**SUPPORTING INFORMATION**

1. **Details of Matrix Calculations**

The size of the coarse-grained rate matrix **A**(diff) presented a numerical challenge, particularly in the case of the 2D model. Out of the (*N* +1)2 microcanonical states in (*q*, *E*)-space, somewhat less than half are physically accessible (Fig 2, main text), but this still generates very large rate matrices. The number of accessible states *l* was substantially reduced by imposing a free energy cutoff *Wmax*. In order to avoid artificially bottlenecking the barrier peak, *Wmax* was assigned a value at least 150 meV greater than the saddle point energy. Converting **A**(2D) to band form substantially increased packing efficiency. We numbered states in column order in the directions of increasing *m* and *n*(*m*). This resulted in a maximum spatial bandwidth *d* equal to the width of the separatrix. The final dimensions of the banded 2D rate matrix **A**(2D) were *l* × 2*d*+1. Typical values for *l* and *d* were ~20,000 and ~70. By taking advantage of eigenvalue and linear routines designed for symmetric band matrices, *k* could be computed in seconds on a desktop computer. The same method was applied to 1D calculations, where **A**(1D) is tridiagonal (*d* = 1) and *l* = *N* + 1.

1. **Error Analysis**

Despite agreements between different methods of computing *k*, it was important to rule out systemic biases in the coarse-graining process. We identified two potential sources of error. The first is poor choice of pseudorandom number generator, leading to erroneous Monte Carlo simulation1,2. To address the possibility of biased sampling, we tested several generators: (1) Numerical Recipes© long-period L’Ecuyer generator with Bays-Durham shuffle (ran2)3; (2) Intel® MKL Library SIMD-oriented Fast Mersenne Twister (FMT)4; (3) Intel® MKL Library multiplicative congruential generator (MCG)5; and (4) Intel® MKL Library combined multiple recursive generator (MRG)6. Rate constant values from three of the four generators were in close agreement across all methods (Fig. S1). The MCG algorithm exhibited greater variability and systemic biases compared to the other three generators and is not recommended for use in our model. Most simulations employed the ran2 algorithm. In the small number of cases with large barriers where the performance of the random number generator was in doubt, repeating the simulations with the FMT and MRG algorithms confirmed the ran2 results.

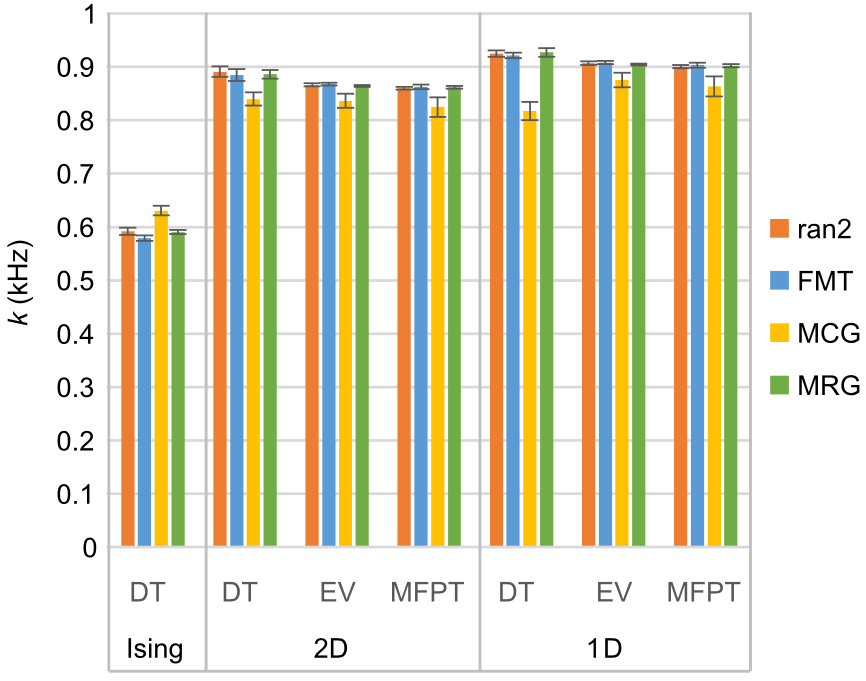


FIG. S1. Forward rate constant *k* (mean ± standard error, *n* = 6) calculated for standard model with different methods (DT = dwell time, EV = eigenvalue analysis, MFPT = mean first passage time) and as a function of pseudorandom number generator. Results were consistent between generators except in the case of the multiplicative congruential generator (MCG), which demonstrated lower accuracy and precision compared to the other three generators.

A second source of potential error related to a rebalancing procedure used to maintain equilibrium relations. Since branching coefficients were obtained from MMC simulation, statistical error was unavoidable. In order to re-establish detailed balance between opposing diffusion rates *akk*′ and *bk*′*k* in accordance with known equilibrium probabilities *pk*, the rates needed to be rebalanced according to the following prescription:

,

,

where *a* and *b* are the adjusted rate constants. Rebalancing preserves the correct equilibrium distribution while maintaining the geometric mean between rate constants. For small relative errors *a* and *b* in the unbalanced rates *akk*′ and *bk*′*k*, the first order relative error in each balanced rate is: (1 + *a* + *b*)1/2 – 1. Thus systemic bias should be minimized if *a* and *b* are themselves unbiased. In practice, omitting the rebalancing step prior to mfpt or dwell time analysis yielded at most a 4% difference (*n* = 6). However, in the case of 2D eigenvalue analysis, the change was significantly larger (9.9-fold increase for unbalanced rates), which was reduced to 30% through noise reduction by using pooled rate constants from 30 experiments. The error in calculating 2D eigenvalues is attributable to use of eigenvalue routines designed for symmetric matrices, as these require strict detailed balance prior to symmetrization7. This was confirmed by observing that**o obtained from the singular matrix **A** was nonzero unless rebalancing was performed. The problem did not extend to 1D eigenvalues because circulatory flow is impossible in 1D, and therefore detailed balance is automatically satisfied. In summary, rate rebalancing was crucial for computing 2D eigenvalues but did not significantly affect other calculations, which, independent of rebalancing, yielded consistent outcomes.

1. **NUMERICAL STRATEGY FOR COMPUTING *k*(, *T*)**

A proposed strategy for computing *k* as a function of  and *T* is summarized by Eq. 33 in the main text, reproduced below.

.

The full Ising dynamics is coarse-grained onto the microcanonical space (*q*, *E*) through Metropolis Monte Carlo sampling as described in the main text. From there, *Db*, *ZR*-1 and *Zb*-1 are computed in the (*,* *T* ensemble. To compensate for coarse-graining error **, one finds the slowest (reactive) spectral component *D*1 of the barrier diffusion coefficient, from which the association *D*1/*Db* ≈ ** is made. The last equation is empirical, but makes intuitive sense and gives numerically correct answers for small barriers in two- and three-dimensional Ising lattices. The *ZR* and *Zb* are local partition functions that sum over individual basin and barrier regions. Their product can be rewritten as exp(*W*‡/*kT*), where the activation free energy*W*‡ = *Wb*  *WR* is the difference between the barrier free energy *WR* = *kT*ln*ZR* and the reactant free energy *Wb* = *kT*ln*Zb*. The activation free energy can be further broken down into its thermodynamic components: *W*‡ =*E*‡ ***S*‡ *q*‡, where activation quantities are defined as *X*‡ = *Xb* *XR*. The regional values of extensive variables are obtained by Boltzmann averaging over the region of interest. Thus *Ex* = 〈*E*〉*x*, *qx* = 〈*q*〉*x*, and *Sx* = *k*〈ln*p*〉*x*, where *x* = {R, b, P} defines the region of summation. In our study, ensemble averaging was performed over microcanonical ensemble states (*m*, *n*)*x* with regional probability distribution functions *pmn* = *Zx*1exp[±(*Em* *k*ln*mnT* *qm*)/*kT*], where the positive sign in the exponent corresponds to barriers and the negative sign to basins. We defined the reactant state as having *q*-values in the range [0, *q*\*] and defined the barrier region by [*q*1, *q*2]. The reactant and barrier ranges overlap, but for well-separated regional probability distributions (Fig. S2a), this has minimal effect on the outcome. Assigning regional states reduced the coarse-grained potential of mean force to a discrete “sawtooth” landscape defined by three (*q*, *W*) coordinates. It is not a requirement that the landscape be piecewise parabolic. The temperature dependence of *W*‡ and its components are shown Fig, S2b, c. The rate constant expression can be rewritten as *k* = *D*1exp(–*W*‡/*kT*).

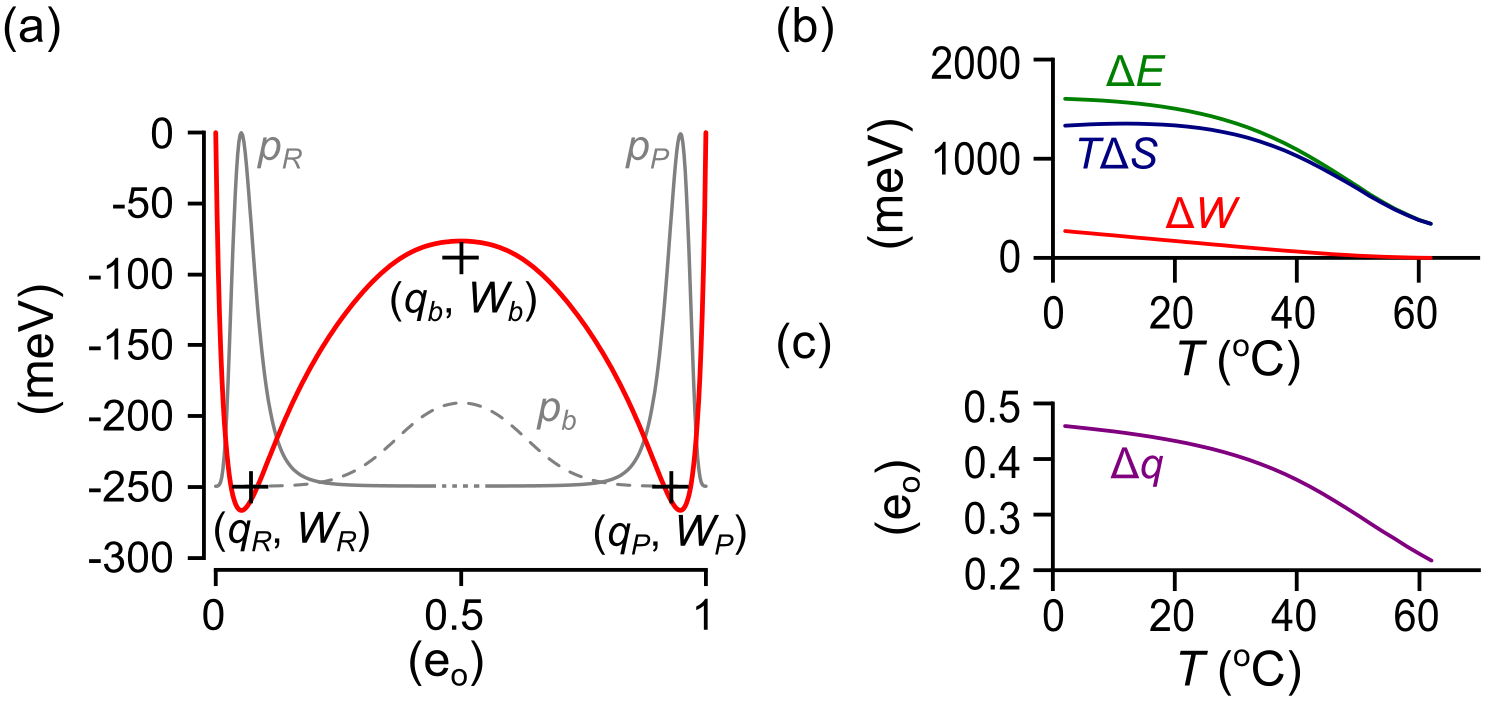


FIG. S2. Discretization of the 1D diffusion landscape into a “sawtooth” landscape (reference model). (a) Black crosses marking regional reaction coordinates (*qx*, *Wx*), which are superimposed onto *W*(*q*) (red curve). Note the small displacements of (*q*, *W*) coordinates from their corresponding *W*(*q*) extrema. Segmental equilibrium probability distributions are colored gray (*x* = R, P: solid curves; *x* = b: dashed curve). (b) Temperature dependence of reaction energies. (c) Temperature dependence of reaction charge.

We should note that in the Ising model the diffusion coefficient *D*(*q*) is nearly constant across the barrier region (Figs. 8 and 15, main text). A model in which *D* varies strongly across the barrier region would add a “spurious drift” term *kT*ln(*D*/*D*1) to the exponent of *Zb*8,9. This does not change the equilibrium distribution between stable states, since *ZR* and *ZP* are unaffected.

**REFERENCES**

1 P.D. Coddington, Int. J. Mod. Phys. C **05**, 547 (1994).

2 A.M. Ferrenberg, D.P. Landau, and Y.J. Wong, Phys. Rev. Lett. **69**, 3382 (1992).

3 W. Press, S. Teukolsky, W. Vetterling, B. Flannery, E. Ziegel, W. Press, B. Flannery, S. Teukolsky, and W. Vetterling, *Numerical Recipes in C*, 2nd ed. (Cambridge University Press, New York, 1992).

4 M. Saito and M. Matsumoto, in *Monte Carlo Quasi-Monte Carlo Methods 2006* (Springer-Verlag, Berlin, Heidelberg, 2008), pp. 607–622.

5 P. L’Ecuyer, Math. Comput. **68**, 249 (1999).

6 P. L’Ecuyer, Oper. Res. **47**, 159 (1999).

7 E. Frehland, Biophys. Chem. **8**, 255 (1978).

8 N.G. Van Kampen, *Stochastic Processes in Physics and Chemistry*, 2nd ed. (North-Holland Personal Library, Amsterdam, 1992).

9 B. Peters, P.G. Bolhuis, R.G. Mullen, and J.-E. Shea, J. Chem. Phys. **138**, 054106 (2013).