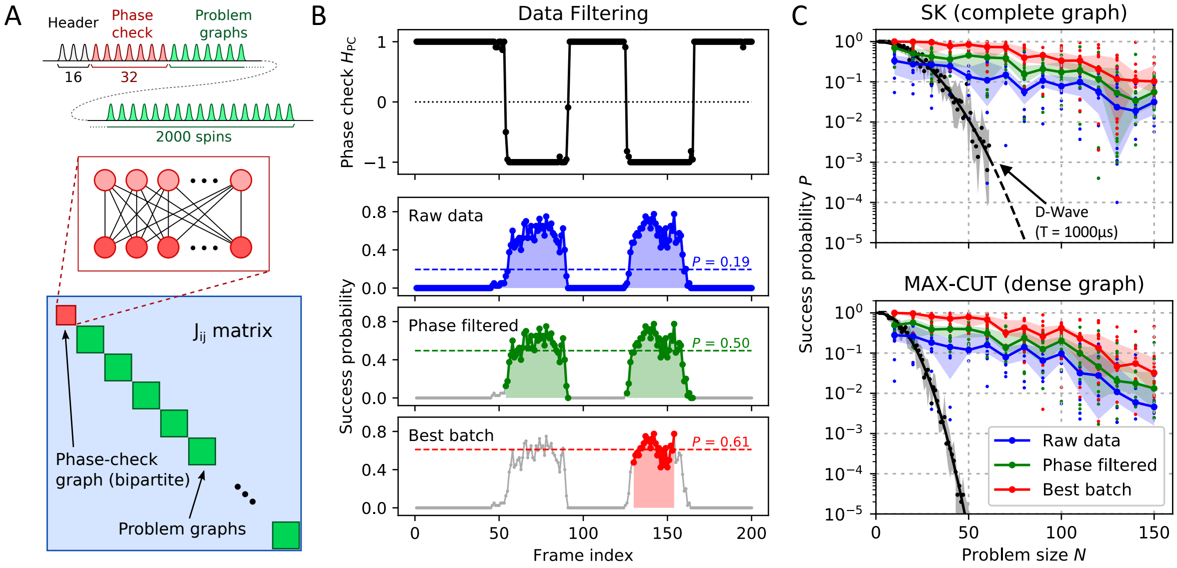

Fig. S5. Data filtering and post-selection in NTT CIM. (**A**) Partitioning of NTT CIM spins into a 16-spin header, a 32-spin phase-check graph, and 2,000 spins for problem graphs, and the resulting J_{ij} matrix. (**B**) Phase-check Hamiltonian $H_{\rm PC}$ as a function of time (frame index), and three post-selection techniques for inferring the success probability. (**C**) NTT CIM success probability for SK and $x = 0.5$ MAX-CUT problems as a function of post-selection method.

value when the CIM is in phase, making it a good proxy for the CIM phase.

In the bottom plots of fig. S5B, three data-filtering techniques are shown. Here we plot the free-running success probability (fraction of instances per frame in the ground state) for an $N = 50$ problem (40 trials running in parallel per frame). Averaging over all frames requires no post-processing, but gives a low success probability because we are including many trials when the machine is out of phase. Filtering on the phase-check graph (green curve) does significantly better; however, we are still averaging over the edges of the phase-check region where the system is only marginally in phase. Still better success probabilities can be found by looking for the best batch of 1,000 consecutive trials (25 consecutive frames) in the series (red curve). This generally corresponds to the the CIM working in its best condition: when the feedback signal is well in phase. This is the success probability we could expect from a well-engineered CIM where the optical phase, pump power, and other optical degrees of freedom have been sufficiently stabilized.

We compare the three post-selection methods in fig. S5C to show that our post-selection techniques give only a constant improvement in success probability, and this constant is never more than an order of magnitude. Thus, we can safely conclude that the CIM's performance advantage does not arise from cherry-picking good samples from the data. The "best batch" method (red curves in fig. S5) is used to process all CIM data reported in the main text.

The data collected from the Stanford CIM was also post-processed to select only the runs on the machine for which the optical setup was optimally stable. However, the procedure for post-selection was slightly different to that used for the data from the NTT CIM. In the case of the Stanford CIM, a recording of the homodyne measurement of the output pulses immediately before a run began was stored. During this recording phase, constant-amplitude pulses were injected into the cavity. If the entire system is phase-stable, then the recorded homodyne measurement results should not show large fluctuations from pulse to pulse. Furthermore, the particular value of the phase of the injected light is also relevant (not just that it is ideally constant), since the computation mechanism relies on interference of injected pulses with pulses in the cavity, and how much interference is obtained is partially determined by the phase of the injection pulses. We therefore post-selected not only for stability, but also for a particular mean value of the homodyne measurement results, which was determined on an instance-by-instance basis. The net effect of this post-selection procedure is to produce success probabilities that represent the probabilities one would obtain if the CIM was always phase-stable whenever a computation was run, and the phase was correctly calibrated for each problem instance.

The post-selected success probabilities were only on average $5\times$ higher than the success probabilities obtained when no post-selection was applied. This implies that even if one is pessimistic about the prospects of improvement to the optical phase stabilization of the CIM, and one assumes that the most stable the machine will ever be is as it was during the experiments reported in this paper, then at worst one should divide the success probabilities for the Stanford CIM reported in this paper by $5\times$. This gives the estimate for the expected success probabilities for a machine that has the same fundamental operating principle as the currently implemented CIM at Stanford, as well as the same experimental imperfections (including phase noise) that the current setup has.

The CIMs at Stanford and NTT were run on the same (randomly-chosen) Ising problems for $N \leq 100$ MAX-CUT (edge density $x = 0.5$) and SK (fully connected). The average success probabilities of the two machines agree to within a factor of 5 (fig. S6).

In order to compare the solution time T_{soln} with D-Wave, we need the physical annealing time for the CIM. A strict minimum for the annealing time is given by the product of the time between pulses (equal to $1/f$ where f is the pump repetition frequency), the size of the problem

N, and the number of round trips per run R

$$T_{\rm ann}^{(\rm min)} = \frac{NR}{f} \tag{S1}$$

This is the effective annealing time if perfect parallelization is achieved and all spins are used for logic (i.e. a negligible fraction of phase-check and dummy spins). Both Stanford and NTT CIMs use $R = 1000$ round trips.

However, the annealing time is generally longer than $T_{\rm ann}^{(\rm min)}$ because dummy spins are added to the cavity to compensate for the delays due to the DAC / ADC electronics in the feedback circuit and to give the FPGA more time to finish the coupling computation. This increases the cavity round-trip time and thus the annealing time.

In the NTT CIM, we used 5056 pulses in a 1-km fiber ring cavity as: 16-spin (header), 32-spin (phase check), 2000-spin (solve problem), 100-spin (blank), 2808-spin (free running in FPGA calculation time), 100-spin (blank). The pump repetition rate is 1 GHz and the round-trip time is 5μs. As only 2000 of 5056 pulses are used, even if perfect parallelism is employed, the annealing time is approximately $2.5\times$ longer than Eq. (S1), or $T_{\rm ann} = (2.5N)\mu$s, where N is the problem size. Fig. S6 plots the NTT CIM time-to-solution both with and without parallelism, to enable a fair comparison with the D-Wave annealer (we did not attempt to parallelize D-Wave to run multiple problems per anneal).

In the Stanford CIM, which did not employ parallelism due to its smaller number of spins, the annealing time is $T_{\rm ann} = 1.6\,\rm ms$ for all problems. The Stanford CIM (*24*) features a 320-m fiber ring cavity that contains 160 optical pulses (repetition rate 100 MHz), of which up to 100 can be used to encode Ising problems. The data in Fig. 2C come from the Stanford CIM, where the above annealing time combined with the formula $T_{\rm soln} = T_{\rm ann}\lceil\log(0.01)/\log(1-P)\rceil$ is used to calculate the time to solution.

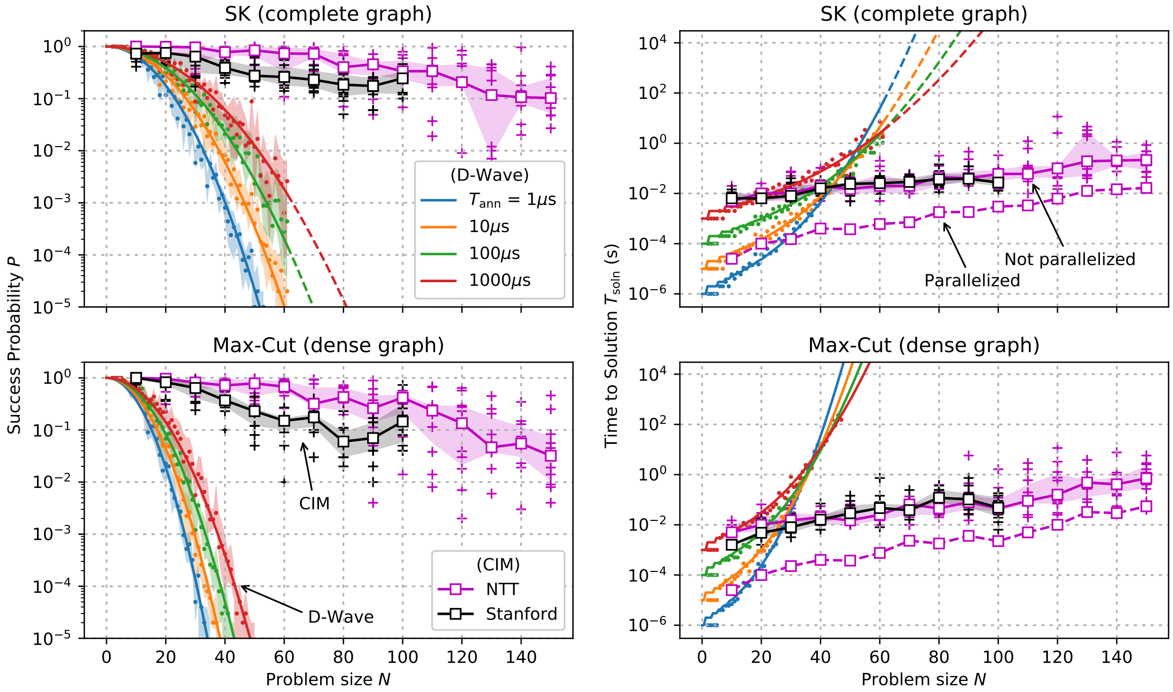

Fig. S6. Comparison of Stanford and NTT CIM performance for SK and dense MAX-CUT problems. Annealing time is 1000 round trips. D-Wave data for $T_{\rm ann} = 1$, 10, 100, and 1000μs are also plotted.

Section S3. C-SDE simulations of CIM

The CIM is a time-multiplexed synchronously-pumped OPO with measurement feedback coupling (fig. S7). It consists of a main loop (red) with a delay line for measurement and feedback (blue). Because the OPO is weakly coupled, we can treat this system using truncated-Wigner theory (*36*), which reduces the quantum dynamics to a set of c-number Langevin equations (c-SDEs). For OPOs with low single-pass gain at threshold, continuous-time stochastic differential equations can be employed (*29*). Since the round-trip gain of our system is high, a discrete-time model is needed, where the evolution of a single round trip (represented as a discrete-time c-SDE) consists of a series of seven discrete steps, each with its appropriate truncated-Wigner

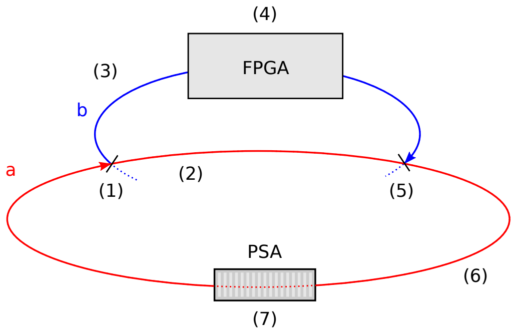

Fig. S7. Abstract schematic of measurement-feedback CIM. State variables are fields a, b. Each process (1)–(7) is described in Table S1.

Table S1. Seven steps in a single round trip for the measurement-feedback CIM and the appropriate truncated Wigner description. Constants are chosen to match the Stanford CIM: $\epsilon L = 3.6\times10^{-4}$, $p = 2.8\times10^{3}$, $\sin(\theta_m) = \sin(\theta_f) = \sqrt{0.1}$, $\sin(\theta_{L1}) = \sqrt{0.6}$, $\sin(\theta_{L2}) = \sqrt{0.5}$, $\sin(\theta_{L3}) = \sqrt{0.6}$.

Step	Description	Truncated-Wigner Model
1	Beamsplitter	$a_i\cos(\theta_m) + w_i^{(1)}\sin(\theta_m) \to a_i$ $a_i\sin(\theta_m) - w_i^{(1)}\cos(\theta_m) \to b_i$
2	Loss	$a_i\cos(\theta_{L1}) + w_i^{(2)}\sin(\theta_{L1}) \to a_i$
3	Loss	$b_i\cos(\theta_{L2}) + w_i^{(3)}\sin(\theta_{L2}) \to b_i$
4	Detection FPGA Modulation	$b_i \to x_i$ $\sum_j J_{ij}x_j \to y_i$ $C(F(t)y_i; y_{\max}) + w_i^{(4)} \to b_i$
5	Beamsplitter	$a_i\cos(\theta_f) + b_i\sin(\theta_f) \to a_i$
6	Loss	$a_i\cos(\theta_{L3}) + w_i^{(5)}\sin(\theta_{L3}) \to a_i$
7	PSA Gain	$p + w_i^{(6)} \to p_i$ $\epsilon L\sqrt{p_i^2 + a_i^2/2} \to B_i$ $e^{B_i}\big(1 + \frac{1}{2}(e^{2B_i} - 1)(1 - (1 + a_i^2/2p_i^2)^{-1/2})\big)a_i \to a_i$

description (Table S1).

Steps 1 and 5 are standard beamsplitters, whose input/output equations match those in classical optics. Steps 2, 3 and 6, which represent loss in the fiber loop and injection channel, can be modeled as beamsplitters with vacuum inputs. Homodyne detection converts the real part of b_i to a classical signal, discarding the imaginary part (only the real parts of optical signals a_i, b_i are treated in this model). The resulting classical signal is processed in the FPGA (step 4). The FPGA result is imprinted onto an optical field using a modulator, adding the vacuum fluctuations of the injected field. The modulation signal is clamped (function $C(z; z_0) \equiv \max(\min(z, z_0), -z_0))$ by the DAC maximum voltage (parameter $y_{\max}$ above). Step 7 is the $\chi^{(2)}$ phase-sensitive amplifier (PSA) gain. The formula is derived by solving the nonlinear field equations (*28, Sec. 2.2*) in a $\chi^{(2)}$ medium (*38, Eq. (8)*). All input vacuum fields are normally distributed random variables: $w_i^{(m)} \sim N(0, \frac{1}{2})$. We note that our model bears resemblance to mean-field annealing approaches to the Ising problem (*62*).

In this paper, the constants are chosen to match the experimental parameters of the Stanford CIM. The model is sensitive to the measurement and feedback couplings (θ_m, θ_f), but less sensitive to the overall loss, which simply increases the amount of quantum noise in the system by a small amount.

The Stanford CIM employs an "injection turn-on" scheme. We start with the feedback turned off and pump the OPO to slightly below threshold. Then the feedback term is slowly increased, lowering the effective threshold of the coupled-OPO system (*24*). This is opposite to the "pump turn-on" technique used in the NTT CIM and optical-feedback systems (*25, 30, 37*), where the coupling (and therefore threshold) stays fixed and the pump is increased. But the fundamental dynamics (bifurcation from squeezed vacuum driven by quantum noise) is the same, and we expect similar computational performance for both machines. The key degree of freedom is the *pump schedule* $F(t)$. For simplicity, we use a linear ramp

$$F(t) = F_{\max} \frac{t}{T_{\text{ann}}} \tag{S2}$$

which increases from zero to $F_{\max}$ over T_{ann} round trips (the runtime, or "annealing time", of the CIM, where $T_{\text{ann}} = 1000$ in the experiments in this paper.)

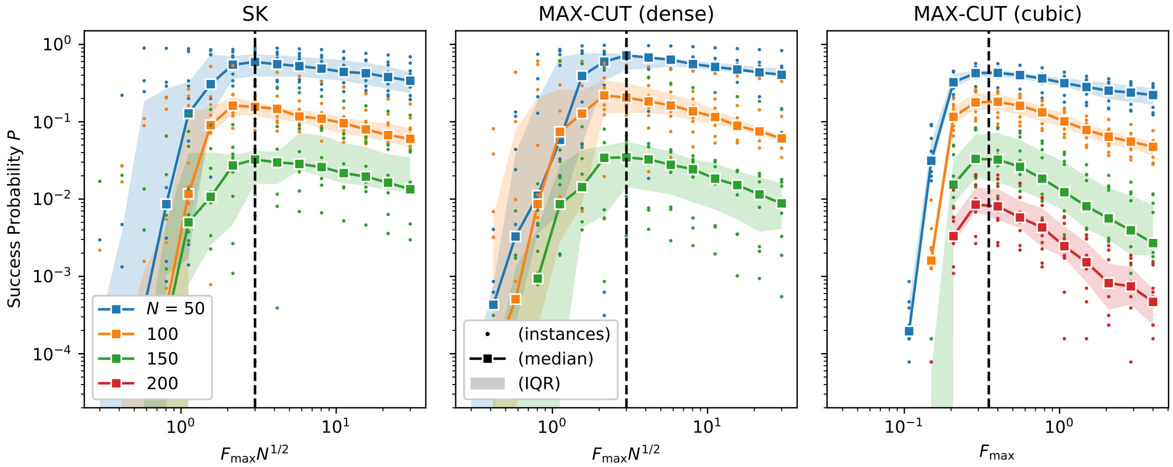

Fig. S8. Simulated CIM success probability as a function of $F_{\rm max}$ for the SK, dense MAX-CUT, and cubic MAX-CUT problems in this paper.

The free parameter $F_{\rm max}$ sets the scale of the feedback strength, and is tuned to maximize the success probability. Intuitively, one wants the feedback term y_i to be comparable to the circulating field a_i, as a small feedback term will not effectively couple the OPOs but a very large term will lead to spurious behavior that no longer maps onto the Ising problem (*29*). Since the injected field is proportional to $F(t)\sum_j J_{ij}a_j$, and since in random non-structured problems, the a_j are expected to be random, it is reasonable to assume that

$$F_{\rm max} \propto \left(\frac{1}{N}\sum_{ij}(J_{ij})^2\right)^{-1/2} \tag{S3}$$

For SK and dense MAX-CUT problems, Eq. (S3) predicts $F_{\rm max} \propto N^{-1/2}$, while for sparse problems, $F_{\rm max}$ should be a constant. This prediction is confirmed numerically in fig. S8. The success probability depends on both N and $F_{\rm max}$, and the peak is always located at $F_{\rm max}N^{1/2} =$ const for SK and dense MAX-CUT, and $F_{\rm max} =$ const for sparse MAX-CUT. The optimal $F_{\rm max}$

is roughly

$$F_{\max} = \begin{cases} 3.0N^{-1/2} & \text{(SK)} \\ 3.0N^{-1/2} & \text{(Dense MAX-CUT)} \\ 0.35 & \text{(Cubic MAX-CUT)} \end{cases} \tag{S4}$$

Using the optimal $F_{\max}$ in Eq. (S4), we simulate the CIM on all of the problems presented in the paper. Fig. S9 shows the result. Strictly speaking, the model is only applicable to the Stanford CIM, but both machines give similar performance that is roughly matches the c-SDE simulations.

C-SDE simulations are run to assess the effect of the annealing time and to determine the

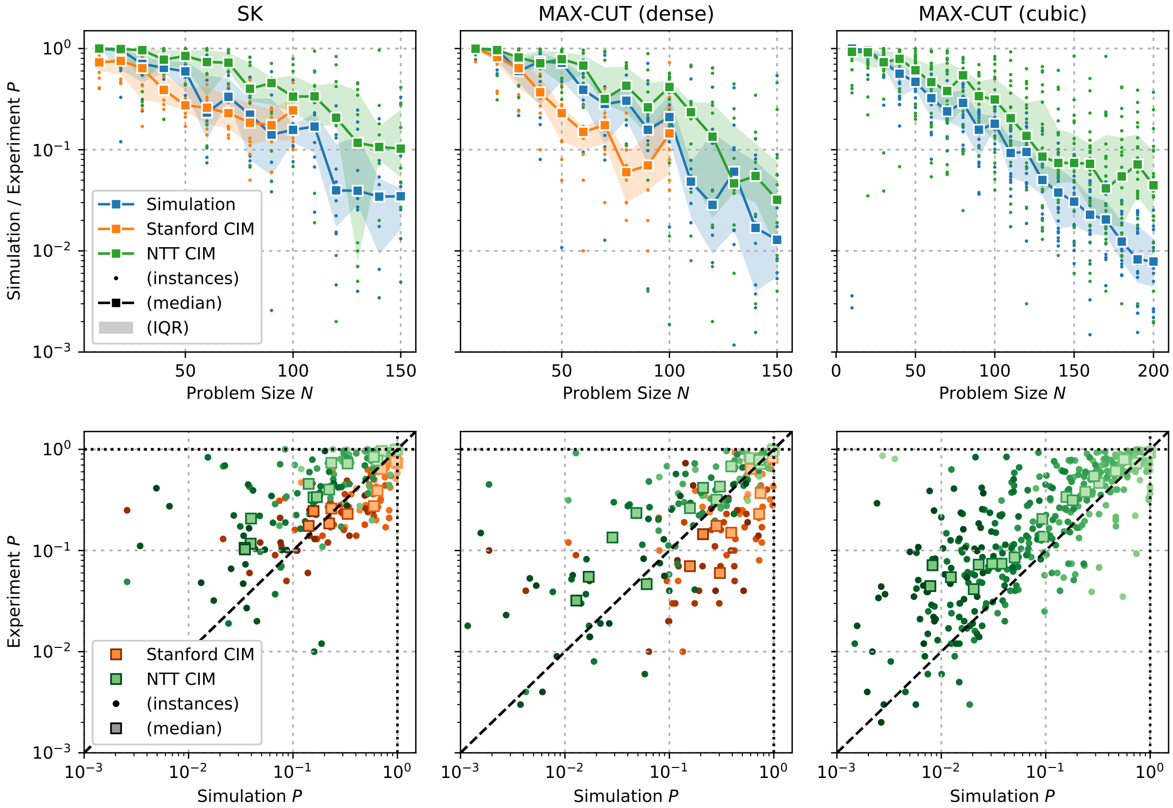

Fig. S9. Comparison of c-SDE simulations with experimental CIM data. Top: success probability as a function of problem size. Bottom: correlation plots between the c-SDE simulated CIM (x-axis) and experimental data (y-axis).

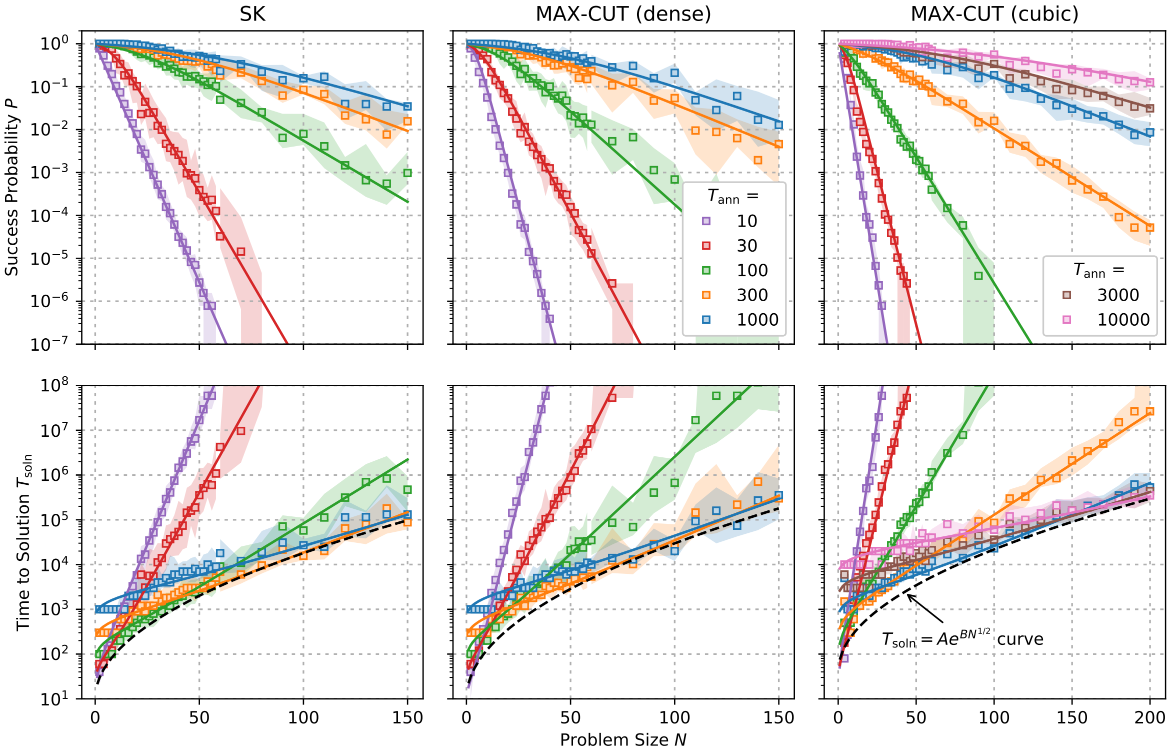

Fig. S10. Simulated CIM success probability and time to solution (in round trips) for SK and MAX-CUT problems. Squares are medians. Shaded region is IQR. Solid lines are fits to Eq. (S4).

optimal-annealing-time scaling of the CIM time to solution. Fig. S10 plots the success probability and time to solution (normalized to the round-trip time) for annealing times ranging from $T_{\text{ann}} = 10$ round trips up to $T_{\text{ann}} = 1000$. We see clear exponential behavior in the asymptotic limit, especially when T_{ann} is small. The plots fit reasonably well to a logistic curve intersecting the origin

$$P(N) = \frac{\alpha}{(\alpha - 1) + e^{\beta N}} \overset{N\to\infty}{\longrightarrow} \alpha e^{-\beta N} \tag{S5}$$

Likewise, the time-to-solution curves are rising exponentials in the large-N limit. When plotted on a logarithmic scale, the intercept of the curves increases with T_{ann}, while the slope decreases. This makes clear that, as in quantum annealing, there is a tradeoff between success probability and annealing time (*18*). The optimal time to solution is given by the lower envelope of these curves. In quantum annealing on glassy chimera-graph problems, an empirical scaling of $T_{\text{soln}} \sim \exp(O(N^{1/2}))$ has been reported (*18, 19, 49*). Curves of the form $Ae^{BN^{1/2}}$ are plotted in fig. S10 for reference. The rough fit suggests, but is not conclusive proof of, a similar time-to-solution scaling for coherent Ising machines.

Section S4. Optimal−annealing time analysis

To obtain the best performance of the D-Wave annealer under a fixed anneal schedule, we optimize T_{soln} with respect to the annealing time. For a fixed T_{ann} we find the square-exponential relation $P = e^{-(N/N_0)^2}$ for SK and dense MAX-CUT problems. Cubic MAX-CUT problems also fit this curve, especially for short anneals. In the range $T_{\text{ann}} \in [1, 2000]\,\mu\text{s}$ of admissible annealing times, we find $N_0 \approx \alpha + \beta \log_{10}(T_{\text{ann}}/\mu\text{s})$, where α and β are problem-dependent constants (Table S2).

The top graphs in fig. S11 plot the dependence of T_{soln} on T_{ann} for fixed N, allowing one to visualize the optimal annealing time for each problem size. The aforementioned fit agrees reasonably with the data for most problem sizes, although we make no claims about its validity outside the range of annealing times tested.

The lower plots in fig. S11 show the D-Wave time to solution in terms of problem size. The lower envelope of the fixed-T_{ann} curves, approximated as a line ($T_{\text{soln}} = Ae^{BN}$), gives the optimal time to solution for the DW2Q. For comparison, the optimal CIM time-to-solution

Table S2. Problem-dependent constants α and β used in the relation $N_{\ 0} = \alpha + \beta \log_{10}(T/\mu\text{s})$ for the success probability exponential $P = e^{-(N/N_0)^2}$.

	SK	MAX-CUT (dense)	MAX-CUT (cubic)
α	15.24	10.05	53.45
β	2.81	1.39	22.15

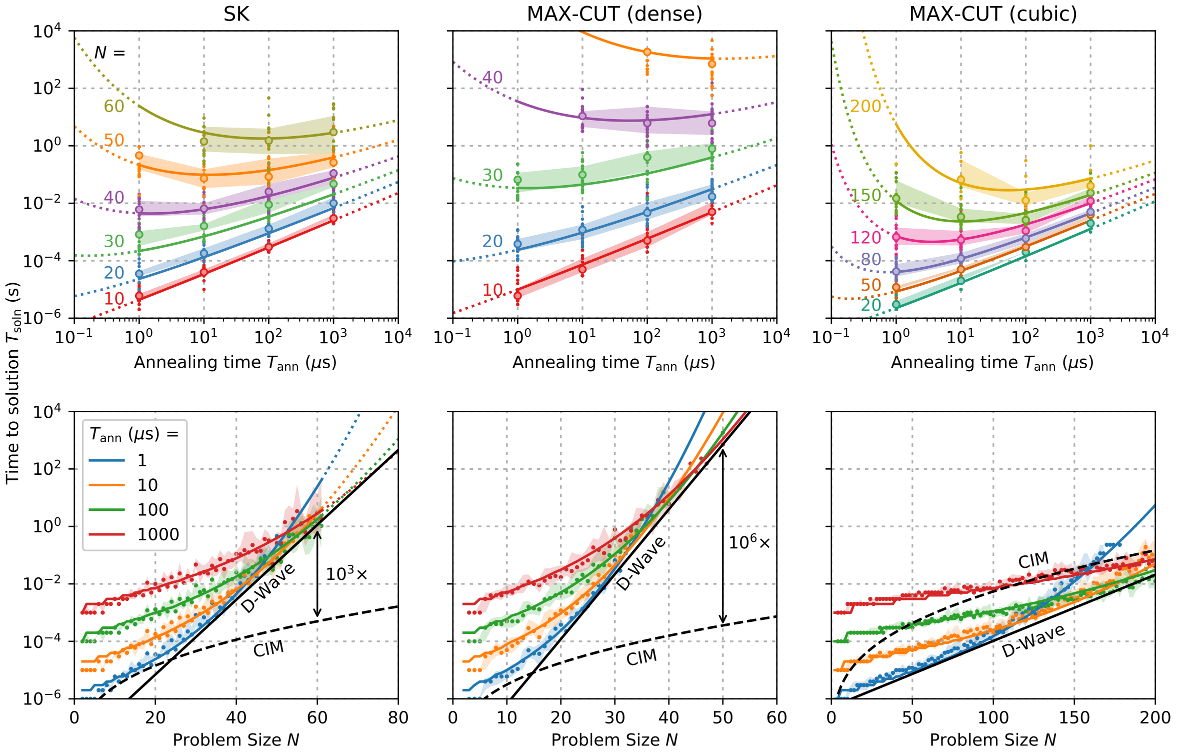

Fig. S11. Time-to-solution analysis for D-Wave at optimal annealing time. Top: D-Wave time to solution T_{soln} as a function of the annealing time for fixed problem sizes, illustrating the optimal anneal time. Bottom: T_{soln} as a function of problem size, with optimal anneal-time curve approximated as a line. CIM time to solution at optimal anneal time (from Fig. S10) plotted for comparison (NTT CIM with parallelization, round-trip time $(2.5N)\mathrm{ns}$).

obtained in fig. S10 is also plotted. The CIM round-trip time used is the value for the NTT CIM accounting for parallelization: $(2.5N)\mathrm{ns}$; see Sec. S2.

Since the optimal annealing time lies in the experimentally accessible regime $[1, 2000]\mu\mathrm{s}$ for only a limited range of problem sizes ($N \in [40, 60]$ for SK, $[30, 50]$ for dense MAX-CUT), it is difficult to estimate the precise shape of the lower envelope by looking at fig. S11. While the data is consistent with an exponential, it is also consistent with many other curves, so we caution against naively extrapolating these curves. Nevertheless, at optimal annealing time, the CIM is substantially faster ($\geq 10^3\times$ for SK, $\geq 10^6\times$ for dense MAX-CUT) at the upper

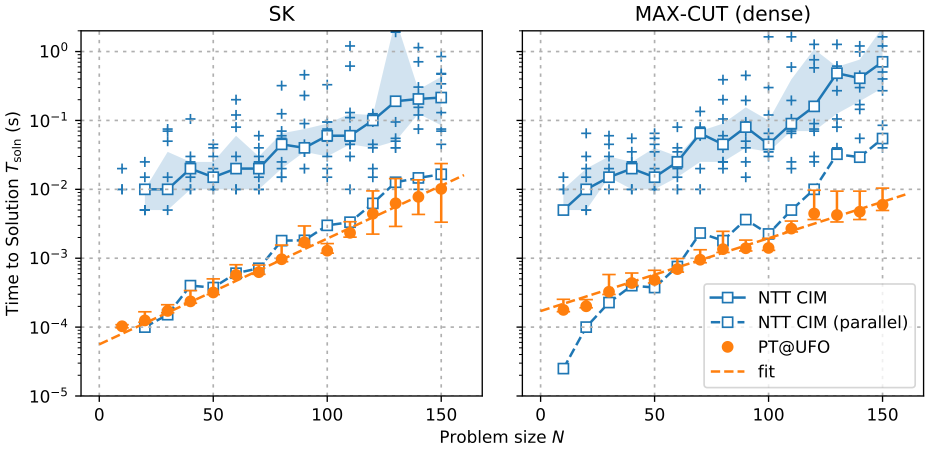

Fig. S12. CIM time to solution compared against the parallel tempering algorithm implemented in the Unified Framework for Optimization (UFO). The error bars for PT@UFO corresponds to the minimum and maximum value of time to solution for that specific size. All UFO runs were performed on Intel Xeon CPU E5-1650 v2 (3.50GHz).

end of experimentally measured problem sizes, while D-Wave has a performance advantage of 10–100× for cubic MAX-CUT, although this advantage narrows with larger problem sizes.

Section S5. Performance of parallel tempering

Parallel tempering is a state-of-the-art classical optimization technique that has been shown to perform well on a variety of Ising problems (*49, 63, 64*). Here, we include results provided by Salvatore Mandrà, which made use of the implementation of parallel tempering in the NASA/TAMU Unified Framework for Optimization (UFO). The comparison shows respectable performance of NTT's parallel CIM compared with PT@UFO. We see that NTT's parallel CIM comes close to the performance of PT@UFO for the SK problem instances in the size range considered, and is also close on the MAX-CUT problem up through the middle range of problem sizes considered, but diverges for larger problem sizes.