Thermal Memory in Glassy Systems

Nicholas M. Boffi and Po-Yi Ho

Recent experiments have demonstrated that disordered, glassy systems can exhibit relaxation dynamics that depend on the system's thermal excitation history: that is, they retain a *thermal memory*. We investigate thermal memory by studying two disordered systems. We first study a model of a random walk in a random environment. We review a real space renormalization group method to obtain exact solutions for the effective dynamics of the system. In particular, we obtain the diffusion front and the two-time diffusion front. The RG technique together with simple numerical simulations is applied to investigate two-temperature effective dynamics. Subsequently, we study a two dimensional generalization of this model: diffusion in the square lattice Ising spin glass. Diffusion in this system is simulated using kinetic Monte Carlo techniques and the Python code used to do so is provided. Two simulation models were considered: one in which the distribution of the bond energies is taken to have an explicit dependence on the maximum temperature the glass has ever been exposed to, and one in which the distribution of bond energies is held fixed, but the system is allowed to relax at a higher temperature before the diffusive dynamics are studied. Slower relaxations are found for higher values of $T_{\max}$ in both cases.

1. INTRODUCTION

Glassy systems often exhibit slow, "aging" dynamics that depend on the system's excitation history, such as the waiting time of a perturbation. A recent experiment, summarized in Fig. 1, demonstrates that such systems can also exhibit thermal memory, in which the relaxation dynamics depend on the maximum temperature to which the system was exposed. The essence of the experiment relies on a new cryogenic fabrication method, which enables the creation of an electron glass at extremely low temperatures. This allows the experimentalists to control the highest temperature to which the glass has ever been exposed. Notably, the rate of the logarithmic relaxation depends on the ratio of the current and maximum temperatures $T/T_{\max}$ of the system [1].

The hypothesized explanation from Ref. [1] for this dependence is that the glass, when fabricated at cryogenic temperatures, is "born" in a highly energetically unfavorable state. Unlike glassy systems created through a quenching procedure, the glasses in this experiment do not start in a high temperature state above the glass transition, and hence the level of equilibration of the final state has no dependence on an experimental quench rate. By tuning $T_{\max}$, one is able to directly control the amount of equilibration the system has *ever* been able to undergo, and hence control the depth of the local minimum in the energy landscape defining the current state of the glass. Higher values of $T_{\max}$ lead to more equilibration, a more stable state, and slower relaxation dynamics.

Here, we study thermal memory in disordered systems in an attempt to further understand this result outside of a qualitative explanation. We first investigate an analytically tractable model of a particle diffusing on a random landscape. The many minima of the random landscape span all scales, and the diffusive behavior of the particle naturally exhibits slow dynamics. We review a real space renormalization group method to obtain exactly the diffusion and two-time diffusion fronts. This model offers a rare example of a disordered system with analytically solvable nonequilibrium dynamics. Simple numerical simulations of the effective dynamics confirm that the particle exhibits both aging and thermal memory.

We then study a more realistic two-dimensional model of diffusion on the square lattice Ising spin glass via kinetic Monte Carlo methods. We consider two versions of this model. The first has an explicit $T_{\max}$ dependence in the distribution of bond energies, and diffusion is studied at fixed current temperature T for several values of $T_{\max}$. The second is an "atomistic" approach in which the distribution of bond energies has no $T_{\max}$ dependence, but systems are allowed to equilibrate for a short time at $T_{\max} > T$ before the diffusion at T is simulated. Data is only recorded from the diffusion simulation at temperature T. The first of these models essentially tests the hypothesis that higher $T_{\max}$ leads to deeper minima and hence slower relaxation, while the second simulates the experiment directly to see if we can capture the emergence of a $T_{\max}$ dependence from the dynamics. We find slower diffusion for higher values of $T_{\max}$ in both cases. The second approach is particularly interesting, as the thermal memory occurs organically from the simulation, directly in line with the recent experiment [1].

I. RSRG OF RWRE

We study a particle in a disordered 1D landscape. The particle performs Arrhenius diffusion across randomly drawn energy barriers. Denote the potential of site i by U_i. Without loss of generality, the barrier heights f_i can be made alternating, such that $f_i = U_i - U_{i+1} = (-1)^i F_i$, where $F_i = |U_i - U_{i+1}|$. Sites i and $i+1$ are connected by bonds of random length l_i. The bond variables are chosen independently from bond to bond. The model is therefore specified by the distribution of barrier heights and lengths $P(F,l)$. We refer to this model as random walks in random environments (RWRE, also known as the Sinai model), see Fig. 2. We focus on the symmetric case where the ascending and descending barriers have the same statistics, but the techniques discussed can be applied to asymmetric cases and finite landscapes.

We review a real space renormalization group (RSRG) analysis of RWRE following Ref. [2]. All results reproduced below can be found in Ref. [2], but we more clearly comment on various parts of the derivation. The RSRG technique allows us to extract effective dynamics that are exact at long times. Deviations from the effective dynamics can be accounted for by rare events in the RG procedure. Below, we review the RSRG analysis and obtain the diffusion front and the two-time diffusion front. The analytical results are compared to stochastic simulations. We then extend the analysis to investigate the two-temperature diffusion front, and compare the observed aging and thermal memory phenomena to the experiment discussed in the introduction.

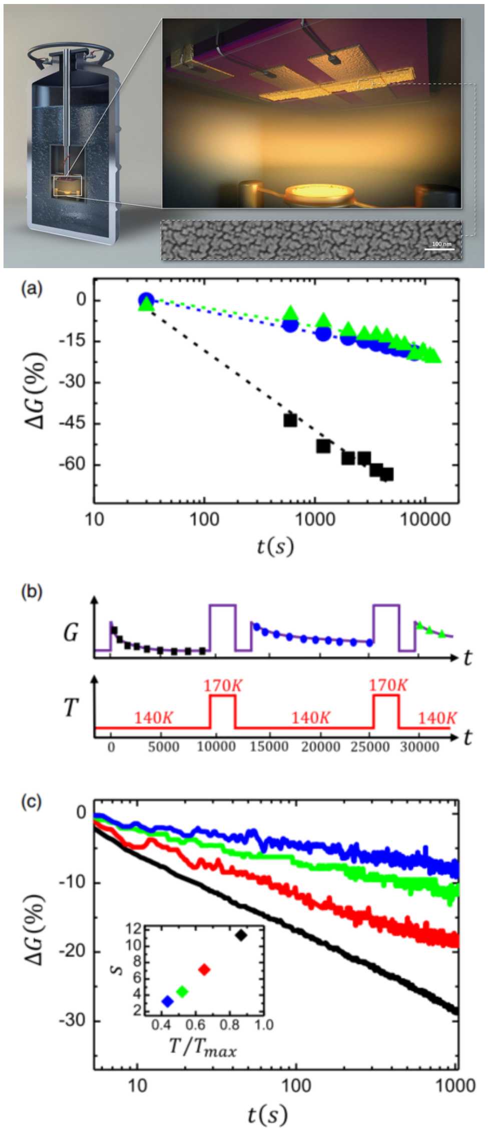

FIG. 1. (Top) A quench-condensed Au film on Si/SiO substrate forms a disordered electronic system. The quenched, disordered geometry and long-range Coulomb interactions lead to frustrated, glassy dynamics. (Bottom) (a) Change in conductance as a function of time for one sample following the heating protocol in (b). (b) The sample was exposed to a maximum temperature $T_{\max} = 170K$. (c) The slope of the logarithmic relaxation depends on $T/T_{\max}$. From Ref. [1].

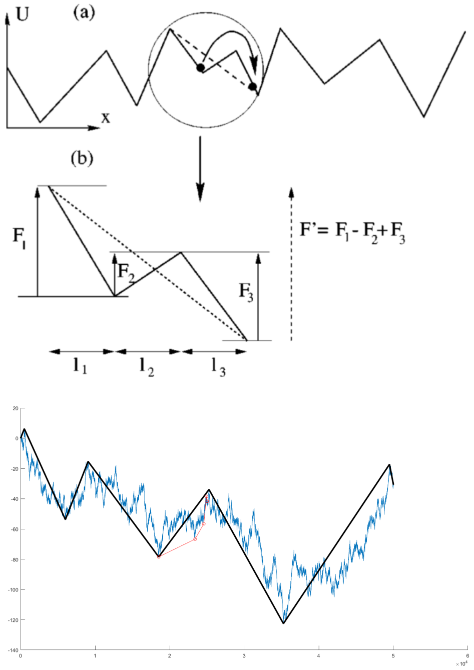

FIG. 2. (a) Effective dynamics of a particle in a random landscape. If the particle is adjacent to a renormalized bond, the particle moves to the bottom of the renormalized valley. (b) The decimation of the smallest barrier, illustrated to be F_2. From Ref. [2]. (Bottom) A numerical demonstration of the RSRG procedure described in the text. Blue is the original landscape. Black is the renormalized landscape. Red traces the effective dynamics of a particle that begins near the middle of the landscape but arrives at the second-from-left renormalized valley.

The RSRG procedure for RWRE is conceptually simple, but offers a rare example of a disordered system whose long-time dynamics are exactly solvable. At long times, the diffusive behavior of the particle is dominated by large barriers. Hence, the RG procedure decimates the bonds with the smallest barriers $\Gamma = F_{\min}$ in a given landscape. Suppose $F_i = \Gamma$ is such a bond. At time scales much longer than the Arrhenius time scale $t_0 \exp(F_i/T)$, the rate for the particle to hop between sites $i-1$ and $i+2$ is approximately the same as it would be if sites i and $i+1$ were decimated, and sites $i-1$ and $i+2$ were connected by a bond with barrier F' and length l',

$$F' = F_{i-1} - F_i + F_{i+1} \tag{1}$$

$$l' = l_{i-1} + l_i + l_{i+1} \tag{2}$$

see Fig. 2. Since this procedure preserves the model structure of alternating, independent bonds, this is our coarse-graining step.

The RG flows are as follows. Since bonds remain independent, it is useful to define renormalized bonds at scale Γ as $\xi_i = F_i - \Gamma$ and the associated distribution as $P_\Gamma(\xi = F - \Gamma, l)$. Following a decimation at scale Γ, the distribution $P_\Gamma(\xi, l)$ is renormalized as

$$\partial_\Gamma P_\Gamma(\xi,l) = \partial_\xi P_\Gamma(\xi,l) + P_\Gamma(0,\cdot) *_l P_\Gamma(\cdot,\cdot) *_{\xi,l} P_\Gamma(\cdot,\cdot) \\ -2P_\Gamma(\xi,l)\int_0^\infty dl' P_\Gamma(0,l') + 2P_\Gamma(\xi,l)\int_0^\infty dl' P_\Gamma(0,l'), \tag{3}$$

where $*$ denotes convolution with respect to the subscript and dotted variables, i.e. $P_\Gamma(\cdot,\cdot) *_{\xi,l} P_\Gamma(\cdot,\cdot) = \int_0^\infty dl' \int_0^\infty d\xi' P_\Gamma(\xi - \xi', l - l') P_\Gamma(\xi', l')$. The first term on the RHS is due to the change in the definition of ξ as Γ increases. The second term accounts for the new bonds created by decimation. A new bond is formed at level ξ' if a decimated bond (at level $\xi = 0$) has neighboring bonds that add to ξ', and similarly for its length. Since the bonds are independent, the probability of the renormalized bond can be obtained as a convolution of the probability of the current bonds. The third term accounts for the decimation of the two bonds adjacent to the smallest barriers. The last term rescales the distribution to keep it normalized by accounting for the overall net loss of a fraction of bonds. This is the logic that we will use to write down other RG flows below. This net loss also gives the evolution of the total number n_Γ of bonds as

$$\partial_\Gamma n_\Gamma = -n_\Gamma 2\int_0^\infty dl' P_\Gamma(0,l'). \tag{4}$$

The average bond length $\bar{l}_\Gamma = \int d\xi dl l P_\Gamma$ similarly evolves as

$$\partial_\Gamma \bar{l}_\Gamma = \bar{l}_\Gamma 2\int_0^\infty dl' P_\Gamma(0,l'). \tag{5}$$

In the symmetric case considered here, the RG flows simplify to

$$\left(\partial_\Gamma - \partial_\xi\right) P_\Gamma(\xi,l) = P_\Gamma(0,\cdot) *_l P_\Gamma(\cdot,\cdot) *_{\xi,l} P_\Gamma(\cdot,\cdot). \tag{6}$$

The RSRG procedure allows us to extract an effective dynamics that approximate the Arrhenius dynamics at long times. In the effective dynamics, the particle at time t is at the bottom of the renormalized valley at scale $\Gamma = T\ln(t/t_0)$ which contains the initial point, see Fig. 2. We set the microscopic timescale t_0 to unity. The effective dynamics become the exact dynamics as $\Gamma = T\ln t$ approaches ∞, which was shown rigorously in Ref. [3]. This can be seen by considering the broadening of P_Γ under RSRG. At large Γ, the renormalized landscape consists entirely of deep valleys with high barriers. Therefore, the particle with high probability will be at the bottom of the valley which contains the initial point. Corrections to the effective dynamics come from rare events when two adjacent bonds have barriers within order T of each other.

The RG flow given by Eq. 6 can be solved exactly as follows. First, we nondimensionalize the barrier and length variables. Length variables can be normalized by $2\sigma = \int dF F^2 P(F)$ where $P(F)$ is the initial unrenormalized distribution. The dimensionless length variable can then be written as $\lambda = l/\Gamma^2$. The dimensionless barrier variable is $\eta = \xi/\Gamma$. The rescaled probability distribution is $P_\Gamma(\eta,\lambda) = \Gamma^3 P_\Gamma(\xi,l)$. Convolutions become multiplications when Laplace transformed. Hence, Laplace transforming Eq. 6 from $\lambda \to p$,

$$\left[\Gamma\partial_\Gamma - (1+\eta)\partial_\eta + 2p\partial_p - 1\right] P_\Gamma(\eta,p) = \\ P_\Gamma(0,p) P_\Gamma(\cdot,p) *_\eta P_\Gamma(\cdot,p). \tag{7}$$

We look for solutions of the form $\tilde{P}(\eta,p) = a(p)\exp(-\eta b(p))$. At $p = 0$, $\tilde{P}(\eta, p = 0) = \int_0^\infty d\lambda' P(\eta,\lambda') = P(\eta)$ must be normalized so that $a(0) = b(0) = P_\Gamma(0)$. Substituting this requirement into the RG flow, we find that for each p, the following nonlinear ordinary differential equations must be satisfied $\partial b/\partial\Gamma = -a^2$, and $\partial a/\partial\Gamma = -ba$. These can be integrated explicitly to show that Eq. 7 has the fixed point

$$\tilde{P}(\eta,p) = \left(\frac{\sqrt{p}}{\sinh\left(\sqrt{p}\right)}\right)\exp\left(-\eta\sqrt{p}\coth\left(\sqrt{p}\right)\right). \tag{8}$$

At $p = 0$, we find that $P(\eta)$ has the form

$$P_\Gamma(F) = \theta(F - \Gamma)\frac{\exp\left(-(F-\Gamma)/\Gamma\right)}{\Gamma}, \tag{9}$$

where θ is the step function. This is a natural result, since the decimation procedure will lead to long regions of ascending or descending bonds. In other words, the coarse-grained probability distribution of barriers is exponential with width $\langle F\rangle = \Gamma = T\ln t$. What is the convergence towards this fixed point $P^*(\eta) = \exp(-\eta)$? We define $P = P^* + q$ for small q. The linearized RG flows are then

$$\left[\Gamma\partial_\Gamma - (\eta+1) - 1\right] q = 2qP^* + q(0)\eta e^{-\eta}. \tag{10}$$

The analysis of the linearized RG flow is too long to be included for our purposes here. Unlike usual RG approaches, the RSRG method applied to RWRE is useful not because it characterizes phase transitions, but because it characterizes effective dynamics that are otherwise difficult to obtain. Nevertheless, it has been shown that perturbations converge as Γ^{-1} towards the fixed point with eigenvector $(\eta - 1)\exp(-\eta)$, corresponding to a shift in Γ. The explicit form of the bond length distribution can be found by integrating Eq. 8.

More importantly, Eq. 4 implies that the number of bonds n_Γ decays asymptotically as Γ^{-2}. Since the average bond length $\bar{l}_\Gamma \sim 1/n_\Gamma$ scales inversely with the number of bonds, $\bar{l}_\Gamma \sim \Gamma^2$. In other words, the position of the particle following the effective dynamics is

$$x \sim T^2 \ln^2 t. \tag{11}$$

This reproduces the classical result obtained by Ref. [3], which introduced the RWRE model. Already, we see that the RWRE model reproduces the slow dynamics aspect of our experimental motivation.

The full expression for the single time diffusion front is obtained as follows. We wish to calculate the probability $P(x,t|0,0)$ that a particle starting at $x=0$ at time $t=0$ is at x at time t under the effective dynamics in our RSRG procedure. Since the particle with high probability will be at the bottom of the renormalized valley containing $x=0$ at $t=0$, the distribution is δ function shaped. To obtain the diffusion front, we average over the quenched disorder (i.e. the various random landscapes) or equivalently the initial conditions. The probability that $x=0$ belongs to a renormalized bond of length l at scale Γ is the length of a given bond divided by the average length of a bond times the probability that a given bond has length l, or $lP_\Gamma(l)/\int dllP_\Gamma(l)$, where $P_\Gamma(l)=\int d\eta P(\eta,l)$. Furthermore, the distance $|x|$ between the starting point and the bottom of the bond is uniformly distributed over $[0,l]$. Hence, averaging over l, we find that

$$\overline{P(x,t|0,0)} = \frac{\int_{|x|}^{\infty} dlP_\Gamma(l)}{2\int dllP_\Gamma(l)}. \tag{12}$$

The substitution of the fixed point can be carried out explicitly. We quote the result from Ref. [2], which is

$$\overline{P(x,t|0,0)} = \frac{1}{T^2\ln^2 t} q\left(\frac{x}{T^2\ln^2 t}\right), \tag{13}$$

where

$$q(X) = \frac{4}{\pi}\sum \frac{(-1)^n}{2n+1}\exp\left(-\pi^2(2n+1)^2|X|/4\right). \tag{14}$$

This agrees with previous rigorous results, validating the RSRG method [4].

II. AGING IN RWRE

We now calculate the two-time diffusion front. The two-time diffusion front is the probability that a particle starting at $x=0$ at $t=0$ first arrives at x' at $t=t'$, then at x at $t=t$,

$$P\left(x,t,x',t'\right) = \overline{P(x,t,x',t'|0,0)}. \tag{15}$$

This quantity is important in aging protocols that apply a perturbation for a waiting time $t_w \equiv t'$. We anticipate that since in the effective dynamics, the barrier scales with both temperature and time as $\Gamma \sim T\ln t$, this quantity will also be important in thermal memory protocols that apply a thermal excitation $T_{\max}$. We focus on large t and t', for which there are several regimes.

We calculate first the probability $D(t',t)$ that a particle remains at the bottom of a renormalized valley and does not move between t' and t. This will be used later in the full expression for the two-time diffusion front. We write the RG flow of D in a similar manner as the RG flow of $P(\xi,l)$. Denote by $D_{\Gamma,\Gamma'}(\xi)$ the probability that a particle on a bond $F=\Gamma+\xi$ does not move between Γ' and Γ. In other words, the bond has not been decimated and the same-valley neighbor has also not been decimated. Therefore,

$$\begin{aligned}\left(\partial_\Gamma - \partial_\xi\right) D_{\Gamma,\Gamma'}(\xi) = &-2P_\Gamma(0) D_{\Gamma,\Gamma'}(\xi) \\ &+ P_\Gamma(0) P_\Gamma(\cdot) *_\xi D_{\Gamma,\Gamma'}(\cdot).\end{aligned} \tag{16}$$

The first term on the RHS accounts for forbidden decimations, while the second term accounts for allowed decimations. At large Γ, P_Γ has reached the fixed point found above. Defining $\alpha=\Gamma/\Gamma'$, we find after substituting the fixed point and rescaling,

$$\left[\alpha\partial_\alpha - (1+\eta)\partial_\eta + 1\right] D_\alpha(\eta) = \int_0^{\eta_1} e^{-\eta_1} D_\alpha(\eta-\eta_1). \tag{17}$$

The RG flow is accompanied by the initial condition $D_{\alpha=1}(\eta)$, which is the probability that a particle is on a bond with $F=\Gamma'(1+\eta)$ at scale Γ'. This probability is just the length of the bond divided by the average length of bonds, or $D_{\alpha=1}(\eta)=\int d\lambda\lambda P(\eta,\lambda)/\bar{\lambda} = e^{-\eta}\left[(1+2\eta)/3\right]$. The solution to Eq. 17 is of the form $Ae^{-a\eta}+Be^{-b\eta}$. The exact expression turns out to be,

$$D_\alpha(\eta) = e^{-\eta}\left[\frac{5+\frac{2}{\alpha-1}e^{1-\alpha}}{3\alpha^2}\right] + e^{-\alpha\eta}\left[\frac{2e^{1-\alpha}}{3(a-1)}\right]. \tag{18}$$

Eq. 18 is the combination of two exponential terms. The first term is $\exp(-F/\Gamma)$ and is expected from the effective dynamics, whereas the second term is $\exp(-F/\Gamma')$ and demonstrates that the system retains memory of the initial phase at scale Γ'. This form demonstrates aging properties in the variable Γ, which leads to both aging properties in t and thermal memory in T. The probability $D(t,t')$ is finally obtained as $\int d\eta D_\alpha(\eta)$, which simplifies to

$$D\left(t,t'\right) = \left(\frac{\ln t'}{\ln t}\right)^2 \left(\frac{5}{3} - \frac{2}{3}e^{-\left(\ln t/\ln t'-1\right)}\right). \tag{19}$$

Eq. 19 shows that when $\alpha \approx 1$ or when $t' \approx t$, $D(t,t') \sim 7/3 - 4/3\alpha$. In other words, the number of jumps across the origin grows as $4/3\ln t$. This is consistent with past rigorous results [4].

The RSRG procedure can again be applied to calculate the full two-time diffusion front. To do so, we consider quantities that track the end points of bonds, and write the corresponding RG flow equations. Define

$$\Omega^{++}_{\Gamma,\Gamma'}\left(\xi,x_L,x_R,x'_L,x'_R\right)$$

as the probability that the initial point belongs to a descending bond with ends $[-x'_L,x'_R]$ at scale Γ' and to a descending bond with barrier height $\xi = F-\Gamma$ with ends $[-x_L,x_R]$ at scale Γ. There are four such quantities, for the four combinations of descending and ascending bonds at the two scales Γ' and Γ'. From Ω, the two-time diffusion front can be obtained as

$$P\left(x,t,x'>0,t'\right) = \theta\left(x\right)\int d\xi dx_L dx'_L \Omega^{++}_{\Gamma,\Gamma'}\left(x_R = x, x'_R = x'\right)$$
$$+\theta\left(-x\right)\int d\xi dx_R dx'_L \Omega^{-+}_{\Gamma,\Gamma'}\left(x_L = -x, x'_R = x'\right), \quad (20)$$

and similarly for $x'<0$. By the usual logic of keeping track of decimated bonds, but now also keeping track of the direction and the ends of the bonds, the RG flows for the four Ω quantities can be written as

$$\left(\partial_\Gamma - \partial_\xi\right)\Omega^{\pm\varepsilon'}_{\Gamma,\Gamma'} = -2P_\Gamma\left(0\right)\Omega^{\pm\varepsilon'}_{\Gamma,\Gamma'}$$
$$+\int dy dl_1 dl_2 P_\Gamma\left(0,l_2\right)P_\Gamma\left(\cdot,l_1\right)*_\xi \Omega^{\pm\varepsilon'}_{\Gamma,\Gamma'}\left(\cdot,x_L=y\right)\delta\left(x_L-\left(y+l_1+l_2\right)\right)$$
$$+\int dy dl_2 dl_3 P_\Gamma\left(0,l_2\right)P_\Gamma\left(\cdot,l_3\right)*_\xi \Omega^{\pm\varepsilon'}_{\Gamma,\Gamma'}\left(\cdot,x_R=y\right)\delta\left(x_R-\left(y+l_2+l_3\right)\right)$$
$$+\int dl_1 dl_3 dy_1 dy_2 P_\Gamma\left(\cdot,l_1\right)*_\xi P_\Gamma\left(\cdot,l_3\right)\Omega^{\pm\varepsilon'}_{\Gamma,\Gamma'}\left(x_L=y_1,x_R=y_2\right)\times$$
$$\delta\left(x_L-\left(y_1+l_1\right)\right)\delta\left(x_R-\left(y_2+l_3\right)\right), \quad (21)$$

where $\varepsilon' = \pm 1$. The first term on the RHS accounts for the forbidden decimations, while the other terms account for contributions from decimated bonds while keeping track of the ends and directions of the renormalized bonds. The RG flows in the asymmetric case must also carefully distinguish between ascending and descending bonds. The RG flows in Eq. 21 are supplemented by the initial condition at $\Gamma = \Gamma'$,

$$\Omega^{\varepsilon\varepsilon'}_{\Gamma,\Gamma'} = \delta_{\varepsilon\varepsilon'}\delta_{x_L x'_L}\delta_{x_R x'_R}\omega^{\varepsilon'}_{\Gamma'}\left(\xi,x'_L,x'_R\right),$$

where ω is the probability that the initial position belongs to a bond with barrier ξ and ends $[-x'_L,x'_R]$ at scale Γ'. Like before, this probability is $\omega = \int dl' P_{\Gamma'}\left(\xi,l'\right)/\bar{l}_{\Gamma'}\delta\left(l'-\left(x'_L+x'_R\right)\right)$. Since the RG flows involve convolutions of probability distributions, they can be solved by taking the Laplace transform to $x_L\to\mu$, $x_R\to\nu$, $x'_L\to\mu'$, $x'_R\to\nu'$, and using the fixed point solution for the distribution of barrier heights. The equations involve only exponential terms, so we look for solutions that are also combinations of exponential terms, $\tilde{\Omega} = Ae^{-\xi u^\varepsilon_\Gamma(\mu)} + Be^{-\xi u^\varepsilon_\Gamma(\nu)} + Ce^{-\xi u^\varepsilon_\Gamma\left(\mu'\right)} + De^{-\xi u^\varepsilon_\Gamma\left(\nu'\right)}$ where the coefficients A through D depend on ε, ε', and all four Laplace variables. Furthermore, we can use the property that the RG flows are decoupled except for the $\xi = 0$ factors to simplify the RG flows into homogeneous ordinary differential equations. The usual techniques for solving ODEs can then be applied. The linearly independent solutions can be found exactly after tedious algebra that span several pages in the appendix of Ref. [2], which we will not reproduce. The reason that these calculations can be carried out exactly is because the involved integrals are only of exponential terms. The final results can be checked to reduce to the results already derived in the appropriate limits.

There are several regimes. First, $t-t'\sim t'^\alpha$ with $\alpha>1$. Since the second phase is long, the bond containing the initial position is typically decimated between the two scales Γ' and Γ and the particle moves. If $\alpha\to\infty$, the time evolution of x at t and of x' at t' are decoupled. If $\alpha>1$, the two-time diffusion front can be scaled as

$$P\left(x,t,x',t'\right) \sim \frac{1}{\Gamma^4}P_\alpha\left(X = \frac{x}{\left(T\ln t\right)^2}, \tilde{X}' = \frac{x'}{\left(T\ln t\right)^2}\right), \quad (22)$$

where $P_\alpha\left(X,\tilde{X}'\right) = D_\alpha\left(\tilde{X}'\right)\delta\left(X-\tilde{X}'\right) + \tilde{P}_\alpha\left(X,\tilde{X}\right)$. The first contribution comes from particles that do not move between scales Γ and Γ', and the second contribution can be calculated as detailed above. An explicit expression is not provided because the inverse Laplace transform is very complicated. When $\alpha\to\infty$, we recover correctly that $D_{\alpha\to\infty}\left(\tilde{X}'\right) = 0$ and $\tilde{P}_{\alpha\to\infty} = \alpha^2 q\left(\alpha^2\tilde{X}'\right)q\left(X\right)$, where the second contribution has decoupled, implying that the system has lost memory of the first phase [2]. The second regime has $t-t'\sim t'^\alpha$ with $\alpha<1$. In this regime, the particles typically do not move between valleys, but only between equilibrated minima at the bottom of a valley. Motion is therefore the result of rare events such as two neighboring minima separated by a to-be-decimated barrier. These rare events contribute subleading corrections to the effective dynamics of the particles. In obtaining the two-time diffusion front, we have also obtained the trajectory of two independent particles in the same quenched landscape, each diffusing with a different temperature T. If one particle has trajectory $x\left(t\right)$ under temperature T, and another has trajectory $x'\left(t\right)$ at temperature $T'<T$, then the effective dynamics gives that $P\left(x,x',t\right)\sim P_{\alpha=\Gamma'/\Gamma=T'/T}\left(\frac{x}{\ln^2 t},\frac{x'}{\ln^2 t}\right)$. Lastly, the probability distribution can be used to calculate the moments of the displacement $x\left(t\right)-x\left(t'\right)$. For example, we quote from Ref. [2] the mean of the displacement,

$$\overline{\langle x\left(t\right)-x\left(t'\right)\rangle} \sim \left(T\ln t'\right)^2\left[a\left(\alpha\right)+e^{1-\alpha}b\left(\alpha\right)\right], \quad (23)$$

where a and b are polynomials of α.

III. THERMAL MEMORY IN RWRE

We now use these results to investigate two-temperature effective dynamics. First, we corroborate the scaling found above by carrying out the RSRG procedure numerically. We find that indeed, $x \sim \Gamma^2$, see Fig. 3. This scaling implies that the RWRE model has thermal memory. If the system evolves for a time t' at temperature $T_{\max}$, the particle will be found at position $x \sim \Gamma^2 \sim T_{\max}^2 \ln^2 t'$. If the system then evolves to time t, where $t - t' \sim t'^{\alpha}$ with $\alpha > 1$, at temperature T, what is the position of the particle at time T? At large t, the particle position must again scale as $\Gamma^2 \sim (\Gamma' + \Delta\Gamma) \sim T^2 \ln^2 t$. Since the particle is already in a valley with barrier $\Gamma' \sim T_{\max} \ln t'$, the effective dynamics will slow down to approach this scaling at large times. Unfortunately, we were not able to obtain an exact expression for the particle position at short times following the temperature shift. Since the RG flows in Eq. 21 also applies for a two-temperature diffusion front, given more time, we would derive an expression for the displacement in terms of $\alpha \equiv (T \ln t) / (T_{\max} \ln t')$. The expression should resemble Eq. 23, and we expect the dynamics to show dependence on the ratio $T/T_{\max}$, as observed in the experiments discussed in the introduction.

Importantly however, numerics of the actual Arrhenius dynamics on disordered landscapes do not show the predicted $T^2 \ln^2 t$ scaling, see Fig. 3. This is not because the simulations have not reached large enough times, because the dynamics are consistent between short times and long times. Nevertheless, the actual dynamics agree qualitatively with the effective dynamics of the RSRG procedure: 1) At large $t < t'$, the dynamics scale universally as $x \sim (\ln t)^y$ with $y \approx 4.5$. The differences in $T_{\max}$ contribute only an offset. 2) After t', the dynamics are different for the systems that have experienced different $T_{\max}$. Systems that have experienced higher $T_{\max}$ show slower dynamics. For example, in systems that experienced $T/T_{\max} = 0.375$, the particle is trapped for a long time after cooling, whereas in systems that experienced $T/T_{\max} = 0.75$, the particle begins to diffuse shortly after cooling. The dynamics after cooling do not scale with $\ln t$ or t. Given more time, it would be useful to investigate the convergence time for the system to resume scaling with $\ln t$. We expect this convergence time to depend on $T/T_{\max}$.

IV. DISCUSSION

The RSRG procedure applied to RWRE allowed us to extract not only system behavior around some fixed point, but also effective dynamics for the particle that is exact at long times. The procedure can be applied to more complicated models, including random walks on finite and biased landscapes. Other than the diffusion and two-time diffusion fronts, the procedure can also be applied to calculate "recurrence" quantities such as the number of returns to the origin or first passage times that are often of interest for disordered systems.

The effective dynamics obtained show both aging and thermal memory. However, numerics show discrepancies between the actual dynamics and the effective dynamics. We therefore could not corroborate the theory with our numerical results. Further works should begin by clarifying the discrepancies, for example by investigating whether the found scaling depends on the underlying distributions used to generate the disordered landscapes or

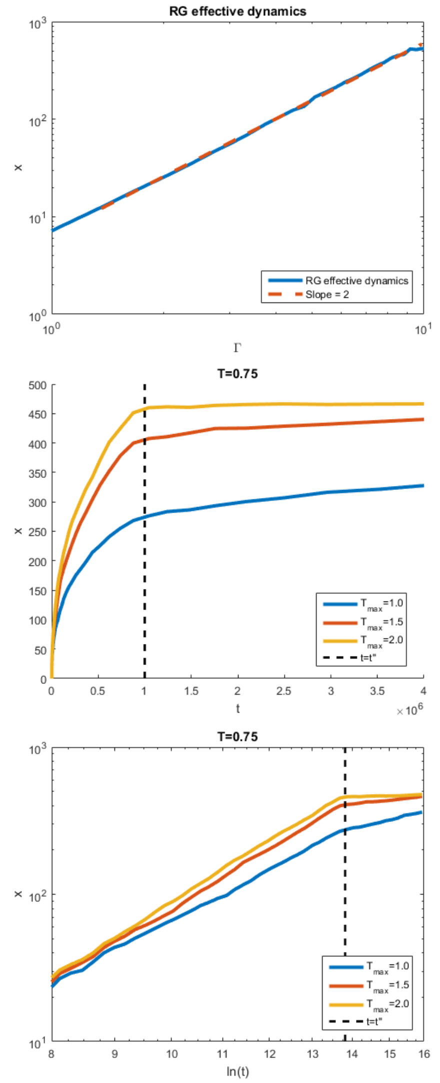

FIG. 3. (Top) Effective dynamics in the RSRG procedure. Numerics of the RSRG procedure show that particle position x, averaged over quenched disorder, indeed scales as Γ^2. (Middle) Simulations of actual Arrhenius dynamics (implemented via the Gillespie algorithm which produces exact trajectories for stochastic processes) on disordered landscapes show slow, logarithmic dynamics that depend on the system temperature. Before $t' = 10^6$, the system is at various values of $T_{\max}$. After t', the system is at $T = 0.75$. (Bottom) Log-log plot of x as a function of $\ln t$. The actual dynamics exhibit thermal memory, since the dynamics after t' depend not only on T, but also on $T_{\max}$.

the underlying Arrhenius dynamics. The existence of a conceptually appealing and analytically solvable RSRG method nevertheless encourages investigating aging and thermal memory phenomena along this direction.

V. A SPIN GLASS MODEL

In tandem with the theoretical effort to predict thermal memory of the glass presented in the previous sections, we consider diffusion on a two-dimensional Ising spin glass. This is the natural generalization of the one dimensional Sinai model, and is justified as the recent experiment was conducted on thin films [1]. In particular, we consider the spin glass Hamiltonian

$$-\beta\mathscr{H} = \sum_{\langle i,j\rangle} \frac{a_{ij}}{T}\sigma_i\sigma_j \tag{24}$$

where the sum runs over all nearest neighbor bonds on the lattice and the bonds are drawn from a uniform distribution. $\sigma_i = 1$ is to be interpreted as a particle and $\sigma_i = -1$ as a vacancy. As a first pass, we encode the dependence on $T_{\max}$ directly in the distribution of bond energies, and assume the width is proportional to $T_{\max}$. Our goal is to understand the manifestation of this dependence of the bond energy distrbution on $T_{\max}$ in the quantitity $\left\langle R^2(t)\right\rangle$, where $R^2(t)$ is the squared distance from the starting position of a tracer particle, and the average is computed over many tracer particles and many realizations of the disorder.

An interesting "atomistic" approach involves fixing the distribution of energy barriers (i.e. no dependence on $T_{\max}$) and simulating the experiment directly. One way to do this is to choose several values $\{T_{\max}^{(1)}, T_{\max}^{(2)}, \ldots\}$. Diffusive dynamics can be simulated at a given $T_{\max}^{(i)}$ for a short time, and the state of the system can be stored and then used for a longer dynamic simulation at a fixed $T < T_{\max}^{(i)}$ for all i. Tracer particles can be tracked in this portion of the simulation where T is identical across simulations. $\left\langle R^2(t, T_{\max}^{(i)})\right\rangle$ can be studied identically as in the description in the previous paragraph, but $P(\Delta E)$, the distribution of energy barriers, could also be computed directly.

This section of the paper is organized as follows. We first provide a brief review of methods for simulating glassy systems, and the inherent challenges therein. We then discuss our chosen method and describe its implementation, commenting on its efficiency and room for extensions and optimizations. We provide results from longer 50×50 simulations for the model in which we assume the distribution of energy barriers has width given by $T_{\max}$ for several values of $T_{\max}$, and demonstrate that diffusion is slower for higher values of $T_{\max}$ as is physically expected. We then provide preliminary results on smaller 25×25 simulations using the "atomistic" approach. Interestingly, we demonstrate that a dependence on $T_{\max}$ emerges naturally from the simulations.

VI. NUMERICAL BACKGROUND

Spin glasses are notoriously difficult to simulate, both in the kinetic Monte Carlo setting and the equilibrium setting [5–7]. The hallmark of a glassy system is a complex energy landscape with numerous metastable states with long escape times. Computationally, this implies that standard Monte Carlo procedures will require intractable simulation length to capture the dynamics [5, 6]. The cause of this phenomenon is simple: the energy barriers to escape a metastable well are in general very high, and so the acceptance probability in a basic Monte Carlo algorithm is disastrously low.

Significantly more progress has been made in spin glass thermalization than kinetics [5]. Algorithms that improve upon equilibration rates seek to avoid the problem of "critical slowing down" which arises in typical single spin flip Monte Carlo sampling. Some methods, such as the Swendson-Wang or Wolff algorithms, build and flip entire clusters of spins at a time, but are not immediately applicable to spin glasses, and in particular not to kinetic spin glass simulations [5]. More complex and recent methods such as parallel tempering and replica-exchange algorithms involve simulating and exchanging multiple copies of the system at different temperatures or configurations. These classes of algorithms have recently enjoyed significant success in spin glass thermalization, but too are inapplicable to kinetic simulations, as they exploit clever but nonphysical dynamics to arrive at a thermalized final state [5, 8, 9].

Indeed, in some sense, the difficulty of kinetic simulations of glasses is unavoidable: we are interested in the dynamics, which by definition are slow. Thermalization, on the other hand, seeks to arrive at a Boltzmann-distributed state by any means necessary, and can do so in any non-physical way so long as the result is sound. An older rejection-free method sometimes known as the N-fold way is essentially the only method to conduct kinetic simulations of a glassy system the authors were able to find to date [10].

VII. NUMERICAL METHOD AND IMPLEMENTATION

We now turn to describe the kinetic Monte Carlo approach we use to simulate the diffusive dynamics, as well as comment on its implementation, which is nontrivial and comprises roughly 950 lines of Python code (not all of which is needed for the generation of the featured results). Our approach is based on the N-fold way [10].

In the context of statistical physics, this method was originally developed by Bortz, Kalos, and Lebowitz as a way to improve the speed of Monte Carlo algorithms for thermalization [10, 11]. The fundamental idea is to employ a rejection-free approach so as to avoid discarded attempts that often take place for low-probability transitions. In switching to a rejection-free approach, particularly for kinetic phenomena, it is important to properly define time. In our case, we anticipate anomalous diffusion due to the uniform distribution of bond strengths. However, because spin exchanges are accepted at every iteration of the algorithm, it is clear that in "simulation time" where one iteration corresponds to $\Delta t = 1$, we will simply observe standard diffusion. Hence, it is paramount to increment time appropriately based on the available transition rates.

This is formalized as follows. We use a *process-list* approach in which we enumerate all possible transitions the system can undergo. For an $N \times N$ grid, there are $2N \times N$ such transitions when considering spin exchanges, where for every particle or vacancy we need only consider the possibility that it exchanges with the particle or vacancy above it or to the right of it. We compute the rate for a given transition as follows:

$$k = e^{-\Delta E/T} \tag{25}$$

i.e., we set units of time such that the jump rate for all transitions is unity, and we consider units such that $k_B = 1$. For the case of the spin glass Hamiltonian, ΔE is computed simply. If we define:

$$\begin{aligned}\mathtt{env}_{ij} = a_{i+1,j}\sigma_{i+1,j} + a_{i-1,j}\sigma_{i-1,j} \\ + a_{i,j+1}\sigma_{i,j+1} + a_{i,j-1}\sigma_{i,j-1}\end{aligned} \tag{26}$$

then $\sigma_{ij} \times \mathtt{env}_{ij}/T$ is the local exponent of the Boltzmann weight ascribed to σ_{ij}. We can then easily compute

$$\Delta E = \sigma_{ij}(\mathtt{env}_{ij} - \mathtt{env}_{i'j'}) + \sigma_{i'j'}(\mathtt{env}_{i'j'} - \mathtt{env}_{ij}) \tag{27}$$

for a swap between σ_{ij} and $\sigma_{i'j'}$. Note that swaps can only occur between σ_{ij} and $\sigma_{i+1,j}, \sigma_{i-1,j}, \sigma_{i,j+1}$ or $\sigma_{i,j-1}$ trivially because we are on a square lattice. Computationally, the shared bond will cancel in Eq. 27 and can be dropped for roughly a factor of $\frac{1}{4}$ speedup (19 floating-point operations are reduced to 15).

The code is organized into three files: `diffuse.py`, `ising.py`, and `union_find.py`. `diffuse.py` contains the necessary functions for the kinetic simulations and `ising.py` contains relevant functions for setting up the simulation as well as some basic thermalization techniques using standard Monte Carlo procedures. `union_find.py` is an efficient implementation of the UnionFind data structure (downloaded, not written by the authors - see the comments in the code) used for implementation of an interesting cluster identification algorithm ultimately unneeded for the generated results.

We start with an initially random configuration of particles and vacancies with a specifiable concentration of particles. This is created in the function `instantiate_grid()` in the file `ising.py`. We simply begin with a grid of vacancies and repeatedly convert random vacancies into particles until we reach the desired particle density. We also draw a symmetric matrix of bond energies a_{ij} from a uniform distribution with the basic function `draw_bonds()`.

A linear array of transition rates conveniently named `rates` is computed in the function `compute_rates()` by looping over the grid and computing the up-switch and right-switch transition rate with the function `compute_ur_rate()` for every spin on the grid using Eqs. 25 and 27. For every spin (i, j), we first store the up transition rate and then the right transition rate. Hence, the two rates for spin (i, j) can be obtained as $\mathtt{rates}[2 * (i + N * j) + \{0, 1\}]$ where the curly braces indicate the two possibilities.

To select what transition actually occurs at a given timestep, we proceed as follows. We create an array of cumulative sums of the available rates. We then draw a random number uniformly from the interval $(0, \sum_i k_i)$ where the sum runs over all possible transition rates. We select the event corresponding to the first partial sum larger than the random number so that events are chosen with probability proportional to the magnitude of their rate. To determine the grid elements involved in the switch and the direction of switch, we use the formula $l = 2 * (i + N * j) + \{0, 1\}$ and a combination of modular and integer arithmetic for l the index of the partial sum chosen. The time increment Δt is determined stochastically by drawing $\Delta t \sim \mathrm{Exp}(\sum_i k_i)$, i.e., we draw from an exponential distribution with rate parameter given by the sum of all transition rates for all possible transitions on the lattice [5, 6, 11].

After each event, the available transition rates change locally. Some thought demonstrates that the only rates which change are those of the next nearest neighbors of the particle and vacancy involved in the swap. Given the indices and hop direction of the *particle* involved, the function `get_nnn()` returns a list of the grid indices whose rates need to be recomputed. This is used by `update_rates()` to update only the rates which have changed from the previous timestep. This procedure can then be iterated: at every timestep, we pick a transition using the random procedure previously described, update the rates, and continue.

There are a few caveats that must be addressed with the rate calculation. First, vacancy-vacancy exchanges involve an energy change of zero but are non-physical; hence we set such rates to zero.

Particle-particle exchanges, on the other hand, are highly physical. In fact, in a ferromagnetic model, we expect the formation of clusters as the diffusive dynamics proceeds, and within a cluster there will be many particle-particle exchanges. It is clear from Eq. 27 that particle-particle exchanges incur an energy cost of zero and hence occur with a rate of 1. Thus, within a cluster, there will be standard diffusion even in the correctly incremented time. This is problematic from a simulation point of view, as we are interested in extracting $\langle R^2(t) \rangle$ for a number of tracer particles. If we allow particle-particle exchanges, the interesting glassy dynamics which occur due to low-probability transitions will essentially never occur, and our simulation (unless run for an extraordinary amount of time) will simply yield a slight modification to standard diffusion within the clusters.

A simple approach to correct this is to set particle-particle rates to zero, as we did for vacancy-vacancy rates. This leads to a more subtle issue: in all likelihood, a tracer particle will end up in a cluster at some point, or perhaps even start within a cluster. If we neglect particle-particle transitions, the only way for a tracer particle to escape the interior of a cluster is for the cluster to shatter; this is nonphysical, and can take an extreme amount of time. In reality, the particle will diffuse throughout the cluster rapidly on the timescale of the glassy transitions. Hence, it will spend time on the boundary, and can easily escape by hopping to a vacancy while it is on the boundary.

We handle this using what we deem a "well-mixed" approximation. We set all particle-particle transition rates to be equal to zero. However, for every switch, we search for and identify the cluster to which the hopping particle belongs. We uniformly select from the cluster the particle that made this transition, and record the modified vector distance update if this uniformly selected particle is one of the tracer particles, as well as update its current position. The tracking of and updates to the tracer particles can be found in the function `diffuse_lt_samp()`. Note that this procedure is an approximation; in reality, it would be necessary to solve a diffusion equation on the domain of the cluster. We expect that this approximation is unfaithful to the true system dynamics but will not destroy a dependence on $T_{\max}$, and in future work it would be interesting to identify the effect of this approximation and understand how it can be improved.

The cluster identification is implemented using a basic breadth-first search on the lattice computed by the function `identify_cluster()`. An alternative method, the Hoshen-Kopelman algorithm, is implemented in the function `hoshen_kopelman()` using the attached UnionFind data structure implementation in `union_find.py`. While the Hoshen-Kopelman algorithm is a fast way to identify all clusters on the grid, it is clear that two or three clusters can be shattered or joined with a single exchange. Because of this fact, there is no simple way to precompute all clusters on the grid and modify them on the fly; to identify the cluster to which a particle belongs, the cluster must be be recomputed on each iteration. In the worst case, if there were a system-spanning cluster, the breadth-first search would take asymptotically as much time as the Hoshen-Kopelman algorithm, and hence BFS is more efficient on average. The Hoshen-Kopelman algorithm is left included for possible extensions to this work.

Even accounting for particle-particle and vacancy-vacancy swapping as described above, we still encounter the so-called "low-barrier problem" [6, 11]. At every timestep, there are consistently several transitions with rates orders of magnitude higher than all others rates that appear in the problem. Left alone, the presence of these highly probable transitions leads to repeated switching back and forth between several states of the system. In other words, the system becomes trapped in a "superbasin" [7, 11]. We take the simplistic approach of artificially raising these lower barriers (lowering the rates) according to a certain specifiable rate threshold. This approximation should be valid, as we expect that these fast transitions will equilibrate on the timescale of interest in the problem [11]. Nevertheless, it would be interesting to probe the effects of thresholding by using more advanced methods, such as those based on hashing the system state to prevent repeatedly revisiting the same states, or those based on transition matrix diagonalization to escape superbasins [11].

Finally, we discuss the method used to track tracers. We randomly select `n_tracers` tracers whose trajectories we track over the course of a simulation. To do so, we keep track of $R_i^2(t)$ where the index i runs over the tracers. We store only the *current* vector (x_i, y_i) distance from the tracer's starting location but the entire $R_i^2(t)$ trajectory. This distance is incremented carefully, accounting for periodic boundary conditions and the larger jumps due to hopping out of a cluster. There is a subtle issue that arises here when considering our goal of averaging over the disorder across simulations. Because the time increment is stochastic, the time points at which the trajectory is stored will not be consistent across simulations. This makes averaging over the disorder challenging *a-priori*.

This is alleviated as follows. We discretize the logarithm of time into points $\log(t)_j$ (note the index is outside the argument of the logarithm). Say tracer particle l's location was last stored at time t_i and say that $\log(t)_k = \max_j(\log(t)_j)$ such that $\log(t)_j < \log(t_i)$. We then set $R_l^2(\log(t)_k) = R_l^2(t_i)$. We ensure the discretization is fine enough such that $\log(t)_{k+1} < \log(t_{i+1})$ where t_{i+1} is the next time *any* tracer particle moves (i.e., there are many discretization points between stochastic time increments at which we record data). If tracer particle l' is the particle that moves at t_{i+1}, we find the k' such that $\log(t)_{k'} < \log(t_{i+1}) < \log(t)_{k'+1}$. We fill in the trajectories of all other tracer particles $m \neq l'$ such that $R_m^2(\log(t)_q) = R_m^2(\log(t)_k)$ for all $k < q \leq k'$. For tracer particle l', we do the same for all q such that $k < q < k'$, and set $R_{l'}^2(\log(t)_{k'}) = R_{l'}^2(t_{i+1})$. In this way, we ensure that all trajectories across all simulations are sampled at identical time points, and hence can average directly. This is all handled in the function `diffuse_tracers_lt_samp()`.

VIII. RESULTS AND DISCUSSION

The model in Eq. 24 was simulated using the code and methods described in the previous section to understand the dependence of the disorder and tracer particle averaged $\langle R^2(t)\rangle$ on T_{max}. Individual realizations of the disorder were simulated in parallel using the `map` and `Pool` functionalities provided by the Python `multiprocessing` library. Simulations were run on ThinkMate workstations with 2016 Intel Xeon processors. Each workstation was able to simulate forty realizations of the disorder simultaneously and simulations were spread over three workstations.

Due to the cluster search, scaling with system size can be very poor; we hence restrict ourselves to an antiferromagnetic model to minimize the size and formation of clusters. We also expect the "well mixed" approximation to be more valid in the antiferromagnetic case for these reasons.

Simulations in the T_{max}-dependent model (a_{ij} drawn uniformly over the interval $(-T_{\text{max}}, 0)$) were conducted on a 50×50 grid of spins with a particle density of $\rho = .75$ and periodic boundary conditions in both dimensions. Diffusive dynamics were simulated at a similarly arbitrary $T = .75$ in each case. Each simulation was allowed to run for 50 Monte Carlo steps (MCS), or steps per particle, totalling $50^3 = 125,000$ steps per simulation. Optimizations that enable scaling up to larger system sizes to understand finite size effects are of significant interest to the authors for future work.

1625 tracer particles were tracked per realization of the disorder. Tracer particle trajectories were averaged in addition to the disorder average, leading to a total of $65,000$ trajectories per value of T_{max}. Each realization of the disorder took roughly 12 hours of wall time to complete. Results for five values of $T_{\text{max}} = .75, 1, 2, 3, 7.5$ are included in Fig. 4. Note that higher values of T_{max} lead to slower diffusion (smaller $\langle R^2(t)\rangle$). Assuming slower diffusion would lead to slower relaxation rates after a perturbation, this precisely corroborates the hypothesis of the experiment.

The trajectory differences apparent in Fig. 4 are most pronounced at long times. This makes intuitive sense; ultimately, the larger barriers for higher T_{max} prevent diffusion of the tracer particles to far distances even for long times, while the lower barriers for lower values of T_{max} eventually enable escape.

We present some preliminary results from the "atomistic" approach in Fig. 5. Due to time constraints, simulations were conducted on a 25×25 grid. 437 tracer particles were followed per simulation with again 40 realizations of the disorder simulated in parallel per value of T_{max}, corresponding to 17480 tracer particles per value of T_{max}. Simulations were allowed to proceed for 20 MCS at T_{max}. The state of the system was then saved and simulated for 100 MCS at $T = .5$ where diffusive data was recorded. Each realization of the disorder took roughly one hour of wall time on the same 2016 Intel Xeon CPU. The width of the uniform distribution was taken to be 5 in all cases ($a_{ij} \in (0, 5)$). The particle density is $\rho = .75$. Interestingly, a T_{max} dependence emerges organically from the simulation. Note that the dependence of $\langle R^2(t)\rangle$ looks more like a power law of $\log(t)$ in these smaller simulations than in the 50×50 simulations. It is unclear if this is a finite size effect or if the "curling away" present in Fig. 4 at long times would look similar if carried out for even longer times. This needs to be investigated further in future work with optimizations that enable longer simulations and larger systems. More discussion can be found in the caption of Fig. 5.

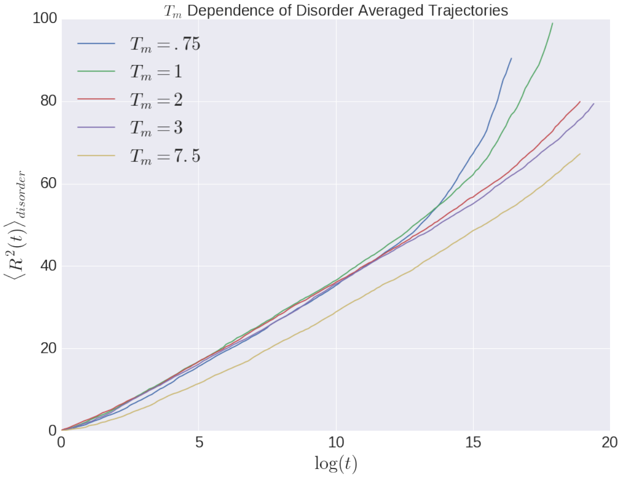

FIG. 4. Depiction of the dependence of $\langle R^2(t)\rangle$ on T_{max} in the model where T_{max} is treated as the width of the uniform distribution from which the bond energies are drawn. Results were computed on a 50×50 grid. There is a clear qualitative difference in the trajectories for different values of T_{max}, in that diffusion is slower for higher T_{max}.

IX. CONCLUSIONS AND FUTURE DIRECTIONS

Via kinetic Monte Carlo methods, we demonstrated a dependence of diffusion in the two dimensional Ising spin glass on the maximal temperature the glass has been exposed to. In the first case, we showed that stronger bond strengths on average, as expected, lead to verifiably different trajectories of $\langle R^2(t)\rangle$. We then showed that one can produce a T_{max}-dependent $\langle R^2(t)\rangle$ trajectory directly from the kinetics of the problem by allowing the system to dynamically evolve at a fixed T_{max} and then computing the trajectory over some $T < T_{\text{max}}$ in analogy to the experiment of Ref. [1]. Together, these simulations provide

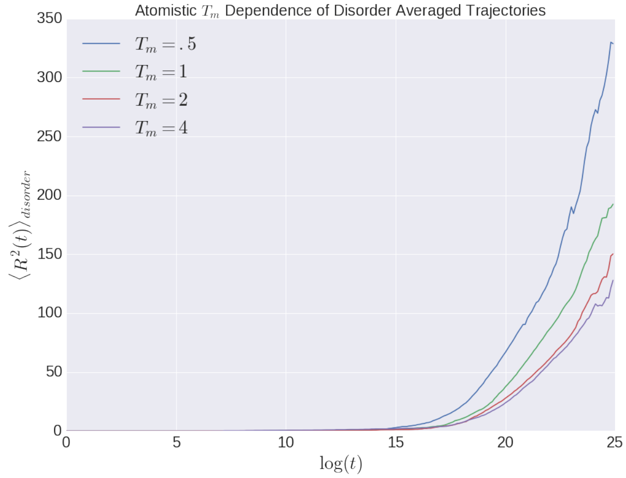

FIG. 5. Depiction of the qualitative dependence of $\langle R^2(t) \rangle$ on $T_{\max}$ in the “atomistic” approach where the distribution of bond energies is held fixed, but the system is allowed to equilibrate at a higher temperature before the dynamics are tracked at a consistent lower temperature. These are preliminary results from a small 25×25 grid. Trajectories scale as a power law in $\log(t)$. Curve fitting provides $\langle R^2(t) \rangle \sim \log(t)^8$, which is interesting to compare to the theoretical one dimensional result of $x \sim \log(t)^2$. However, due to the small system size, it is unclear if this law should be trusted. Note that there is a clear difference in the trajectories for different values of $T_{\max}$, and that this emerges naturally from the simulation.

further support for the hypothesis that exposing the glass to higher values of $T_{\max}$ allows for the control of the level of equilibration of the glass, and hence the strength of the bonds on average.

The majority of this work involved developing the code to be able to conduct these simulations. Now that the code has been developed and the expected dependence on $T_{\max}$ realized, many possibilities are open. Future work involves optimizations for scaling to larger system sizes, categorizing the dependence on $T_{\max}$ when it enters explicitly as a parameter, understanding the effects of rate thresholding and exploring more advanced methods to overcome the low barrier problem, considering the effects of the choice of distribution of energy barriers, comparing antiferromagnetic and ferromagnetic models, and understanding the effects of ρ and T on the results. Finally, it is very interesting to explore the “atomistic” approach further, particularly on larger grids, and see if a quantitative dependence on $T_{\max}$ can be extracted from some relevant quantity computable from the simulation. In the ideal case, the atomistic approach can be combined with a method to introduce a perturbation in the system, from which relaxation can be measured as in the experiment.

X. AUTHOR CONTRIBUTIONS AND ACKNOWLEDGMENTS

NB focused on the 2D Ising spin glass model and Monte Carlo simulation methods. PH focused on RWRE. Both authors discussed both aspects of the project together and wrote the paper equally. The authors thank Mehran Kardar, Ariel Amir, and Chris Rycroft for important discussions.

[1] A. Eisenbach, T. Havdala, J. Delahaye, T. Grenet, A. Amir, A. Frydman, Glassy dynamics in disordered electronic systems reveal striking thermal memory effects, Phys Rev Lett 117 (2016) 116601.

[2] P. LeDoussal, C. Monthus, D. S. Fisher, Random walker in one-dimensional random environments: Exact renormalization group analysis, Phys Rev E 59 (5).

[3] Y. G. Sinai, The limiting behavior of a one-dimensional random walk in a random medium, Theory of Probability and its Applications 27 (2).

[4] A. O. Golosov, Limit distributions for random walks in random environments, Soviet Math Dokl.

[5] D. P. Landau, K. Binder, A Guide to Monte Carlo Simulations in Statistical Physics, 4th Edition, Cambridge University Press, 2014. doi:10.1017/CBO9781139696463.

[6] H. G. Katzgraber, Introduction to Monte Carlo MethodsarXiv:0905.1629.
URL http://arxiv.org/abs/0905.1629

[7] P. Kratzer, Monte Carlo and Kinetic Monte Carlo Methods A Tutorial, Multiscale Simulation Methods in Molecular Sciences - Lecture Notes 42 (2009) 51–76.

[8] J. Houdayer, A cluster monte carlo algorithm for 2-dimensional spin glasses, The European Physical Journal B - Condensed Matter and Complex Systems 22 (4) (2001) 479–484. doi:10.1007/PL00011151.
URL http://dx.doi.org/10.1007/PL00011151

[9] Z. Zhu, A. J. Ochoa, H. G. Katzgraber, Efficient cluster algorithm for spin glasses in any space dimension, Phys. Rev. Lett. 115 (2015) 077201. doi:10.1103/PhysRevLett.115.077201.
URL https://link.aps.org/doi/10.1103/PhysRevLett.115.077201

[10] A. Bortz, M. Kalos, J. Lebowitz, A new algorithm for Monte Carlo simulation of Ising spin systems, Journal of Computational Physics 17 (1) (1975) 10–18. doi:10.1016/0021-9991(75)90060-1.
URL http://www.sciencedirect.com/science/article/pii/0021999175900601

[11] A. Voter, Introduction To the Kinetic Monte Carlo Method, Radiation Effects in Solids 235 (2007) 1–23.
URL http://www.springerlink.com/index/u62k8537u4636118.pdf