Population annealing with topological defect driven nonlocal updates for spin systems with quenched disorder

David Cirauqui [1,2] Miguel Ángel García-March,[3] J. R. M. Saavedra [1]
Maciej Lewenstein [2,4] and Przemysław R. Grzybowski[2,5]

[1]*Quside Technologies SL, Mediterranean Technology Park, 08860 Castelldefels (Barcelona), Spain*
[2]*ICFO–Institut de Ciències Fotòniques, The Barcelona Institute of Science and Technology, 08860 Castelldefels (Barcelona), Spain*
[3]*Instituto Universitario de Matemática Pura y Aplicada, Universitat Politècnica de València, 46022 València, Spain*
[4]*ICREA–Institució Catalana de Recerca i Estudis Avançats, E-08010 Barcelona, Spain*
[5]*Institute of Spintronics and Quantum Information, Faculty of Physics, Adam Mickiewicz University, Umultowska 85, 61-614 Poznań, Poland*

Population annealing, one of the currently state-of-the-art algorithms for solving spin-glass systems, sometimes finds hard disorder instances for which its equilibration quality at each temperature step is severely damaged. In such cases, therefore, one cannot be sure about having reached the true ground state without vastly increasing the computational resources. In this work, we seek to overcome this problem by proposing a quantum-inspired modification of population annealing. Here we focus on the three-dimensional random plaquette gauge model, whose ground-state energy problem seems to be much harder to solve than that of the standard spin-glass Edwards-Anderson model. In analogy to the toric code, by allowing single bond flips we let the system explore nonphysical states, effectively expanding the configurational space by introducing topological defects that are then annealed through an additional field parameter. The dynamics of these defects allow for the effective realization of nonlocal cluster moves, potentially easing the equilibration process. We study the performance of this method in three-dimensional random plaquette gauge model lattices of various sizes, and we compare it against population annealing. With that we conclude that the newly introduced nonlocal moves are able to improve the equilibration of the lattices, in some cases being superior to a normal population annealing algorithm with a computational resource investment that is 16 times higher.

I. INTRODUCTION

Spin-glass models and, more generally, spin systems with quenched disordered interactions, consist of a set of particles or nodes, each taking a value from a range of possibilities (binary in the case of the Ising models [1]) with randomly valued interactions between them, where the topology of the connections can be arbitrarily complex. Due to spin frustration and randomness, such models exhibit tremendously rich behaviors and have long been one of the main focuses of statistical physics. Analytically challenging even at the mean-field level [2] and numerically hard for nonplanar topologies [3], their study continues to rely mostly on numerical simulations.

In the past few decades, spin-glass models have proven useful beyond their applications in physics, for example as models related to the behavior of neural networks [4,5] and to the real-world optimization problems, especially for those that accept a quadratic unconstrained binary optimization (QUBO) mapping [6,7]. In the QUBO picture, an optimization problem is mapped to a set of binary variables such that its ground-state configuration encodes the optimal solution to the original problem. QUBO mapping applications range from fields as diverse as protein folding [8,9], to logistics [10–12] and many others, and therefore devising ways to efficiently find the ground states of spin-glass models constitutes a goal of utmost importance for both scientific and industrial problems.

Many physics-inspired algorithms have been devised for that end. One way is to use standard Markov chain Monte Carlo (MCMC) methods to mimic thermal fluctuations and utilize various temperatures to explore the rugged energy landscapes of spin-glass systems. Inspired by the process of slow thermal cooling and heating in the metallurgic industry, simulated annealing (SA) [13] was one of the first algorithms of that type to prove its value in real-life applications. As an evolution of SA, and taking advantage of massively parallel computing platforms such as graphics processing units (GPUs) available nowadays, the recently proposed population annealing (PA) [14,15] is an extended ensemble Monte Carlo algorithm that combines SA's thermal annealing with a population of independent replicas of the system under study. Thermal annealing is then significantly sped up by the replicas' population resampling procedure done according to the Gibbs distribution. For thermally equilibrated and sufficiently large replica population, resampling allows us to change the population temperature in a single step while the population remains approximately equilibrated. This has been found to improve, in many cases, the performance of the algorithm over SA [16]. Although PA is a sequential Monte Carlo method [17], it still relies on a basic METROPOLIS algorithm [18] to independently equilibrate each of the replicas after every resampling step in order to ensure that the correlations

between replicas introduced by resampling are eliminated. The METROPOLIS algorithm is known to slowly equilibrate, e.g., in the critical region or in the case of rugged landscapes [19], in which it gets trapped by local minima. The latter is problematic in spin systems with quenched disorder at low temperatures, for example when studying three-dimensional Edwards-Anderson (EA) spin-glass systems [20]. This fact can sometimes cause PA to encounter hard disorder instances for which population cannot be properly equilibrated [17]. One way to overcome the METROPOLIS algorithm deficiencies is to use cluster moves. In the case of uniform, ferromagnetic Ising models, the celebrated Swendsen-Wang [21] and Wolff algorithms [22] work very efficiently. The case of spin-glass models is more problematic as clusters may grow fast, potentially spanning the whole lattice. The Houdayer cluster algorithm [23] of isoenergetic moves was designed specifically for those models. It offers substantial speedup in two dimensions, and it has recently been efficiently adapted in three dimensions (and other connection topologies), albeit only in specific temperature and frustration windows [24]. Therefore, the search for efficient cluster moves in Monte Carlo algorithms for spin-glass models, especially moves compatible with massively parallel computations, remains an important challenge.

On the other hand, while the new advances on quantum annealing (QA) machines [25,26] pose them as firm contenders to efficiently solve these kinds of QUBO problems, they are still far away from surpassing classical computation. As a counterpart, new quantum-inspired algorithms for classical computers have recently been developed, taking advantage of the current dominant position of such platforms. Simulated quantum annealing (SQA) [27] constitutes the paradigmatic example within this field, in which a classical simulation of the adiabatic transition of a wave function from the initial state to the ground state of the spin-glass model is performed. In this case, the simulation of a coherent exploration of the Hilbert space upon changing the Hamiltonian replaces the simulation of thermal annealing.

In this work, we propose a quantum-inspired method to improve the exploration of the optimization and quenched disorder spin problems in PA by the evolution (annealing) of certain local constrains. Taking inspiration from the quantum toric code [28], the creation, movement, and annihilation of topological defects within the lattice allow for the conduction of nonlocal moves. In two dimensions, such moves can be performed on the standard spin-glass random-bond Ising model with an expanded configurational space (i.e., allowing for topological defects), and they are equivalent to cluster updates. In three dimensions we propose a numerically hard model, equivalent to the random three-dimensional (3D) random plaquette gauge model (RPGM) [29], whose expanded configurational space allows for respective topological defects and nonlocal moves, even though the analogy with the spin clusters is not that straightforward. The evolution of such defect states starts with relaxed constrains (free defect creation) and is controlled externally by the increase of a field parameter added into the original Hamiltonian that progressively penalizes their proliferation. The preparation of the starting thermally equilibrated replica population is straightforward in any temperature as defect constrains are relaxed. Therefore,

annealing can now be conducted starting from any temperature, and it can be performed in a two-dimensional parameter space (temperature and defect control parameter). In this regard, we define an adaptive step procedure that is able to optimize the annealing of the system in this parameter space, gaining flexibility over PA, which is seen to increase the thermalization of the systems without compromising other related and desired properties. Importantly, the proposed method is itself a variation of PA, and is therefore massively parallelizable as well, leading to parallelizable effective nonlocal moves.

One should note that an efficient general solution to the numerically hard (NP-complete in the worst-case scenario) spin-glass and optimization problems, with increasing lattice sizes, is most probably impossible on classical computers. Yet even for systems and lattice sizes amenable for current computational resources, the hard disorder cases are the limiting factor. Different annealing methods (temperature in PA, adiabatic wave-function changes in SQA, topological defect constrains in the present variation of PA) have the potential to address different hard disorder cases. Indeed, that is what we observe: disorder cases that are difficult for standard PA are usually not that hard in our defect driven PA.

The paper is organized as follows. In Sec. II we first give a description of PA and discuss a metric for the confidence of the solution found corresponding to the global minimum, which will eventually allow us to compare the newly proposed method. We then introduce the random 3D Ising gauge theory model in plaquette representation that we named the random field wall (RFW) model, and we describe its topological defects dynamics, to finally propose the defect driven population annealing (DPA) modification. In Sec. III we first discuss the hardness of the RFW model, we compare it against the EA one, we apply DPA to RFW lattices of linear sizes $L = 4, 6, 8$, and we compare the results against one of the state-of-the-art algorithms, PA, which, in the task of finding ground states of spin glasses, is equivalently efficient to other methods such as parallel tempering (PT) [16]. Further, in this same work PA was found to reach the same ground states as other completely different types of algorithms, such as the hybrid genetic algorithm [30]. Finally, we conclude this work with a discussion in Sec. IV.

II. METHODOLOGY

A. Population annealing

Population annealing is a generalized ensemble extension of SA in which a family of a total of R_0 replicas of the same system are independently simulated in parallel. PA starts considering all R_0 replicas at infinite temperature, at which equilibration is easy but encountering the global ground state is difficult, and anneals the population towards a low temperature at which equilibration is difficult but the probability of finding the ground state is higher. We denote the annealing schedule as $S_\beta = [\beta_0, \ldots, \beta_{N_T}]$, with β the inverse temperature and $\beta_i < \beta_{i+1}$ $\forall i$, where i labels each resampling and equilibrating step. Equilibration of the replicas is performed through a Markov chain Monte Carlo (MCMC) method for a number of sweeps N_{sweeps}, and then the population is resampled according to the Gibbs distribution, which is

known to enhance the efficiency of the algorithm compared to SA. This resampling step includes elimination and proliferation of replicas that somewhat resembles the selection part of a genetic algorithm, in which replicas with a lower energy (that is, better fitness) are set to have several offspring, while those having a higher one tend to be eliminated. When resampling from an inverse temperature β to β', the normalized weight for replica j is defined as

$$\tau_j(\beta, \beta') = R_0 \frac{e^{-(\beta' - \beta)E_j}}{\sum_r e^{-(\beta' - \beta)E_r}}, \qquad (1)$$

where R_0 is the initial number of replicas of the system, and the index r runs over all replicas, and we choose its number of descendants n_j as

$$n_j = \begin{cases} \lceil \tau_j \rceil & \text{with probability } P_{\text{ceil}} = \tau_j - \lfloor \tau_j \rfloor, \\ \lfloor \tau_j \rfloor & \text{with probability } P_{\text{floor}} = 1 - P_{\text{ceil}}. \end{cases} \qquad (2)$$

This choice of probability minimizes the variance of n_j and lets the population's size vary around a mean value R_0 with a fluctuation of $\sqrt{R_0}$. This choice reduces the correlation between replicas in the descendant population [17].

An additional and important advantage of PA is that it yields an estimate of the free energy of the simulated system at no additional cost. For the annealing schedule S_β described above, the estimated free energy at inverse temperature β_k, $\tilde{F}(\beta_k)$, is [31]

$$-\beta_k \tilde{F}(\beta_k) = N_s \ln 2 + \sum_{i=0}^{k-1} \ln Q(\beta_i, \beta_{i+1}), \qquad (3)$$

where N_s is the total number of spins in the system, and $Q(\beta, \beta')$ is the normalization factor used by PA in the computation of the resampling weights, Eq. (1):

$$Q(\beta, \beta') = \frac{1}{R} \sum_r e^{-(\beta' - \beta)E_r}. \qquad (4)$$

B. Confidence on the solution found

Due to their heuristic nature, algorithms such as SA or PA may not reach the global, *true* ground state of a certain model at hand. In such cases, one has to conform with a measure of the likelihood of the solution found being a global minimum, or as close as possible to it. Specifically for PA, we can assess this likelihood by means of measuring two different parameters on the final population of the algorithm's run. The first one, g_0, is the fraction of replicas that, in the final population, are in the same minimum energy state detected during the entire simulation, $g_0 = N_0/R$, with N_0 counting the number of times that this state appears in the final stage. Given that PA can accurately estimate the free energy of the simulated system, g_0 can be estimated as well from its definition

$$g_0 = \frac{2e^{-\beta E_0}}{Z(\beta)} = 2e^{\beta(E_0 - F)}, \qquad (5)$$

with E_0 the ground-state energy of the system. The value obtained by measuring g_0 from N_0 in the simulation should coincide with Eq. (5) in the limit of infinite replicas [16]. The

second considered parameter is the effective number of surviving families N_{eff}, which measures the number of replicas that have found the lowest energy state *independently*. The effective number of surviving families can be measured as

$$N_{\text{eff}} = \exp[S_f], \qquad (6)$$

with

$$S_f = -\sum v_j \log v_j \qquad (7)$$

the family entropy, and v_j the fraction of replicas present at the end of the simulation that descend from replica j in the initial population. Effectively, family entropy is a measure of the equilibration or thermalization of the sample [17].

In the case of PA, the only way to be sure of having found the ground state would be to use an infinitely large population or an infinitely slow annealing process. So if the size of the final population R is large enough and the same lowest-energy state appears frequently in thermal equilibrium, we are able to confidently state that the found configuration is the global minimum. Therefore, the higher both parameters g_0 and N_{eff} are after a simulation, the higher is the confidence we might have in the solution found being the global ground state of the system. The problem in PA is then to make sure that those two parameters are high enough, and particularly the latter, since it has been seen that it sometimes encounters hard instances for which thermalization is harder, and that therefore exhibit a very low final family entropy [17,32]. In the following, we explore problems of different hardnesses, with a focus on those that PA finds the hardest to solve.

C. Bond and wall representations of the models and toric code topological defects

For simplicity, consider first a standard 2D spin lattice with only nearest-neighbor interactions, defined by the Hamiltonian

$$H = \sum_{\langle i,j \rangle} J_{ij} s_i^z s_j^z, \qquad (8)$$

with periodic boundary conditions. We can shift to its bond representation by applying the change of variable $\sigma_b^z = s_i^z s_j^z$:

$$H = \sum_b J_b \sigma_b^z. \qquad (9)$$

It is important to stress that, even if the Hamiltonian looks deceptively easy, it does not correspond to free particles, as it must be understood along with the constraints associated with the set of feasible configurations (see below). In the bond picture of a spin lattice, the new binary variables σ_b represent the state of the bond between two spins, which depends on their relative alignment instead of that of the spins themselves. The bond is either said to be up (the spins connected by it have the same value, thus $\sigma_b^z = 1$) or down (the spins are not aligned, and $\sigma_b^z = -1$). Note that therefore a bond configuration can be translated into a spin one with up to a decision of the value of an initial spin, which corresponds to the up-down degeneracy of the spin picture.

However, in contrast to the spin representation, not all $\{\sigma_b^z\}$ configurations represent a physical state. For a bond
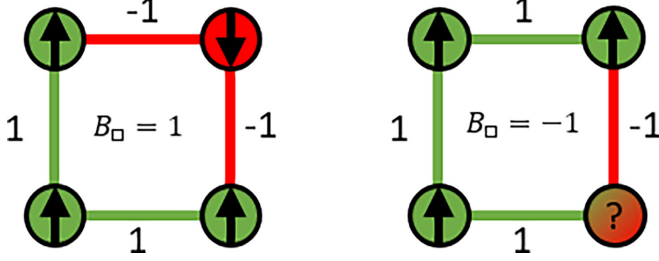
FIG. 1. Physical states in the bond representation are characterized by $B_\square = 1$. When a plaquette yields $B_\square = -1$, spins located on its vertices cannot be unequivocally determined, and therefore the configuration does not represent a physical state of the Ising model. We thus say it contains a defect.
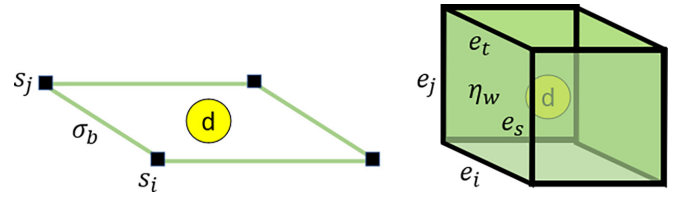

FIG. 2. The 3D random field wall (RFW) model is a generalization of the random 2D bond model. By extruding a plaquette and all its elements onto an extra dimension, we obtain a cube surrounded by edge spins that are 4-to-4 connected by a wall. The site enclosed inside the cube can hold a defect.

configuration to represent a physical state of the Ising model, the products of the bonds in any close path must be 1 (see Fig. 1). To account for this, we introduce plaquette and line operators:

$$B_\square = \prod_{b \in \square} \sigma_b^z, \tag{10}$$

$$B_| = \prod_{b \in |} \sigma_b^z, \tag{11}$$

where $b \in \square$ and $b \in |$ are the indices of the bonds pertaining to a given plaquette (illustrated by a square, $\square$, due to their spatial disposition) and line (illustrated by a line, $|$, due to their spatial disposition), respectively. Note that $B_\square$ is equivalent to TC's B operator, while the spin flip is equivalent to TC's star operator, A_s,

$$s_i^x = \prod_{b \in +} \sigma_b^x, \tag{12}$$

where $b \in +$ are the indices of the bonds surrounding spin s_i (again, illustrated by a cross, $+$, due to their spatial disposition around the spin). The condition for being able to recover a proper spin configuration translates into assuring that (i) all plaquettes yield 1, and (ii) all straight lines across the lattice (both horizontal and vertical) yield 1.

If this condition is not fulfilled for a given plaquette or line, we say that it contains a (topological) defect (see Fig. 1). In what follows, we refer to such configurations as nonphysical.

Let us now shift to an appropriate 3D extension of such a 2D Ising model with topological defects in bond representation. The Kramers-Wannier dual [33] to the 3D uniform Ising model is the 3D Ising gauge theory model [34] where spins are located on lattice edges and are subjected to plaquette interactions on each cube face. The latter model can be represented as a "wall model" where the walls are the new binary variables taking values of cube face plaquettes, the plaquette interaction J is now a field acting on walls, and walls are not independent variables but are subjected to the constraint that in each cube the product of all six planes must equal 1. The given configuration of walls determines energy and represents all 2^N gauge-equivalent states (where N is the number of vertices), thus in this representation entropy

is greatly reduced. To visualize the 3D wall model, consider an extrusion onto the new dimension of the elements of the 2D bond model described so far. This means that we add an extra dimension to all of them, and therefore degrees of freedom change as follows:

(i) Spins (points) become edge spins (lines).

(ii) Bonds (lines) become walls (planes).

And with respect to operators (constraints):

(i) Plaquettes (which may contain a defect) become cubes (which may contain a defect).

(ii) Lines (which may contain a defect) become planes (which may contain a defect).

As we previously had a total of four spins connected by bonds (each bond connecting two single spins) and forming a plaquette surrounded by four bonds that could have a defect in it, we now have 12 edge spins connected by walls (each wall connecting four edges) and all of them forming a cube surrounded by six walls that can contain a defect as well (see Fig. 2). In this case, the change of variable between edge spin and wall models is

$$\eta_w^z = e_i^z e_j^z e_s^z e_t^z, \tag{13}$$

where $e_{\{i,j,s,t\}}$ are the edge spins surrounding the wall η_w.

Plaquette and line operators translate to cube and plane operators as

$$B_c = \prod_{w \in c} \eta_w^z, \tag{14}$$

$$B_p = \prod_{w \in p} \eta_w^z, \tag{15}$$

respectively, and again the physicality of the lattice is imposed by $B_c = 1 \,\forall c$ and $B_p = 1 \,\forall p$. In Eqs. (14) and (15), $w \in c$ and $w \in p$ are the indices of the walls defining the cube c and the plane p, respectively.

The Hamiltonian of the 3D Ising gauge model in the wall representation reads

$$H = \sum_w J_w \eta_w^z. \tag{16}$$

However, analogously to Eq. (9), this is not a free-particle model. It has to be understood along with the constraints associated with the set of feasible configurations given by $B_c = 1 \forall c$ and $B_p = 1 \forall p$. Of course, instead of imposing these constrains on the phase space, they can be included in the Hamiltonian through Lagrange multipliers. In the following

sections, we will employ a similar approach to introduce topological defects into the model.

If we now consider quenched disorder signs of plaquette interactions in the 3D Ising gauge model, we get the 3D random plaquette gauge model, which appeared recently in a study of stability of topological quantum memory [29] based on the TC. Following Monte Carlo simulations revealed that a small concentration of negative sign plaquettes drives such a system from the "Higgs phase" to the disordered "confining phase" [35,36]. Furthermore, the phase structure of the 3D RPGM model was studied using Wu-Wang duality [37]. Here we study the 3D random plaquette gauge model with Gaussian disorder for plaquette interactions in the "wall" representation, which we will refer to as the 3D random field wall (RFW) model. This model is not dual to the 3D Edwards-Anderson spin-glass model (also called the random-bond Ising model), which consists of up/down spins arranged in a cubic lattice with only nearest-neighbor couplings, whose values are drawn from a normal distribution. In fact, the 3D RFW relation to the spin-glass physics is unclear. Replica symmetric mean-field solutions for random gauge theories [38] predict the occurrence of the "gauge-glass" phase, although it is probably realized in higher dimensions. Still, we found the problem of finding the ground state of the 3D RFW model to be numerically hard (harder than the 3D Anderson-Edwards spin-glass model for a typical disorder case) and therefore suitable for testing PA and our algorithm. Finally, we note that the original 2D spin lattice can be seen as a special case of a 3D RFW lattice with a single layer of cubes, and in which we restrict the spin and wall flips to those perpendicular to the lattice plane.

D. Defects dynamics

Besides the usual spin-flip movement, Eq. (12) [and Eq. (17) for the 3D case, see below], we can now introduce TC topological defects into the lattice by allowing single-bond (-wall) flips with the operator σ_b^x (η_w^x), thus extending the configurational space beyond the one representing physically feasible solutions. Note that this operator not only can create a pair of defects but can also make them move independently of each other around the lattice and eventually annihilate them if they happen to collide. This defect propagation introduces cluster-based, nonlocal dynamics into the MCMC simulation: if the pair of defects created follow a longer closed path until they collide, all the spins enclosed in it are effectively updated as a cluster, since all surrounding bonds have been updated (see an example in Fig. 3).

From this perspective, the spin-flip operator can be seen as creating a pair of defects and moving them around a given spin (vertex); see Fig. 4. Analogously to the 2D case, for the RFW the edge spin flip consists of updating all walls surrounding a given edge spin,

$$e_\mu^x = \prod_{w \in +} \eta_w^x, \tag{17}$$

where $w \in +$ are indices of the walls located around the edge spin μ. Analogously to the 2D case, here the notation $+$ is
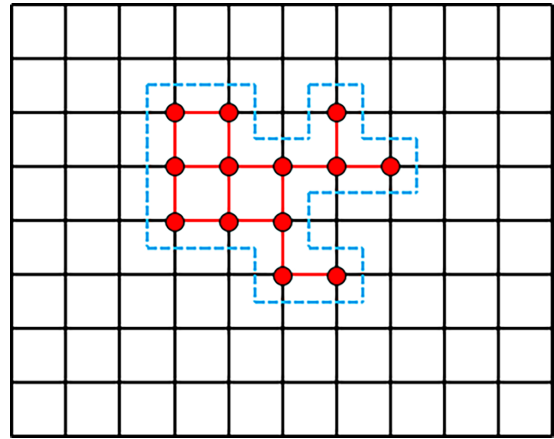

FIG. 3. Allowing single-bond flips enables cluster, nonlocal updates in the lattice. By applying single bond-flips, a pair of defects can be created and moved around the lattice. When they collide and annihilate, the closed path they have followed (blue dashed line) effectively realizes a nonlocal cluster update of the spins it encloses (red dots). The inner bonds are left unchanged in such an update.

intended to illustrate the spatial disposition of the four bonds surrounding the spin on the vertex μ.

E. Defect-driven population annealing

We are now ready to introduce our algorithm, which combines PA's features with the nonlocal cluster updates discussed before. We refer to this modification as defect driven population annealing (DPA). To this aim, we introduce an additional field parameter κ, with which we control the appearance of defects in the lattice by assigning an energy penalty to each of these generated defects. First of all, note that at any time we can count the number of cube and plane defects within a lattice, $N_c^{(D)}$ and $N_p^{(D)}$, with the help of operators Eqs. (14) and (15), as

$$N_c^{(D)} = N_c - \sum_c B_c, \tag{18}$$

$$N_p^{(D)} = N_p - \sum_p B_p, \tag{19}$$

where the summations run over all cubes and all planes in the lattice, and N_c and N_p are the total number of cubes and planes, respectively. Accounting for this penalty in Eq. (16),
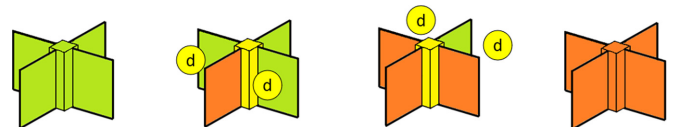

FIG. 4. Graphical representation of an edge spin flip. Green and orange label ± 1 values for the walls and the edge spin, while yellow means that there are defects in the adjacent cubes and therefore the edge cannot be unequivocally determined. We stress that, even if only one edge spin is shown for clarity, each of the walls is surrounded by (is coupling) four edge spins, and thus the shown defects must be seen as pertaining to a whole cube (for instance, they could move to the cube on its top instead).
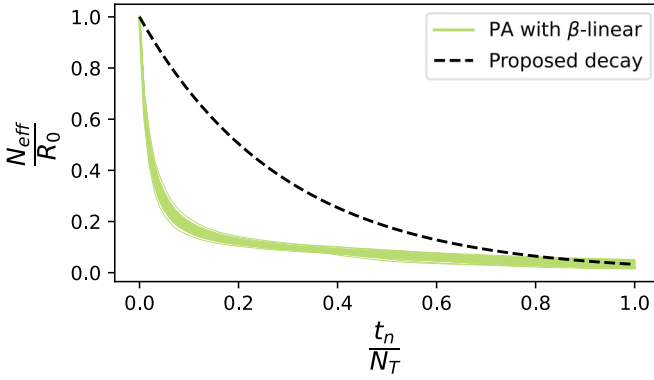
FIG. 5. Entropy-preserving annealing. Comparison between the decay in family entropy obtained in PA simulations of $L = 4$ RFW lattices with a β-linear annealing schedule (green lines) and the objective, implemented decay (black dashed line). Each blue line is the decay obtained for a different disorder. Although final family entropy may seem very similar in this scale, its fluctuations in PA simulations describe differences between hard and easy disorder cases.

the final Hamiltonian of the RFW model in the unconstrained phase space (i.e., including nonphysical states) reads

$$
H = \sum_w J_w \eta_w^z + \kappa \left[\left(N_c - \sum_c B_c \right) + \left(N_p - \sum_p B_p \right) \right].
$$

(20)

Note that the κ term incorporates constraints into the Hamiltonian similarly to the Lagrange multiplier method. However, we use it differently. By initializing the simulation at $\kappa = 0$ we ensure that the system can be readily thermalized at any chosen temperature and acquires a lot of defects. Then, by changing this parameter to high enough values, we make sure that the constraints are fulfilled at the final population. Therefore, at the final stages of the annealing process those configurations containing defects are penalized and rare. The configurations without defects correspond to the physical configurations of the model we want to solve. As in PA, we run the simulation on a population of R_0 independent replicas, and at each step in the annealing schedule we let them evolve with an MCMC procedure to ensure thermal equilibration, after which a resampling step is also carried out. Since at every resampling step both β and κ are updated, the normalized weights now are

$$
\tau_j(\beta, \beta', \kappa, \kappa') = R_0 \frac{e^{-[\beta' E_j(\kappa') - \beta E_j(\kappa)]}}{\sum_r e^{-[\beta' E_r(\kappa') - \beta E_r(\kappa)]}},
$$

(21)

with $E(\kappa) = \sum_w J_w \eta_w + \kappa[N_c^{(D)} + N_p^{(D)}]$.

1. Family entropy-preserving annealing

As has been discussed, a bad thermalization of the system is characterized by a low final value of family entropy, which is reduced every time a resampling step is carried on the population. In standard population annealing, in which a linear schedule in β is classically used in the literature [16], the family entropy tends to follow a geometric decay, Fig. 5.

However, the difference between hard and easy disorder cases manifests itself in subtle changes in the decay line and, importantly, its end value. Since our defect driven population annealing algorithm utilizes annealing in a two-parameter space, namely the temperature and the κ, we use a different schedule. To maximize the family entropy value during the whole simulation, we propose to adapt the annealing process such that, at each resampling step, a constant portion of the number of surviving families is lost. This way the family entropy follows an exponential decay towards an objective value, which we can tune externally. To this end we implement an adaptive step procedure for both annealing schedules, S_κ and S_β. Before each resampling step $(\beta, \kappa) \rightarrow (\beta', \kappa')$, we compute the optimal values of κ' and β' needed in order for the desired portion of families to be lost, such that the obtained family entropy follows the specified decay

$$
\frac{N_{\text{eff}}(t_n)}{R_0} = \left(\frac{N_{\text{eff}}^{\text{desired}}}{R_0} \right)^{\frac{t_n}{N_T}},
$$

(22)

where t_n is the simulation step, $N_{\text{eff}}(t_n)$ is the number of surviving families at that time, and $N_{\text{eff}}^{\text{desired}}$ is the objective number of surviving families at the end of the simulation.

2. Remaining defects

It may happen that only annealing through κ is not enough to ensure that we get rid of the defects at the end of the simulation and that some of them get stuck in potential traps, thus ending up with physically unfeasible lattices. In such cases, two additional procedures can be applied. On the one hand, an N-fold way algorithm [39] among the already existing defects only (thus they can be moved and annihilated, but no new defects can be created) is used to increase their mobility and speed up their dynamics. On the other hand, an attraction potential between defects can be used to reduce the probability of them getting stuck inside potential traps while forcing them to collide. This is done by the introduction into Hamiltonian Eq. (20) of the additional term

$$
H_G = G \sum_{i=1}^{N_{\text{def}}} \sum_{j=0}^{i-1} \frac{1}{D_{ij}^\alpha},
$$

(23)

where G is a negative constant that can be tuned, N_{def} is the number of defects present in the lattice, D_{ij} is the euclidean distance between defects at sites i and j, and α can be used to make the interaction potential shorter- or longer-ranged.

3. Outline of the algorithm

To conclude, in this section we present a pseudocode of the defect driven population annealing algorithm.

As in regular PA, the algorithm starts by randomly initializing the configurations of each replica in the population. The annealing schedules, S_β and S_κ, last a total of N_β steps, and then for each annealing step the algorithm estimates the next point in the schedules, (β', κ'), such that the family entropy obtained from the resampling follows the proposed exponential decay. After resampling, each replica r is thermalized with a total number of sweeps N_{sweeps} at (β', κ'). To this end, a number of N_{spins} spin updates and wall updates are proposed and accepted with a METROPOLIS algorithm. In case the

ALGORITHM 1. Defect driven population annealing.

1: **for** each replica r **do**
2:     r ← initialize
3:     **for** i_β in N_β **do**
4:         $N_{eff}^{(desired)}$ ← ProposedDecay[i_β]
5:         Estimate (β', κ') such that $N_{eff}[i_\beta] \approx N_{eff}^{(desired)}$
6:         Resample($\beta, \beta', \kappa, \kappa'$)
7:         **for** each replica r **do**
8:             **for** sweep in N_{sweeps} **do**
9:                 **for** n in N_{spins} **do**
10:                     s ← U($0, N_{spins}$)
11:                     Metropolis(r, s, β', κ')
12:                     **if** $i_\beta < i_\beta^{(critic)}$ **then**
13:                         w ← U($0, N_{walls}$)
14:                         Metropolis(r, w, β', κ')
15:                     **else**
16:                         **if** $N_{defects}$(r) $\neq 0$ **then**
17:                             NFoldWay (r, β', κ')
18:                         **else**
19:                             w ← U($0, N_{walls}$)
20:                             Metropolis(r, w, β', κ')
21:         $\beta \leftarrow \beta'$
22:         $\kappa \leftarrow \kappa'$

simulation is reaching its end ($i_\beta \geqslant i_\beta^{(critic)}$), and a given lattice still has sites containing topological defects, the dynamics are accelerated with an NFoldWay algorithm instead of using a regular METROPOLIS one, such that no more defects can be created but only moved around the lattice [with the additional gravitational term, Eq. (23)] so that their probability of closing a path and annihilating increases. After all replicas are thermalized, the next annealing steps start until the schedules reach their maximum length N_β.

III. RESULTS

A. Hardness of the RFW model

One of the standard and most widely used spin-glass models is the so-called Edwards-Anderson model [20], for which reliable results on its three-dimensional version have been reported for sizes of up to $L = 10$ [16], or equivalently, a total of $N_s = 1000$ spins. Bigger lattice sizes have also been explored in the literature, but with a lower degree of confidence on the quality of the thermalization achieved [17]. On the other hand, reducing the frustration of the problem solved by considering easier disorder distributions can indeed allow for the exploration of larger lattices. For example, by considering 3D lattices with nearest neighbors and couplings distributed uniformly as $J = \pm 1$, which in effect greatly reduces the spin frustration and has the potential of exponentially increasing the degeneracy of the ground state, results for sizes of up to $L = 40$ have been obtained [40]. As discussed earlier, the RFW model does not correspond to the standard 3D-EA model but allows for the effective dynamics of topological defects. It is well known that multispin interacting models (with *multi* referring to more than two) can in general be very difficult to sample and optimize. The RFW model, Eq. (16), is equivalent to the 3D random Ising gauge model with four-spin
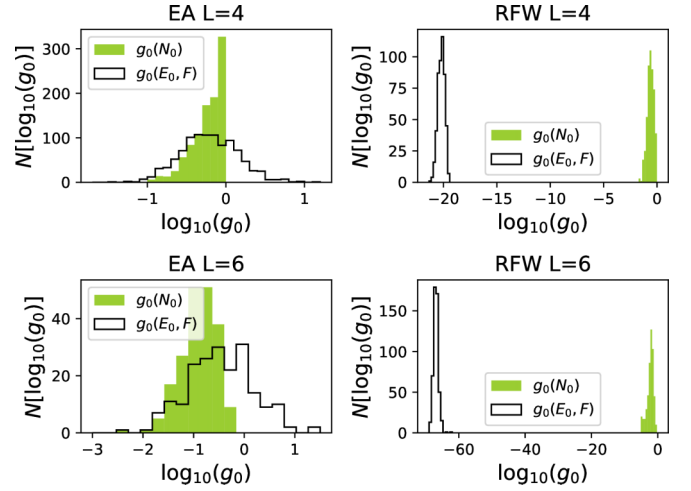

FIG. 6. RFW models are computationally harder than their EA counterparts. Histograms of the values of g_0 obtained with the two discussed independent measures, for various disorder instances of EA (left panels) and WRF (right panels), of sizes $L = 4$ (top panels) and $L = 6$ (bottom panels).

plaquette interactions, so indeed it can be expected to be difficult. On the other hand, the form of the interactions depends on the representation. The 2D Edwards-Anderson model, Eq. (9), which lacks hard instances, looks like a free-particle model with four-particle constraints in the bond representation. Similarly the RFW model in the wall representation looks like a free-particle model with six-particle constraints. Furthermore, the 2D EA model in the bond representation can be seen as a special case of a 3D RFW lattice with a single layer of cubes, and in which we restrict the spin and wall flips to those perpendicular to the lattice plane. Because of that, the multispin nature of the constraints in the wall representation of the RFW model is not an obvious argument that the RFW model is more difficult to simulate than the already hard 3D EA model. Also, there are not many literature results about it. Therefore, in this section we explore the hardness of the RFW model, given by Eq. (16), by comparing it against the standard 3D EA one. The RFW model is, as we show below, much more difficult to solve for typical disorder cases, as many more computational resources must be used in order to properly thermalize it. This difference in computational hardness can be seen straightforwardly when standard PA is used in the two models. Since PA yields independent measures for both g_0 and F, if the system is truly equilibrated their values should be properly related through Eq. (5). We solve several instances of EA and RFW lattices of sizes $L = 4$ and 6, and we measure $g_0(N_0)$ from the final population and $g_0(E_0, F)$ from the estimated free energy, using the same amount of computational resources for both models at each size. We observe how, while the measures of g_0 for the EA lattices are mostly similar in both studied sizes, RFW lattices yield much poorer results, Fig. 6. Further, the difference between $g_0(N_0)$ and $g_0(E_0, F)$ appears to increase for larger RFW lattices but not for EA ones, hinting that the increase in hardness with size is more dramatic in the former model. In what follows, if the contrary is not specified, the presented results for g_0 are obtained by measuring $g_0(N_0)$.
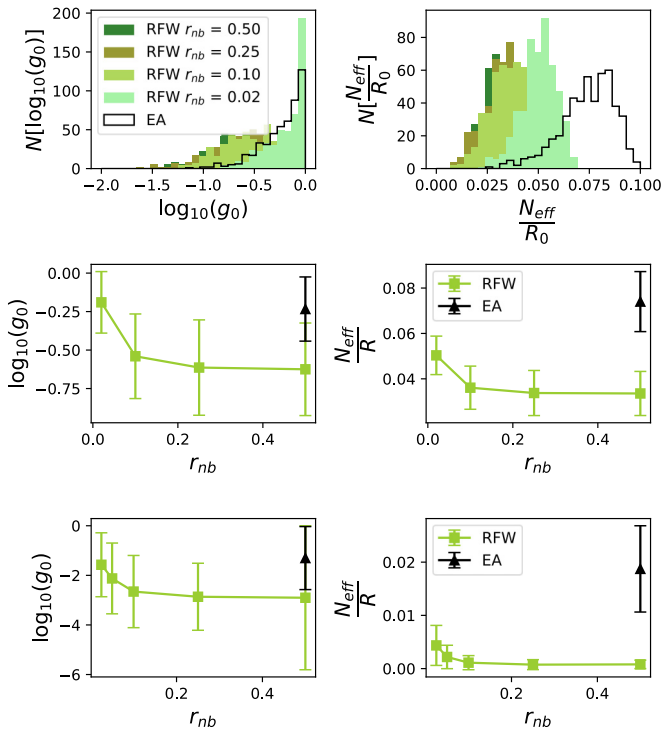
FIG. 7. The hardness of RFW models against EA clearly emerges when varying the frustration level of the solved lattices. Top panels: histograms of g_0 and $\frac{N_{\text{eff}}}{R_0}$ obtained for $L = 4$ RFW lattices of various ratios of negative bonds, r_{nb}, along with the ones obtained for Edwards-Anderson lattices. Center panels: mean values and standard deviations of such histograms, as a function of r_{nb}. Bottom panels: mean values with standard deviations of the same histograms but for $L = 6$ lattices. All simulations are obtained with PA with the same set of parameters. Legends in the left and right panels are the same but are not shown to improve visibility of the plots.

Another way we can take a look at the comparably much greater difficulty of the RFW model is by reducing the spin frustration created by the disorder by tuning the ratio of negative-valued bonds it contains. To do so, we generate normal Gaussian distributions, take their absolute value, and change the sign of a randomly selected portion of them. The lower the ratio of negative bonds that the disorder instance has, the lesser the amount of frustrated spins it can potentially have, and thus the easier it will be to find its minimum energy configuration, eventually reaching the trivial limit of a ferromagnetic disorder. We consider three different ratios of negative bonds, $r_{\text{nb}} = \{0.25, 0.10, 0.02\}$, and we solve a total of 1000 disorder instances of RFW lattices of size $L = 4$ for each of them with PA. In the top panels of Fig. 7 we show the histograms of the measured parameters discussed in previous sections, g_0 and $\frac{N_{\text{eff}}}{R_0}$ for such cases, along with those obtained for unbiased Gaussian disorders (equivalent to $r_{\text{nb}} = 0.5$) and those obtained for $L = 4$ lattices under the Edwards-Anderson model. Again, we use the same parameters for all of them. In the center panels of Fig. 7, we plot the dependence of the mean values of such histograms with r_{nb}, with error bars showing its standard deviations. We observe how the histograms tend to shift to higher values for smaller r_{nb} and therefore for easier disorders.

We highlight two main consequences from the results shown in Fig. 7. On the one hand, we confirm that, as discussed earlier, higher values of g_0 and the family entropy are correlated with higher confidence in the solution found being the global ground state and, therefore, it can be used as a metric to this end. On the other hand, it shows how strikingly harder the RFW model is compared to typical Edwards-Anderson instances with uncorrelated disorders, obtaining comparable values of g_0 only when restricting the frustration of disorders to a ratio of negative bonds of about 2%, while still getting much worse values for the family entropy. Looking at the same curves for $L = 6$, we observe pretty similar results; see the bottom panels of Fig. 7. Although the g_0 obtained for the EA model is approximately equal to the one obtained for RFW with about 2% negative bonds, the family entropy is still far from it even for such low-frustration cases. It is also worth noting that, for $L = 6$, this difference is even greater than for $L = 4$.

Further, we implement $L = 8$ RFW lattices with Gaussian disorder ($r_{\text{nb}} = 0.5$), and we attempt to solve them with standard PA using a large amount of computational resources. For equally large EA lattices, the parameter set $(R_0, N_T, N_{\text{sweeps}}) = (5 \times 10^5, 200, 10)$ is reported to be *more-than-adequate resources* in [16], where R_0 is the initial population size, N_T is the number of temperatures in the annealing schedule, and N_{sweeps} is the number of sweeps carried on per temperature (each sweep consists of attempting N_{spins} MC updates on the lattice). For comparison, in our simulations we used $(R_0, N_T, N_{\text{sweeps}}) = (2.6 \times 10^5, 500, 20)$ and repeatedly solved 20 different disorder realizations in order to see how many times the algorithm converged to the same minimum energy state. Among all of them, the convergence ratios between independent runs varied from a minimum of 0% (in 11 out of the 20 studied disorders) to a maximum of 12% (in only three of them).

In light of these conclusions about the hardness of the present model, in what follows we restrict ourselves to a thorough study of RFW lattices of $L = 4$ and 6 with purely Gaussian disorders (that is, $r_{\text{nb}} = 0.5$). Finally, we present some preliminary results for $L = 8$ lattices as well. As the exposed results suggest, the strict equilibration of RFW lattices constitutes a problem out of our current computational capabilities for most of the cases. We therefore stress that we do not focus on achieving a proper equilibration of the lattices *per se*, but rather on its use as a measure of the confidence of the solution found being the global ground state of the system.

B. Comparison between PA and DPA

To gain insight into how the different algorithms perform, we solve, for each lattice size, several disorders with couplings randomly drawn from a Gaussian distribution. When we talk about PA, we explicitly work with Eq. (16), which has the constraints built in within the phase space. On the other hand, when we talk about DPA, we work with Eq. (20), which allows for the relaxation of the constraints and thus the introduction of topological defects through the κ term. For each algorithm we will compare the obtained g_0 and N_{eff}. However, we do not use the comparison between $g_0(N_0)$ and $g_0(E_0, F)$ as a metric for the equilibration of the system simulated with the

two algorithms. This is because DPA effectively explores a larger phase space at the initial stages of the simulation, and the amount of replicas needed to accurately estimate free energy is larger. This is further exacerbated as equilibration is generally poor for the RFW model, as seen in Fig. 6. This implies that for a given population size, DPA can yield worse F estimations than PA. Instead, we directly compare the obtained g_0 and N_{eff} as a metric for confidence in the ground-state solution. For a fair comparison, we should spend the same amount of computational work W on both PA and DPA simulations. For that, we define the computational work W as the total number of MCMC updates attempted during a simulation, namely

$$W = N_{sweeps} N_T R_0, \qquad (24)$$

with N_T the total number of steps in the annealing schedule, N_{sweeps} the number of sweeps per temperature, and R_0 the initial number of replicas. Note that, as discussed above and according to Eq. (2), the population size at step i, R_i, fluctuates during the simulation and therefore a more accurate formula would be $W = N_{therm} \sum_{i=0}^{N_T} R_i$. However, since we use a re-sampling protocol that minimizes fluctuations in population size, we can accurately approximate $R_0 \approx N_T^{-1} \sum R_i$ and use the former. It should also be noted that the contribution of the resampling steps to the total computational work can be neglected. To study a fair comparison between both algorithms, we always use the same initial number of replicas and the same number of temperatures in the annealing schedules. In PA we implement the widely used linear in inverse temperature schedule, $S_\beta = [0, 5]$. Recently, more optimal annealing schemes such as that using a culling fraction have been proposed [41], but we restrict the present study to the classical one for simplicity of implementation, as the main feature of DPA resides in the introduction of the nonlocal moves rather than the annealing scheme itself. To further compare both algorithms, we also implement the entropy-preserving steps in PA in the final section. For the DPA simulations, after an exploration of the parameter space we found $(\beta_0, \kappa_0) = (2, 0)$ to be, generally, a good starting point for the entropy-preserving adaptive steps in the $L = 4$ and 6 cases. For $L = 8$ we used $(\beta_0, \kappa_0) = (2, 0)$ as well, even though due to long simulation times the exploration of the parameter space was not that extensive and thus this initial point is more prompt to be optimized. Our method is therefore able to thermalize the system at finite temperatures as long as $\kappa_0 = 0$, which shows it has a better plasticity over PA.

Nevertheless, since in DPA we perform one spin flip and one wall flip per iteration, we distinguish between two different measures of computational work. In the first scenario, assuming that random number generation is the main bottleneck in MC algorithms, a strictly fair comparison would impose that $N_{sweeps}^{(PA)} = 2N_{sweeps}^{(DPA)}$, even if this evidently yields a much poorer thermalization for the former. On the other hand, if a fast enough random number generator is assumed, one can directly count the computational work as the total number of MCMC updates attempted along the simulation. As we have discussed previously, each spin flip consists of four wall flips, and conversely, each wall flip can be counted as one-fourth of a spin flip. This means that, in this case, to use
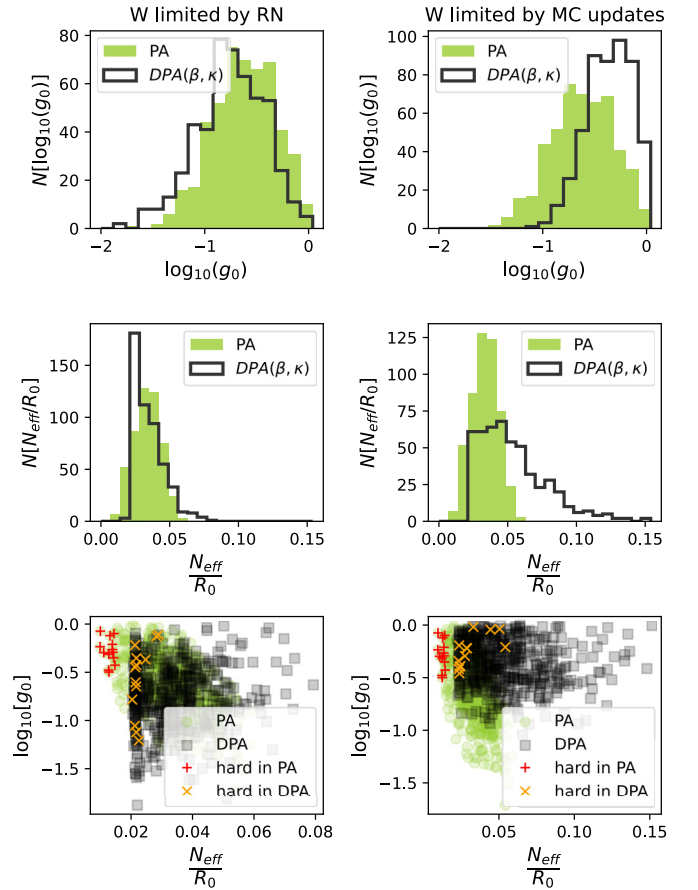

FIG. 8. DPA outperforms PA for $L = 4$ lattices when MCMC process simulations are the computational bottleneck. Histograms of g_0 and N_{eff}/R obtained by solving, by both PA and DPA, a total of $N_{dis} = 500$ different disorders of $L = 4$ RFW lattices, with bonds randomly distributed according to a normal distribution. Simulations using the same amount of random numbers (left panels) and of MCMC updates (right panels). Crosses in the bottom panels mark these instances PA finds the hardest (red), and how these same instances score in DPA (orange).

the same W one has to impose $N_{sweeps}^{(PA)} = \frac{5}{4} N_{sweeps}^{(DPA)}$ (recall that each thermalization step in DPA consists of one spin flip and one wall flip, which equate to five walls being updated). We address and compare both scenarios using $N_{sweeps}^{(PA)} = 10$ and $N_{sweeps}^{(DPA)} = 5$ for the first and $N_{sweeps}^{(PA)} = 10$ and $N_{sweeps}^{(DPA)} = 8$ for the second.

1. $L = 4$

We first apply our method to the study of $L = 4$ RFW lattices and take a look at the histograms of the parameters g_0 and N_{eff} obtained for 500 different disorders, comparing the solutions obtained for the same disorders with a regular PA algorithm. We plot our results in Fig. 8, the left panels corresponding to the first scenario discussed (W limited by the amount of random numbers) and the right panels corresponding to the second (W limited by the number of MC updates). The top graphs show the histograms for g_0, the middle graphs the histograms for $\frac{N_{eff}}{R_0}$, and the bottom graphs the scatter plot relating both parameters for each of the solved

disorders. In these bottom panels, we also mark the disorders that regular PA finds to be the hardest to solve, identified as those obtaining the lowest values of family entropy (red crosses) and how these same disorders score when solved by DPA (black crosses). In this case, we consider a disorder to be hard whenever PA is not able to reach a final effective number of surviving families $\frac{N_{\text{eff}}}{R_0} = 0.015$ or, equivalently, about 2% of all instances of disorder.

In the first scenario (Fig. 8, left panels), we observe a certain tradeoff between parameters (Fig. 8, top left and center left panels), resulting in an equivalent performance of the two methods. The fact that the adaptive steps procedure over the two control parameter space (β, κ) is able to properly drive the evolution of the system and set an objective cutoff family entropy is nevertheless noticeable (Fig. 8, center left panel). Also, the fact that the family entropy is no worse than for PA is relevant, taking into account that, in this scenario, the total amount of spin updates is lower and thus one might expect a worse thermalization. With the cutoff imposed on family entropy, scores on this parameter are increased even for hard cases, but they seem to lay, generally, in the lower range for DPA as well (Fig. 8, bottom left panel).

On the other hand, in the second scenario (Fig. 8, right panels) we see a substantial improvement with the DPA method, as the histograms for both measured parameters seem to be shifted towards bigger values and thus imply better metrics than those obtained with standard PA (Fig. 8, top right and center right panels). As can be seen in the bottom right panel, even some of the hard cases' metrics are improved, as they manage to escape from the lower range of family entropy without diminishing its g_0 score.

In Fig. 9 we take a closer look at the results obtained by the two methods by comparing the measured parameters obtained by both for each of the solved disorder realizations. A brighter color indicates a higher density of points, and the red straight line is plotted to indicate the region in which both methods yield the same results for a given parameter. We also marked those disorder instances classically labeled as hard for PA with red crosses. Again, the left and right panels correspond to the first and second scenarios, respectively.

Most importantly, as can be seen in the top panels of Fig. 9, both algorithms find the same ground-state energy for all disorders. The distribution of g_0 and N_{eff} is shown in the center and bottom panels, respectively. In them we can see that for the same amount of RN consumed (first scenario, left panels), the obtained g_0 is slightly higher for PA but certainly comparable to DPA for the vast majority of disorders (Fig. 9, middle left panel), even for most of the hardest instances. On the other hand, the family entropy is better with our method (Fig. 9, bottom left panel), which implies that a better thermalization of the replicas is achieved (again, recall that the number of total spin updates is lower, and thus this could have been expected to be lower as well). In the second scenario, DPA obtains clearly better results than PA for both parameters, g_0 (Fig. 9, middle right panel) and N_{eff} (Fig. 9, bottom right panel).

Lastly, we focus on cases of disorder realizations that are hard for the PA algorithm, that is, those cases resulting in low family entropy. From Eq. (1) it follows that a possible way to overcome the difficulty of equilibration and improve the final
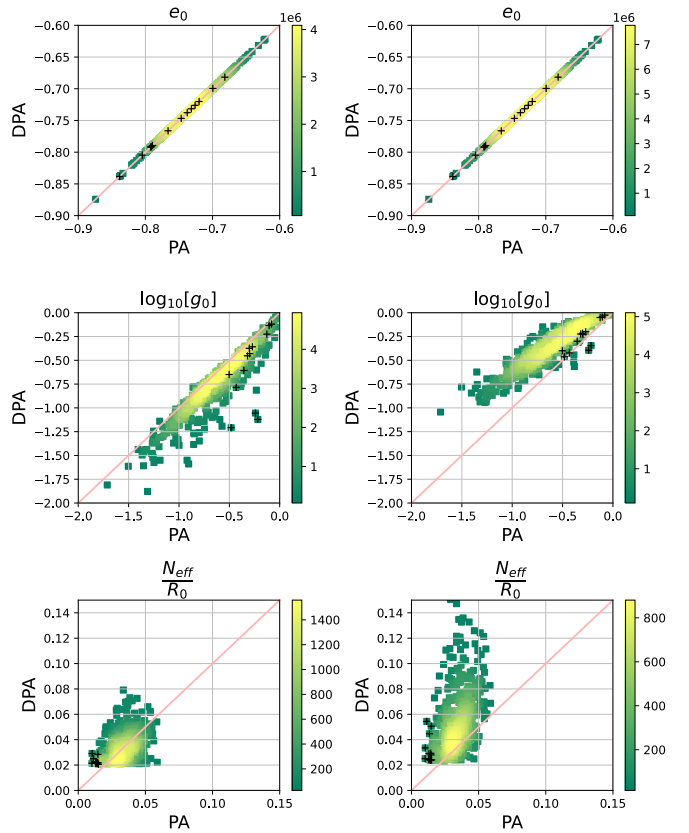

FIG. 9. DPA provides on average higher family entropies than PA when solving $L = 4$ lattices. Comparison of results obtained by PA and DPA for $L = 4$ RFW lattices for the three measured parameters. From top to bottom: ground-state energy, g_0, and effective number of surviving families. Simulations using the same amount of random numbers (left) and of MCMC updates (right). Black crosses mark these instances that PA finds the hardest.

family entropy obtained by PA is to reduce the differences $\Delta\beta = \beta' - \beta$ by increasing the number of temperature steps. Following this idea, to further contrast both methods, we now take a collection of disorders that PA finds difficult to equilibrate and for which a very similar g_0 is obtained for PA and DPA, so that they lay close to the $g_0^{\text{PA}} = g_0^{\text{DPA}}$ red line in the center panels of Fig. 9, and we repeatedly solve them several times with an increasing number of temperature steps in the PA annealing schedule (i.e., more adiabatically). For the cases at hand, we find that for PA to obtain a family entropy similar to that obtained by DPA, we would have to invest approximately between 4 and 16 times as much computational power depending on the disorder instance, Fig. 10. Furthermore, one should also note here that, contrary to the addition of more replicas, that extra computational effort would not be parallelizable in PA, since the annealing schedule must always be followed sequentially.

2. L = 6

We now apply the same study to $L = 6$ RFW lattices. As for the $L = 4$ case, we solve several different Gaussian disorders with both PA and DPA considering the two scenarios discussed above, and we plot the obtained histograms in
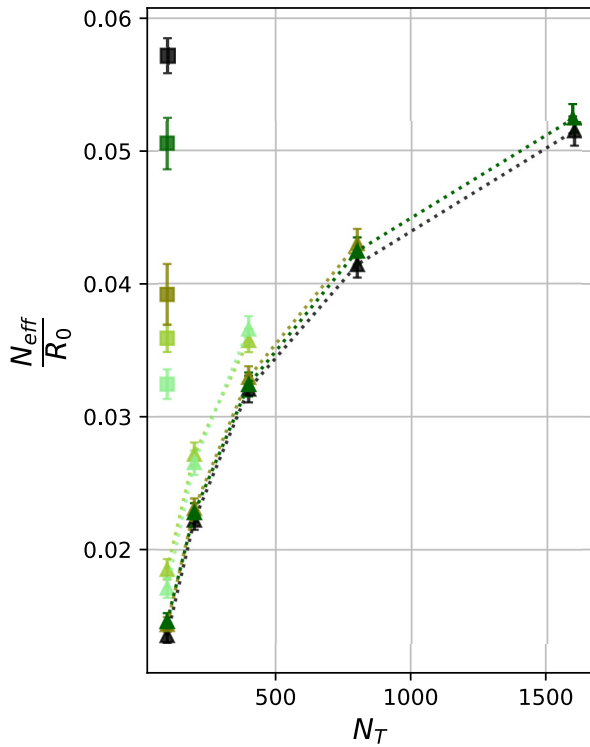
FIG. 10. Hard cases on PA require more adiabatic annealing processes to reach an equivalent family entropy to DPA. Study of the amount of computational power that would have to be invested to solve $L = 4$ RFW lattices with PA, in order to obtain a similar final family entropy as with DPA, for hard instances for which $g_0^{\text{PA}} \approx g_0^{\text{DPA}}$. Error bars are standard deviations over $N_{\text{rep}} = 100$ repetitions of the same process at each value of N_T. Square markers are obtained with DPA and triangle ones with PA using an increasing number of temperature steps. Each color indicates a different disorder.

FIG. 11. DPA enables thermalization of $L = 6$ lattices, even when these are hard for the PA algorithm. Histograms of g_0 and N_{eff}/R obtained by solving, by both PA and DPA, a total of $N_{\text{dis}} = 1000$ different disorders of $L = 6$ RFW lattices, with bonds randomly distributed according to a normal distribution. Simulations using the same amount of random numbers (left) and of MCMC updates (right). Crosses in the bottom panels mark these instances that PA finds the hardest (red), and how these same instances score in DPA (orange).

Fig. 11. For the first scenario (Fig. 11, left panels), we again see a certain tradeoff between g_0 and N_{eff}. Nevertheless, the gain in family entropy is now more noticeable than for smaller lattices, as now PA does encounter some hard instances for which it obtains a single surviving family, preventing adequate thermalization. On the contrary, the adaptive steps in DPA are capable of driving the population toward nonzero values, thus ensuring proper thermalization (Fig. 11, center left panel). When considering the computational work limited by the number of MC updates, the g_0 obtained with DPA is comparable to that obtained with PA, while the gain in family entropy remains the same. Again, those hard instances of PA are properly thermalized under DPA. On top of that, comparing the results obtained for each individual disorder between the two scenarios (Fig. 11, bottom panels), we see that, while for PA the hard instances are more or less evenly distributed along the range of g_0 values, they are shifted towards bigger values, especially for the second scenario.

Looking at the top panels of Fig. 12, we again confirm that both methods find the same ground-state energy for all disorder realizations. Speaking of g_0, the hardest instances are more or less uniformly distributed around their mean in both scenarios (Fig. 12, middle panels), while they are shifted towards PA when restricted by the generation of random

numbers, but centered between the two methods when restricted by the amount of MC updates.

We finally study in Fig. 13, as in the case $L = 4$, a collection of instances that PA finds hard, and we obtain a similar value of g_0 with both methods. Again, we use an increasingly adiabatic process in PA to see how much more computational power would be necessary to get results comparable to those obtained with DPA. In this case, DPA achieves an equivalent performance to PA with about between 2 and 5 more computational investment (in terms of W, Eq. (24)), depending on the instance. We note a more linear behavior of the family entropy with N_T than in Fig. 13, probably because the considered values of N_T are not big enough. This indicates that $L = 6$ is at the limit of our numerical capabilities.

3. $L = 8$

To study hard EA disorders on big lattice sizes for which a proper thermalization is not guaranteed, one classically relies on running many times the same instance independently.
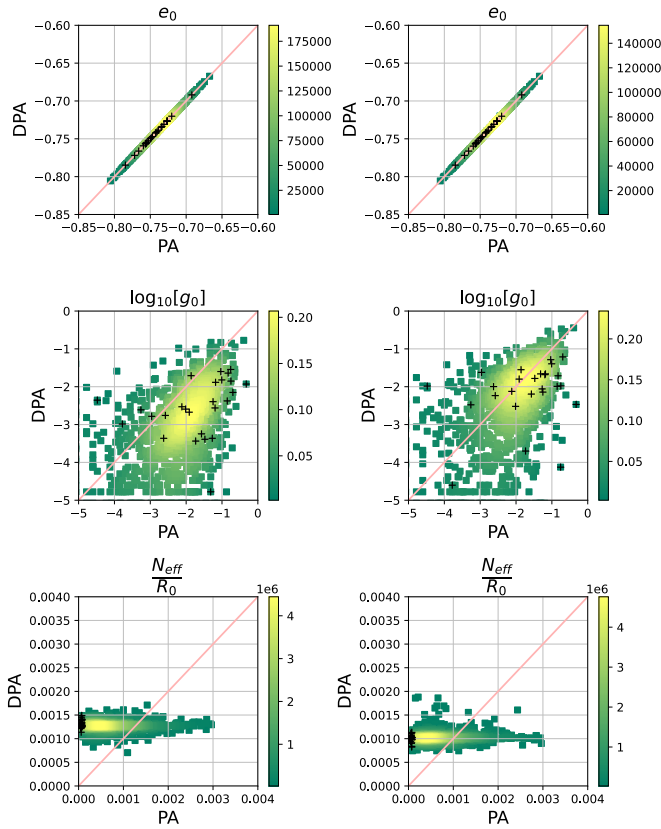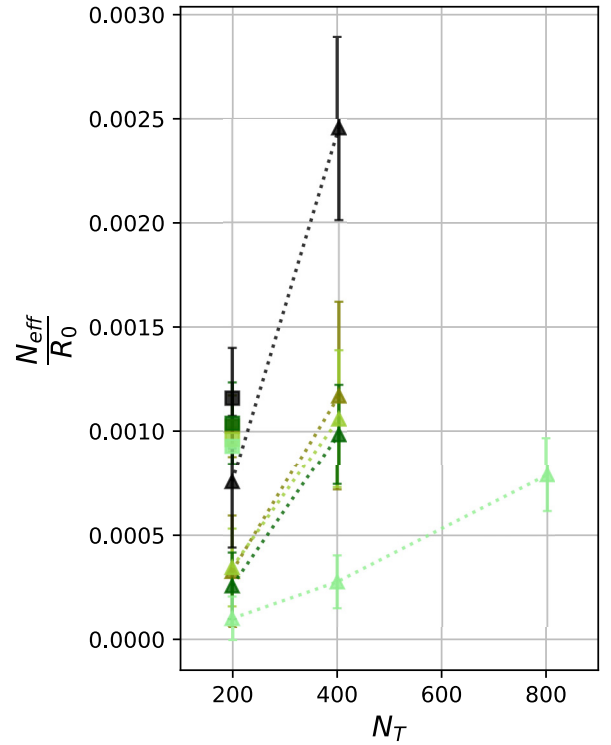
FIG. 12. DPA provides on average higher family entropies than PA when solving $L = 6$ lattices. Comparison of results obtained by PA and DPA for $L = 6$ RFW lattices for the three measured parameters. From top to bottom: ground-state energy, g_0, and effective number of surviving families. Simulations using the same amount of numbers (left) and of MCMC updates (right). Black crosses mark these instances that PA finds the hardest.

FIG. 13. Hard cases on PA require more adiabatic annealing processes to reach an equivalent family entropy to DPA. Study of the amount of computational power that would have to be invested to solve $L = 6$ RFW lattices with PA, in order to obtain a similar final family entropy as with DPA, for hard instances for which $g_0^{\text{PA}} \approx g_0^{\text{DPA}}$. Error bars are standard deviations over $N_{\text{rep}} = 100$ repetitions of the same process at each value of N_T. Square markers are obtained with DPA and triangle ones with PA using an increasing number of temperature steps. Each color indicates a different disorder.

We apply this methodology to $L = 8$ RFW lattices with Gaussian disorder to test how both algorithms perform. Concretely, we study 20 different disorder instances and run each of them 50 times with each algorithm, considering the computational work limited by the amount of MC updates (second discussed scenario in the previous sections). As the systems are not thermalized, we only focus here on the minimum energies per spin found during the simulations, $e_{\min}$, and not on the previously discussed metrics. Contrary to what should be expected from a thermalized system, none of the algorithms clearly converges to the same energy among different runs for a given disorder. In fact, when studying it with PA, for 11 out of the 20 studied cases the same minimum energy was not found in any of the runs. For 4 of them the minimum energy was found twice, for 2 of them four times, and only for the 3 easiest ones was it repeatedly found six times. In Fig. 14 we show, for one of the three easy studied instances (top panel) and for one of the 11 hard ones (bottom panel), the minimum energies found among independent runs by each algorithm.

On the one hand, we observe that the spread of the minimum energies found among several runs of the same disorder tends to be larger with DPA than with PA, probably due to the larger configurational space caused by the introduction of topological defects. On the other hand, when comparing
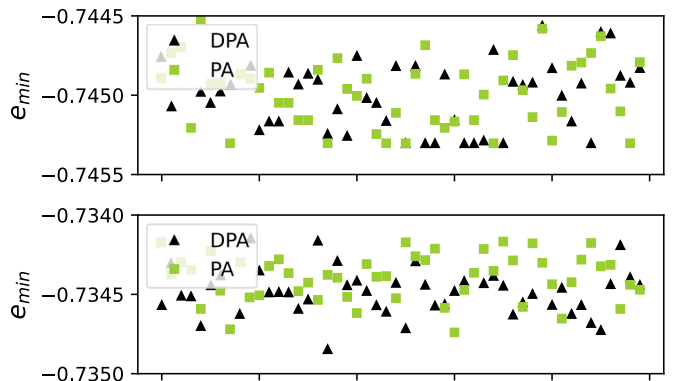

FIG. 14. Both algorithms have a low convergence ratio to the same minimum energy state among independent runs. Upper panel: example of an easy case for which both algorithms find the same solution after running it several times (both PA and DPA show a convergence ratio of 12%). Bottom panel: example of a hard case for which both algorithms completely fail to converge towards any minimum energy state, and for which DPA finds the best solution between the two.
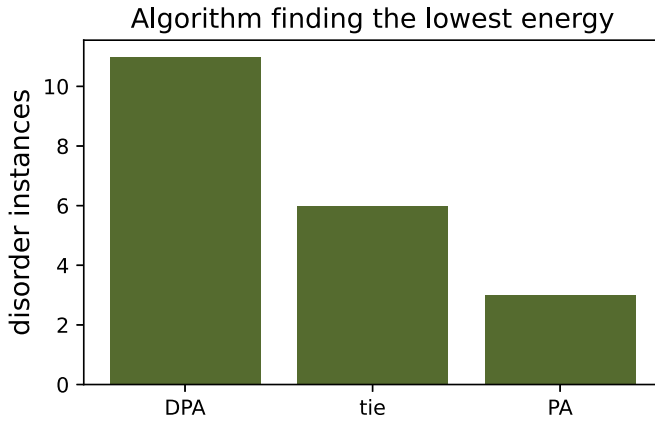
FIG. 15. DPA finds lower-energy states. When studying nonthermalized RFW $L = 8$ lattices, DPA is generally able to find lower energies than PA using the same computational resources (in the same amount of MCMC updates scenario). Results obtained after solving 20 different disorder instances, 50 times each with each algorithm.

the minimum energy found with both algorithms, we observe that DPA generally performs better, yielding lower energies (see Fig. 15, the "tie" cases contain easy disorders). The fact that DPA generally finds states with lower energies when solving hard instances for which thermalization is poor (or, equivalently, for which no convergence to the same solution is achieved) indicates that the proposed nonlocal moves are indeed useful to explore energy landscape in hard disorder instances, effectively allowing the system to escape local minima.

C. Family entropy-preserving steps in PA

To further test the effectiveness of the newly introduced nonlocal moves, we implement the family entropy-preserving adaptive steps in PA. We use the same parameters used in the simulations of previous sections and find the results shown in Fig. 16 for $L = 4$ (left panels) and $L = 6$ (right panels). While the adaptive steps are again able to generally impose a cutoff value on the final family entropy, the obtained g_0 is quite worse than in regular PA. For $L = 6$ lattices the cutoff is actually not achieved for some disorders, some of which are even seen as easy (obtaining a high value of N_{eff}) by PA. This fact suggests that different annealing schedules may make PA find hard or easy different instances. More noticeably, the adaptive steps make PA fail to find the same ground state than when using the standard annealing schedule in some $L = 6$ cases. We therefore conclude that the entropy-preserving annealing steps are not effective by themselves when restricted to a one-dimensional annealing space, and thus the nonlocal moves must be the reason why DPA is able to improve thermalization of hard instances, as seen in previous sections.

IV. CONCLUSIONS AND OUTLOOK

In this paper, we introduced a method, inspired by the toric code, for exploring the rugged energy landscape of 2D random-bond Ising and 3D random plaquette gauge models based on creation, movement, and annihilation of topological
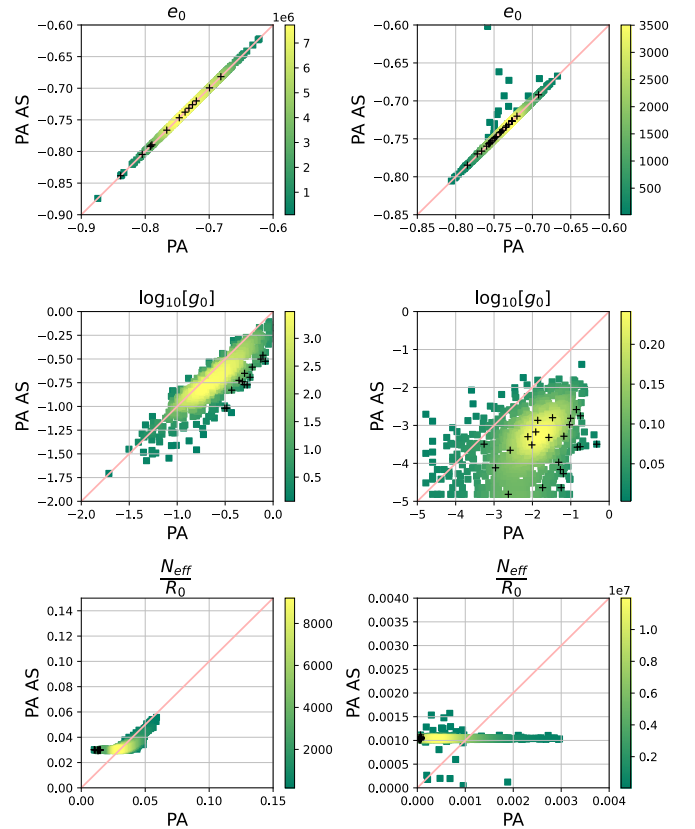

FIG. 16. The entropy-preserving steps are not optimal in standard PA. For $L = 4$ (left panels) and $L = 6$ (right panels) RFW lattices, the adaptive steps procedure is not effective when the parameter space is reduced to only temperature. Black crosses mark the disorders that normal PA finds the hardest to solve.

defects. The advantages of this approach are as follows: (i) from the point of view of the original phase space of the models, the effect of the dynamics of the topological defects is equivalent to nonlocal moves, which allow the system to overcome energy barriers that other exploration strategies based on thermal fluctuation or tunneling do not, and (ii) topological defect moves are compatible with massively parallel implementations. The disadvantage of this approach is a substantial enlargement of the phase space in the early stages of simulation.

We implemented this method in a state-of-the-art population annealing algorithm by adding moves related to topological defects and an extra field parameter in the Hamiltonian, κ, which energetically penalizes them. The presented defect driven population annealing (DPA) algorithm utilizes annealing in a two-parameter space: the temperature and the κ, where starting temperature may now be arbitrary. This offers additional application flexibility, but leaves the annealing path undetermined. To this end, we devised a family entropy-preserving adaptive step procedure that effectively navigates the (β, κ)-space in order to drive the replica population towards an objective value of final family entropy. The merging of more advanced annealing schemes such as culling fraction with the exploration of the two-dimensional

annealing parameter space leaves room for improvement, which we leave as a possible extension of this work.

We have tested the algorithm on the 3D random plaquette gauge model, as the 2D random bond Ising model lacks numerically hard disorder cases. We leave the problem of algorithm performance in the 2D random bond Ising model for future study. The ground-state problem of the 3D RPGM with Gaussian disorder was solved in a representation we named the random field wall model, using standard PA as a benchmark. We found that the RFW model is substantially more numerically demanding than the 3D Edwards-Anderson model, which limited our thorough simulations to lattice sizes $L = 4$ and 6. Bigger lattices have been explored, but we leave its thorough study for a future extension of the present work. The results for DPA on those lattices have shown that it is capable of improving thermalization in comparison with PA, effectively avoiding running out of family entropy even for the hardest disorder instances.

When focusing on the instances that PA finds the hardest, we have been able to observe that DPA can be superior to regular PA with a computational investment 2–16 times higher, approximately, depending on the case. Yet sometimes DPA trades improvement of family entropy for a decrease in g_0 metric. Finally, when studying hard cases for which thermalization is not properly achieved, the proposed nonlocal moves still show an advantage in overcoming energy barriers and thus yield lower energies. It is also worth noting that the results are greatly improved when a fast enough random number generator is used, such that the Monte Carlo updates themselves constitute the real bottleneck for the simulations and thus the fair measure of the computational work invested.

[1] S. G. Brush, History of the Lenz-Ising model, Rev. Mod. Phys. **39**, 883 (1967).

[2] G. Parisi, Mean field theory of spin glasses: Statics and dynamics, Les Houches **85**, 131 (2007).

[3] F. Barahona, On the computational complexity of Ising spin glass models, J. Phys. A **15**, 3241 (1982).

[4] D. J. Amit, H. Gutfreund, and H. Sompolinsky, Spin-glass models of neural networks, Phys. Rev. A **32**, 1007 (1985).

[5] J. L. van Hemmen, Spin-glass models of a neural network, Phys. Rev. A **34**, 3435 (1986).

[6] A. Lucas, Ising formulations of many NP problems, Front. Phys. **2**, 5 (2014).

[7] A. Lucas, Hard combinatorial problems and minor embeddings on lattice graphs, Quantum Inf. Proc. **18**, 203(2019).

[8] A. Irbäck, L. Knuthson, S. Mohanty, and C. Peterson, Folding lattice proteins with quantum annealing, Phys. Rev. Res. **4**, 043013 (2022).

[9] N. M. D. Oliveira, R. M. D. A. Silva, and W. R. D. Oliveira, QUBO formulation for the contact map overlap problem, Int. J. Quantum Inf. **16**, 1840007 (2018).

[10] F. Hernández, K. Díaz, M. Forets, and R. Sotelo, *Application of Quantum Optimization Techniques (QUBO Method) to Cargo Logistics on Ships and Airplanes* (IEEE, 2020), pp. 1–1.

[11] F. Phillipson and I. Chiscop, *Multimodal Container Planning: A QUBO Formulation and Implementation on a Quantum Annealer* (Springer International, Cham, 2021).

[12] J. F. A. Sales and R. A. P. Araos, Adiabatic quantum computing for logistic transport optimization, arXiv:2301.07691.

[13] S. Kirkpatrick, C. D. Gelatt, Jr., and M. P. Vecchi, Optimization by simulated annealing, Science **220**, 671 (1983).

[14] J. Gubernatis, *The Monte Carlo Method in the Physical Sciences: Celebrating the 50th Anniversary of the Metropolis Algorithm* (American Institute of Physics, New York, 2003).

[15] J. Machta, Population annealing with weighted averages: A Monte Carlo method for rough free-energy landscapes, Phys. Rev. E **82**, 026704 (2010).

[16] W. Wang, J. Machta, and H. G. Katzgraber, Comparing Monte Carlo methods for finding ground states of Ising spin glasses: Population annealing, simulated annealing, and parallel tempering, Phys. Rev. E **92**, 013303 (2015).

[17] W. Wang, J. Machta, and H. G. Katzgraber, Population annealing: Theory and application in spin glasses, Phys. Rev. E **92**, 063307 (2015).

[18] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, Equations of state calculations by fast computing machines, J. Chem. Phys. **21**, 1087 (1953).

[19] M. E. J. Newman and G. T. Barkema, *Monte Carlo Methods in Statistical Physics* (Clarendon Press, 1999).

[20] S. F. Edwards and P. W. Anderson, Theory of spin glasses, J. Phys. F **5**, 965 (1975).

[21] R. H. Swendsen and J.-S. Wang, Nonuniversal critical dynamics in Monte Carlo simulations, Phys. Rev. Lett. **58**, 86 (1987).

[22] U. Wolff, Collective Monte Carlo updating for spin systems, Phys. Rev. Lett. **62**, 361 (1989).

[23] J. Houdayer, A cluster Monte Carlo algorithm for 2-dimensional spin glasses, Eur. Phys. J. B **22**, 479 (2001).

[24] Z. Zhu, A. J. Ochoa, and H. G. Katzgraber, Efficient cluster algorithm for spin glasses in any space dimension, Phys. Rev. Lett. **115**, 077201 (2015).

[25] A. B. Finnila, M. A. Gomez, C. Sebenik, C. Stenson, and J. D. Doll, Quantum annealing: A new method for minimizing multidimensional functions, Chem. Phys. Lett. **219**, 343 (1994).

[26] S. Morita and H. Nishimori, Mathematical foundation of quantum annealing, J. Math. Phys. **49**, 125210 (2008).

[27] E. Crosson and A. W. Harrow, Simulated quantum annealing can be exponentially faster than classical simulated annealing, in *Proceedings of the 2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)* (IEEE, Piscataway, NJ, 2016), pp. 714–723.

[28] A. Y. Kitaev, Fault-tolerant quantum computation by anyons, Ann. Phys. **303**, 2 (2003).

[29] E. Dennis, A. Kitaev, A. Landahl, and J. Preskill, Topological quantum memory, J. Math. Phys. **43**, 4452 (2002).

[30] M. Palassini and A. P. Young, Triviality of the ground state structure in Ising spin glasses, Phys. Rev. Lett. **83**, 5126 (1999).

[31] P. L. Ebert, D. Gessert, and M. Weigel, Weighted averages in population annealing: Analysis and general framework, Phys. Rev. E **106**, 045303 (2022).

[32] A. Barzegar, C. Pattison, W. Wang, and H. G. Katzgraber, Optimization of population annealing Monte Carlo for large-scale spin-glass simulations, Phys. Rev. E **98**, 053308 (2018).

[33] H. A. Kramers and G. H. Wannier, Statistics of the two-dimensional ferromagnet. Part I, Phys. Rev. **60**, 252 (1941).

[34] F. J. Wegner, Duality in generalized Ising models and phase transitions without local order parameters, J. Math. Phys. **12**, 2259 (1971).

[35] C. Wang, J. Harrington, and J. Preskill, Confinement-Higgs transition in a disordered gauge theory and the accuracy threshold for quantum memory, Ann. Phys. **303**, 31 (2003).

[36] T. Ohno, G. Arakawa, I. Ichinose, and T. Matsui, Phase structure of the random-plaquette Z_2 gauge model: Accuracy threshold for a toric quantum memory, Nucl. Phys. B **697**, 462 (2004).

[37] K. Takeda and H. Nishimori, Self-dual random-plaquette gauge model and the quantum toric code, Nucl. Phys. B **686**, 377 (2004).

[38] G. Arakawa and I. Ichinose, Z_N gauge theories on a lattice and quantum memory, Ann. Phys. **311**, 152 (2004).

[39] A. B. Bortz, M. H. Kalos, and J. L. Lebowitz, A new algorithm for Monte Carlo simulation of Ising spin systems, J. Comput. Phys. **17**, 10 (1975).

[40] M. Baity-Jesi, R. A. Baños, A. Cruz, L. A. Fernandez, J. M. Gil-Narvion, A. Gordillo-Guerrero, D. Iñiguez, A. Maiorano, F. Mantovani, E. Marinari, V. Martin-Mayor, J. Monforte-Garcia, A. M. Sudupe, D. Navarro, G. Parisi, S. Perez-Gaviro, M. Pivanti, F. Ricci-Tersenghi, J. J. Ruiz-Lorenzo, S. F. Schifano *et al.*, Critical parameters of the three-dimensional Ising spin glass, Phys. Rev. B **88**, 224416 (2013).

[41] C. Amey and J. Machta, Analysis and optimization of population annealing, Phys. Rev. E **97**, 033301 (2018).