Exchange Monte Carlo: An efficient workhorse for optimization problems
Exchange Monte Carlo: An efficient workhorse for optimization problems

Helmut G. Katzgraber

http://katzgraber.org
Exchange Monte Carlo: An efficient workhorse for optimization problems

Helmut G. Katzgraber

http://katzgraber.org
Outline

• Brief overview of optimization problems:
  • Why do we need finite-temperature algorithms?

• Exchange (parallel tempering) Monte Carlo:
  • Advantages over simple Monte Carlo.
  • Algorithm description.

• Tuning the method:
  • Traditional vs feedback-optimized.
  • Some extensions and applications.

• Presented research done in collaboration with D. Huse, S. Trebst and M. Troyer.
Outline

- Brief overview of optimization problems:
  - Why do we need finite-temperature algorithms?

- Exchange (parallel tempering) Monte Carlo:
  - Advantages over simple Monte Carlo.
  - Algorithm description.

- Tuning the method:
  - Traditional vs feedback-optimized.
  - Some extensions and applications.

- Presented research done in collaboration with D. Huse, S. Trebst and M. Troyer.
Outline

- Brief overview of optimization problems:
  - Why do we need finite-temperature algorithms?

- Exchange (parallel tempering) Monte Carlo:
  - Advantages over simple Monte Carlo.
  - Algorithm description.

- Tuning the method:
  - Traditional vs feedback-optimized.
  - Some extensions and applications.

- Presented research done in collaboration with D. Huse, S. Trebst and M. Troyer.
Exchange vs parallel tempering

- **History:**
  - The method was developed in 1996 by Hukushima and Nemoto. They called it “exchange Monte Carlo.”
  - Marinari & Parisi developed a similar method called simulated tempering and suggested a variation they dubbed “parallel tempering Monte Carlo”.
  - Recently, I discovered that Geyer presented another version in 91 in a proceedings book. So far, I could not get my hands on it...

- **Conundrum:**
  - What should we call it?
  - Traditionally it is called “parallel tempering.”
  - Politically correct probably “exchange Monte Carlo.”
Optimization problems
Overview of optimization problems

- Typical goals one usually wants to tackle:
  - Minimize a cost function of a problem (e.g., calculate the ground-state energy of a system).
  - Compute an observable (e.g., energy, magnetization, ...) at low temperature.

- Typical optimization problems:
  - Physical, biological, chemical systems
  - k-SAT
  - NPP
  - Vertex cover problem
  - TSP, ...

\[
\sum_{i \in A_1} a_i = \sum_{i \in A_2} a_i
\]

\[
A = \{a_1, a_2, \ldots a_n\} \rightarrow A_1 \cup A_2
\]

3-SAT

(\text{x}_{11} \text{ OR } \text{x}_{12} \text{ OR } \text{x}_{13}) \text{ AND } (\text{x}_{21} \text{ OR } \text{x}_{22} \text{ OR } \text{x}_{23}) \text{ AND...}
Several physical problems which contain randomness can be classified as (often hard) optimization problems. Due to the randomness, one obtains competing interactions and thus a complex energy landscape.

Examples:

- Spin glasses: $\mathcal{H} = -\sum_{ij} J_{ij} S_i S_j$
  
- Structural glasses
- Polymers in random media (interfaces)
- Biomolecules (proteins, ...)
- Quantum wave functions
- Reconstruction of geological structures from seismic measurements, ...
Several physical problems which contain randomness can be classified as (often hard) optimization problems.

Due to the randomness, one obtains competing interactions and thus a complex energy landscape.

**Examples:**
- Spin glasses: \[ \mathcal{H} = - \sum_{i,j} J_{ij} S_i S_j \]
  - \( \mathcal{P}(J_{ij}) \) random
- Structural glasses
- Polymers in random media (interfaces)
- Biomolecules (proteins, ...)
- Quantum wave functions
- Reconstruction of geological structures from seismic measurements, ...
What will be discussed here?

- **So far:**
  - Methods to compute ground states of complex systems (zero-temperature methods).
  - Methods to study phase transitions and overcome critical slowing down (for example cluster algorithms).
  - Methods to tackle systems which undergo first-order transitions (for example Wang-Landau algorithm).
  - Methods for infinite temperature (series expansions).

- **In this lecture:**
  - Study systems with rough energy landscapes at low temperatures.
  - Introduction to exchange (parallel tempering) Monte Carlo.
What will be discussed here?

- **So far:**
  - Methods to compute ground states of complex systems (zero-temperature methods).
  - Methods to study phase transitions and overcome critical slowing down (for example cluster algorithms).
  - Methods to tackle systems which undergo first-order transitions (for example Wang-Landau algorithm).
  - Methods for infinite temperature (series expansions).

- **In this lecture:**
  - Study systems with rough energy landscapes at low temperatures.
  - Introduction to exchange (parallel tempering) Monte Carlo.
Simple benchmark model family
Prototype for a magnet: the Ising model

- **Hamiltonian:**
  \[ H = - \sum_{\langle ij \rangle} J_{ij} S_i S_j - H \sum_i S_i \]
  \[ J_{ij} = 1 \quad \forall i, j \quad i \neq j \]

- **Order parameter:**
  \[ m = \frac{1}{N} \sum_i S_i \] (magnetization)

- **Some properties:**
  - Describes many magnetic systems (also other spin symmetries).
  - Exact solution in 2 space dimensions (Onsager).
  - Nearest-neighbor interactions, in 2D \( T_c \sim 2.27 \).
Adding frustration...

- **General Hamiltonian:**
  \[ \mathcal{H} = - \sum_{\langle ij \rangle} J_{ij} S_i S_j \]
  - Blue line: +J
  - Red line: −J

- **Introduce frustration between the spins:**
  - Ferromagnet
  - Fully-frustrated

**Properties of the fully-frustrated Ising model:**
- Huge ground-state degeneracy and complex energy landscape.
- \( T_c = 0 \) in 2D.
- \( \prod J_{ij} < 0 \quad \forall \ i, j \)
- What happens if we add randomness to the frustration?
Adding frustration...

- **General Hamiltonian:**
  \[ H = - \sum_{\langle ij \rangle} J_{ij} S_i S_j \]

- **Introduce frustration between the spins:**

- **Properties of the fully-frustrated Ising model:**
  - Huge ground-state degeneracy and complex energy landscape.
  - \( T_c = 0 \) in 2D.
  - \( \prod_{ij} J_{ij} < 0 \quad \forall \ i, j \)
  - What happens if we add randomness to the frustration?
Adding frustration...

- **General Hamiltonian:**
  \[ H = - \sum_{\langle ij \rangle} J_{ij} S_i S_j \]

- **Introduce frustration between the spins:**

  ![Diagram of spins with frustration](image)

  **ferromagnet**
  ![Diagram of fully-frustrated model](image)
  **fully-frustrated**

- **Properties of the fully-frustrated Ising model:**
  - Huge ground-state degeneracy and complex energy landscape.
  - \( T_c = 0 \) in 2D.
  - \( \prod_{ij} J_{ij} < 0 \quad \forall \ i, j \)

- **What happens if we add randomness to the frustration?**
Adding frustration and disorder: spin glass

- **Spin-glass Hamiltonian:**
  \[ \mathcal{H} = - \sum_{\langle i,j \rangle} J_{ij} S_i S_j \quad S_i \in \{\pm 1\} \]

- **Details about the model:**
  - Unconventional “order.”
  - Only mean-field solution.
  - Bimodal random bonds: high degeneracy.
  - Gaussian random bonds: unique ground state.
  - 2D \( T_c = 0 \)
  - 3D \( T_c = 0.951(2) \)
  - 4D \( T_c = 1.805(10) \)

- Will be discussed in detail tomorrow and not discussed here further.
Reminder: Simple Monte Carlo
Equation of State Calculations by Fast Computing Machines

Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, and Augusta H. Teller,
Los Alamos Scientific Laboratory, Los Alamos, New Mexico

AND

Edward Teller, * Department of Physics, University of Chicago, Chicago, Illinois
(Received March 6, 1953)

A general method, suitable for fast computing machines, for investigating such properties as equations of state for substances consisting of interacting individual molecules is described. The method consists of a modified Monte Carlo integration over configuration space. Results for the two-dimensional rigid-sphere system have been obtained on the Los Alamos MANIAC and are presented here. These results are compared to the free volume equation of state and to a four-term virial coefficient expansion.

I. INTRODUCTION

The purpose of this paper is to describe a general method, suitable for fast electronic computing machines, of calculating the properties of any substance which may be considered as composed of interacting individual molecules. Classical statistics is assumed,

II. THE GENERAL METHOD FOR AN ARBITRARY POTENTIAL BETWEEN THE PARTICLES

In order to reduce the problem to a feasible size for numerical work, we can, of course, consider only a finite number of particles. This number N may be as high as several hundred. Our system consists of a square† con-

• They knew this was important... and they were right!
Equation of State Calculations by Fast Computing Machines

Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, and Augusta H. Teller,
Los Alamos Scientific Laboratory, Los Alamos, New Mexico

AND

Edward Teller,* Department of Physics, University of Chicago, Chicago, Illinois
(Received March 6, 1953)

A general method, suitable for fast computing machines, for investigating such properties as equations of state for substances consisting of interacting individual molecules is described. The method consists of a modified Monte Carlo integration over configuration space. Results for the two-dimensional rigid-sphere system have been obtained on the Los Alamos MANIAC and are presented here. These results are compared to the free volume equation of state and to a four-term virial coefficient expansion.

I. INTRODUCTION

The purpose of this paper is to describe a general method, suitable for fast electronic computing machines, of calculating the properties of any substance which may be considered as composed of interacting individual molecules. Classical statistics is assumed,

II. THE GENERAL METHOD FOR AN ARBITRARY POTENTIAL BETWEEN THE PARTICLES

In order to reduce the problem to a feasible size for numerical work, we can, of course, consider only a finite number of particles. This number \(N\) may be as high as several hundred. Our system consists of a square† con-
The purpose of this paper is to describe a general method, suitable for fast electronic computing machines, of calculating the properties of any substance which may be considered as composed of interacting individual molecules. Classical statistics is assumed,

I. INTRODUCTION

THE purpose of this paper is to describe a general method, suitable for fast electronic computing machines, of calculating the properties of any substance which may be considered as composed of interacting individual molecules. Classical statistics is assumed,

II. THE GENERAL METHOD FOR AN ARBITRARY POTENTIAL BETWEEN THE PARTICLES

In order to reduce the problem to a feasible size for numerical work, we can, of course, consider only a finite number of particles. This number $N$ may be as high as several hundred. Our system consists of a square\(^{+}\) con-

• They knew this was important… and they were right!
50 years later at a Los Alamos meeting:

- Only M. Rosenbluth attended, although with terminal cancer.
- Metropolis mainly contributed CPU time on MANIAC.
- von Neumann and Ulam invented the Monte Carlo method in 1946 and pointed out that it could be used for simulations.
- Teller: Statistical averages can be made as ensemble averages.
- Interesting author list: two couples. How often does this happen?

Why Los Alamos?

- The US was building the atomic bomb. At least one good thing came out.
Reminder: simple Monte Carlo

- In statistical mechanics we want to compute the average of an observable $O$:
  \[
  \langle O \rangle = \sum_n P_n^{eq} O_n
  \]
  \[
  P_n^{eq} = \frac{e^{\beta E_n}}{\sum_n e^{-\beta E_n}}
  \]

- **Problem:** The number of states is exponentially large.

- **Solution:** Statistically sample a few (smartly chosen) states to obtain an estimate of $O$ but with a statistical error.
  
  - If we chose the states according to $P_n^{eq}$ and ensure detailed balance we obtain a Markov chain for $\langle O \rangle_{est}$
    
    \[
    \langle O \rangle_{est} = \frac{1}{M} \sum_i^M O_i
    \]
    
    where $M$ is the number of states sampled.

- **Metropolis algorithm:** accept a new configuration if \(e^{-\Delta E/T} > \text{rand}()\)
  
  \[
  P_{\text{accept}} = \min(1, e^{-\Delta E/T})
  \]
Why does simple Monte Carlo fail here?

- The systems we are interested in have rugged energy landscapes.
- At low temperature, when $\Delta E$ is large
  
  $$P_{\text{accept}} = \min(1, e^{-\Delta E/T})$$

  is “never” accepted.

- How can we resolve the problem?
  - Tunnel through barrier.
  - Heat up the system to overcome the barrier.

- Note:
  - Simulated annealing has similar problems (stuck in valleys).
Why does simple Monte Carlo fail here?

- The systems we are interested in have rugged energy landscapes.
- At low temperature, when \( \Delta E \) is large
  \[
  P_{\text{accept}} = \min(1, e^{-\Delta E/T})
  \]
  is “never” accepted.

- How can we resolve the problem?
  - Tunnel through barrier.
  - Heat up the system to overcome the barrier.

- Note:
  - Simulated annealing has similar problems (stuck in valleys).
Why does simple Monte Carlo fail here?

- The systems we are interested in have rugged energy landscapes.
- At low temperature, when $\Delta E$ is large,
  \[ P_{\text{accept}} = \min(1, e^{-\Delta E/T}) \]
  is “never” accepted.

How can we resolve the problem?
- Tunnel through barrier.
- Heat up the system to overcome the barrier.

Note:
- Simulated annealing has similar problems (stuck in valleys).
Exchange (parallel tempering) Monte Carlo
Top 10 reasons to use exchange MC:

Hukushima & Nemoto (96)
Top 10 reasons to use exchange MC:

1. Very efficient.

Hukushima & Nemoto (96)
Top 10 reasons to use exchange MC:

1. Very efficient.
2. Simple to implement.

Hukushima & Nemoto (96)
Top 10 reasons to use exchange MC:

1. Very efficient.
2. Simple to implement.
3. Only few parameters.
Top 10 reasons to use exchange MC:

1. Very efficient.
2. Simple to implement.
3. Only few parameters.
4. It is practical (several T’s).

Hukushima & Nemoto (96)
Top 10 reasons to use exchange MC:

1. Very efficient.
2. Simple to implement.
3. Only few parameters.
4. It is practical (several $T$'s).
5. Small numerical overhead.

Hukushima & Nemoto (96)
Top 10 reasons to use exchange MC:

1. Very efficient.
2. Simple to implement.
3. Only few parameters.
4. It is practical (several T’s).
5. Small numerical overhead.
6. It is easy to parallelize.

Hukushima & Nemoto (96)
Top 10 reasons to use exchange MC:

1. Very efficient.
2. Simple to implement.
3. Only few parameters.
4. It is practical (several $T$'s).
5. Small numerical overhead.
6. It is easy to parallelize.
7. Wide application range.

Hukushima & Nemoto (96)
Top 10 reasons to use exchange MC:

1. Very efficient.
2. Simple to implement.
3. Only few parameters.
4. It is practical (several $T$'s).
5. Small numerical overhead.
6. It is easy to parallelize.
7. Wide application range.
8. Mix with other algorithms.

Hukushima & Nemoto (96)
Top 10 reasons to use exchange MC:

1. Very efficient.
2. Simple to implement.
3. Only few parameters.
4. It is practical (several \( T \)'s).
5. Small numerical overhead.
6. It is easy to parallelize.
7. Wide application range.
8. Mix with other algorithms.
9. It is ‘Made in Japan.’
Top 10 reasons to use exchange MC:

1. Very efficient.
2. Simple to implement.
3. Only few parameters.
4. It is practical (several $T$s).
5. Small numerical overhead.
6. It is easy to parallelize.
7. Wide application range.
8. Mix with other algorithms.
9. It is ‘Made in Japan.’
10. I told you so...

Hukushima & Nemoto (96)
Exchange (parallel tempering) Monte Carlo

**Idea:**
- Simulate $M$ copies of the system at different temperatures with $T_{\text{max}} > T_c$ (typically $T_{\text{max}} \sim 2T_c^{\text{MF}}$).
- Allow swapping of neighboring temperatures: easy crossing of barriers.

**What has to be tuned?**
- Number of temperatures.
- Position of temperatures.

Hukushima & Nemoto (96)
Parallel tempering: algorithm and details

• Brief outline of the algorithm:
  • Perform a Monte Carlo update between *neighboring* replicas.
  • Best to keep temperatures and change pointers to configurations.

\[
P(S_{m+1} \leftrightarrow S_m, \beta_{m+1} \leftrightarrow \beta_m) = e^{-\Delta} : \Delta > 0
\]
\[
P(S_{m+1} \leftrightarrow S_m, \beta_{m+1} \leftrightarrow \beta_m) = 1 : \Delta \leq 0
\]
\[
\Delta = (\beta_{m+1} - \beta_m)(E_m - E_{m+1})
\]
\[
\beta = 1/T
\]  
[obeys detailed balance]

• How often do we call the swap routine?
  • Optimal ratio of lattice sweeps (*N* spin updates) and swaps is 1:1, i.e., after each lattice sweep, perform an attempted parallel tempering swap [see Katzgraber et al., JSTAT P03018 (2006)].
How fast is fast? Example: 3D spin glass

- Equilibration times:
  \[ \tau_{eq}^{PT} \approx 300 \text{ MCS} \]
  \[ \tau_{eq}^{SM} \approx 10^6 \text{ MCS} \]

- Equilibration test (later):
  \[ q_l(E) = \frac{2T|E|}{z} + 1 \]
  \[ z = 2D \]

Once both agree, the system is in equilibrium (only Gaussian disorder).

Katzgraber et al. PRB (01)
Equilibration times:
\[ \tau_{\text{eq}}^{\text{PT}} \approx 300 \text{ MCS} \]
\[ \tau_{\text{eq}}^{\text{SM}} \approx 10^6 \text{ MCS} \]

Equilibration test (later):
\[ q_l(E) = \frac{2T|E|}{z} + 1 \]
\[ z = 2D \]

Once both agree, the system is in equilibrium (only Gaussian disorder).

Katzgraber et al. PRB (01)
How fast is fast? Example: 3D spin glass

- Equilibration times:
  \[ \tau_{eq}^{PT} \approx 300 \text{ MCS} \]
  \[ \tau_{eq}^{SM} \approx 10^6 \text{ MCS} \]

- Equilibration test (later):
  \[ q_l(E) = \frac{2T|E|}{z} + 1 \]
  \[ z = 2D \]

Once both agree, the system is in equilibrium (only Gaussian disorder).

Katzgraber et al. PRB (01)
How do we choose the temperatures?
How many temperatures do we need?

- **Two possible scenarios:**
  - Temperatures too far apart: parallel simple Monte Carlo chains.
  - Temperatures too close: overhead.

- **What determines the number $M$ of temperatures?**
  - The energy distributions of the system at $T_1$ and $T_2$ have to overlap.
  - Because $\Delta E \sim C_V \quad \rightarrow \quad M \sim \sqrt{N^{1+\alpha/d\nu}}$
  - Note: Systems for which $C_V|_{T \to 0} \rightarrow 0$ require many temperatures.

- In principle, we need as many temperatures such that the method works. Measure? Acceptance rates.
Measuring acceptance rates

- **Definition:**
  \[ A = \frac{N_{\text{accept}}}{N_{\text{trial}}} \]

- **Traditional wisdom:** Tune the temperature set such that...
  - \( 0.2 \leq A \leq 0.9. \)
  - \( A \) is approximately independent of temperature.

- **Detailed implementation which gives flat acceptance rates:** Incomplete beta function law [uses \( A = f(C_V) \)].

- **Notes:**
  - A quick run (no need to equilibrate) will immediately produce stable acceptance rates (easy tuning by hand).
  - It has been claimed that \( A \approx 0.3 \) is optimal.

*Predescu et al., JSTAT (03)*

*Rathore et al., J. Chem. Phys. (05)*
Practical/traditional approaches

- Geometric progression:
  - Works well when $C_V \sim \text{const}$ (like for spin glasses).
  - Iteratively construct a temperature set and tune $M$ with $\lambda$.
    \[
    \frac{1}{T_i} = \lambda R^{i-1} \frac{1}{T_{\text{min}}} \quad R = \left[ \frac{T_{\text{min}}}{T_{\text{max}}} \right]^{1/(M-1)}
    \]

- By hand:
  - If $C_V$ diverges (phase transition) start from a geometric progression.
  - Interlace extra temperatures by hand.
  - Tedious, but after a while you get a feeling for it.

- But... are temperature-independent acceptance rates optimal?
  - Replicas do a random walk in temperature space.
Optimizing the ensemble \( \{T_i\} \)

- Track random walk of the replicas in temperature space...
  
  \[ T_{\text{max}} \]
  
  \[ T_{\text{min}} \]

- Goal:
  - Minimize the round-trip time \( t_{\text{rt}} \) and ensure that \( t_{\text{up}} \sim t_{\text{dn}} \).
  - This shall ensure an efficient sampling of temperature space.

- How?
  - Tune the ensemble \( \{T_i\} \) at a fixed number of temperatures \( M \).
  - Quick run at the beginning of the simulation.
Conventional temperature sets...

- Start from a temperature set with $A(T) \sim \text{const.}$
- Track one replica and measure the local diffusivity $D$ in the ensemble $\{T_i\}$.

- Ising model:
  - Bottleneck at $T_c$!

- Fully-frustrated Ising model:
  - Bottleneck at $T = 0$.

- Goal: change $\{T_i\}$ so that $D \sim \text{const.}$ for each $\Delta T_i = T_{i+1} - T_i$. 

Conventional temperature sets...

- Start from a temperature set with $A(T) \sim \text{const.}$.
- Track one replica and measure the local diffusivity $D$ in the ensemble $\{T_i\}$.

- Ising model:
  - Bottleneck at $T_c$!

- Fully-ferustrated Ising model:
  - Bottleneck at $T = 0$.

- Goal: change $\{T_i\}$ so that $D \sim \text{const.}$ for each $\Delta T_i = T_{i+1} - T_i$. 

![Graph showing the diffusivity $D$ as a function of temperature $T$ for different $L$ values (6, 10, 20, 30, 40). The graph highlights the bottleneck at $T_c$.](image-url)
Conventional temperature sets...

- Start from a temperature set with $A(T) \sim \text{const.}$.
- Track one replica and measure the local diffusivity $D$ in the ensemble $\{T_i\}$.

- Ising model:
  - Bottleneck at $T_c$!

- Fully-frustrated Ising model:
  - Bottleneck at $T = 0$.

- Goal: change $\{T_i\}$ so that $D \sim \text{const.}$ for each $\Delta T_i = T_{i+1} - T_i$. 

![Graph showing diffusivity $D$ vs. temperature $T$ for different $L$ values in 2d FFIM.](image)
Conventional temperature sets...

- Start from a temperature set with $A(T) \sim \text{const}$.
- Track one replica and measure the local diffusivity $D$ in the ensemble $\{T_i\}$.

- Ising model:
  - Bottleneck at $T_c$!

- Fully-frustrated Ising model:
  - Bottleneck at $T = 0$.

- Goal: change $\{T_i\}$ so that $D \sim \text{const.}$ for each $\Delta T_i = T_{i+1} - T_i$. 
Feedback method

Katzgraber et al., JSTAT (06)

- Track replicas for $N_t$ MCS and compute a histogram $n(T)$ of the number of times a replica hits a given $T$.
- Compute a histogram of directed walkers $n_{up}(T)$.
- Calculate the fraction $f(T) = n_{up}(T')/n(T')$ [\(D(T) = (df/dT)^{-1}\)].
- Calculate \(\{T'_i\}\) from \(\{T_i\}\):
  \[
  \frac{1}{\Delta T'_i} = C \sqrt{\frac{1}{\Delta T_i} \frac{df}{dT_i}} \sim \frac{1}{\sqrt{\Delta T_i D(T_i)}}
  \]
- Choose $C$ such that \([T'_{\text{min}}, T'_{\text{max}}]\) maps back to \([T_{\text{min}}, T_{\text{max}}]\). Iterate!
Feedback method

- Track replicas for $N_t$ MCS and compute a histogram $n(T)$ of the number of times a replica hits a given $T$.
- Compute a histogram of directed walkers $n_{up}(T)$.
- Calculate the fraction $f(T) = n_{up}(T)/n(T)$ [$D(T) = (df/dT)^{-1}$].
- Calculate $\{T'_i\}$ from $\{T_i\}$:
  \[
  \frac{1}{\Delta T'_i} = C \sqrt{\frac{1}{\Delta T_i}} \frac{df}{dT_i} \sim \frac{1}{\sqrt{\Delta T_i D(T_i)}}
  \]
- Choose $C$ such that $[T'_{\text{min}}, T'_{\text{max}}]$ maps back to $[T_{\text{min}}, T_{\text{max}}]$. Iterate!

Katzgraber et al., JSTAT (06)
Feedback method

Katzgraber et al., JSTAT (06)

- Track replicas for $N_t$ MCS and compute a histogram $n(T)$ of the number of times a replica hits a given $T$.
- Compute a histogram of directed walkers $n_{up}(T)$.
- Calculate the fraction $f(T) = n_{up}(T)/n(T)$  \[ D(T) = (df/dT)^{-1} \].
- Calculate $\{T'_i\}$ from $\{T_i\}$:
  \[
  \frac{1}{\Delta T'_i} = C \sqrt{\frac{1}{\Delta T_i} \frac{df}{dT_i}} \sim \frac{1}{\sqrt{\Delta T_i D(T_i)}}
  \]
- Choose $C$ such that $[T'_{\text{min}}, T'_{\text{max}}]$ maps back to $[T_{\text{min}}, T_{\text{max}}]$. Iterate!
Feedback method: Ising model

- Start from a geometric progression temperature set (not good!).
- Feedback maximizes the **rate of round trips**.
- Feedback reallocates resources where needed (critical point): **Acceptance rates not constant**.
Start from a geometric progression temperature set (not good!).

Feedback maximizes the rate of round trips.

Feedback reallocates resources where needed (critical point):
Acceptance rates not constant.
Feedback method: Ising model

- Start from a geometric progression temperature set (not good!).
- Feedback maximizes the rate of round trips.
- Feedback reallocates resources where needed (critical point): Acceptance rates not constant.
Evolution of the temperature set

- Data for the Ising model.
- After few iterations the temperature set converges.
- The method reallocates more temperatures to the bottlenecks.
Evolution of the temperature set

- Data for the Ising model.
- After few iterations the temperature set converges.
- The method reallocates more temperatures to the bottlenecks.
Feedback method: FFIM

- Start from a geometric progression temperature set (not good!).
- Feedback maximizes the rate of round trips.
- Feedback reallocates resources where needed (ground state): Acceptance rates not constant.
Feedback method: FFIM

- Start from a geometric progression temperature set (not good!).
- Feedback maximizes the rate of round trips.
- Feedback reallocates resources where needed (ground state): Acceptance rates not constant.
Feedback method: FFIM

- Start from a geometric progression temperature set (not good!).
- Feedback maximizes the rate of round trips.
- Feedback reallocates resources where needed (ground state):
  Acceptance rates not constant.
Comparison of fractions

In general, a “flat” \([A(T) \sim \text{const}]\) temperature set is not too bad, but not optimal.

- If \(C_v\) diverges, a geometrical progression is bad.
What happens for systems with disorder?

- **Problems:**
  - We need to deal with averages.
  - Each fraction depends on the given disorder configuration.
  - Each disorder configurations converges independently...
  - The round-trip times are fat tail distributed.
  - Each disorder realization has its own temperature set.

- **Solution:**
  - Fortunately $C_v \sim \text{const.}$ and a “flat” temperature set is close to optimal “on average.”
What happens for systems with disorder?

**Problems:**
- We need to deal with averages.
- Each fraction depends on the given disorder configuration.
- Each disorder configurations converges independently...
- The round-trip times are fat tail distributed.
- Each disorder realization has its own temperature set.

**Solution:**
- Fortunately $C_v \sim \text{const.}$ and a “flat” temperature set is close to optimal “on average.”
What happens for systems with disorder?

- Problems:
  - We need to deal with averages.
  - Each fraction depends on the given disorder configuration.
  - Each disorder configuration converges independently...
  - The round-trip times are fat tail distributed.
  - Each disorder realization has its own temperature set.

- Solution:
  - Fortunately $C_v \sim \text{const.}$ and a “flat” temperature set is close to optimal “on average.”

3D spin glass

Do not use for glasses...
Simulations in (bio)chemistry

- Example: Protein A

- It can happen that the replicas will only move in subspaces of the phase space (A, B, C).
- Feedback optimization helps overcome these problems easily. Simulations otherwise impossible.
Simulations in (bio)chemistry

- **Example:** Protein A

- It can happen that the replicas will only move in subspaces of the phase space (A, B, C).

- Feedback optimization helps overcome these problems easily. Simulations otherwise impossible.
Simulations in (bio)chemistry

- Example: Protein A

- It can happen that the replicas will only move in subspaces of the phase space (A, B, C).

- Feedback optimization helps overcome these problems easily. Simulations otherwise impossible.

Use for biology applications...
Some final thoughts (and extensions)...
Heuristic ground state search

- **Outline of the algorithm:**
  - Perform an exchange MC run with $T_{\text{min}}$ close to zero.
  - Simulate two copies of the system with different Markov chains.
  - The system is in thermal equilibrium:
    - after each lattice sweep record the spin configuration and energy for the lowest $T$ if these match in both copies.
    - If a lower energy is found, replace the previously recorded energy.
  - Measure for $1/4t_{\text{eq}}$.

- For small to intermediate system sizes where other algorithms do not work well the method provides heuristic ground states with high probability (98% up).
Extensions and combinations

- Any control variable can be used:
  - Field
  - Temperature and field
  - Coupling constants in QCD
  - Frequencies (e.g., in a Holstein model)
  - ...

- Combinations with other algorithms possible:
  - Exchange Monte Carlo molecular dynamics (biomolecules).
  - Exchange quantum Monte Carlo (quantum spin glasses).
  - Bayesian periodogram (planet search in star systems).
  - Iterative search methods (combinatorial problems).
  - Cluster exchange Monte Carlo (diluted spin glasses).
  - ...
Thank you.