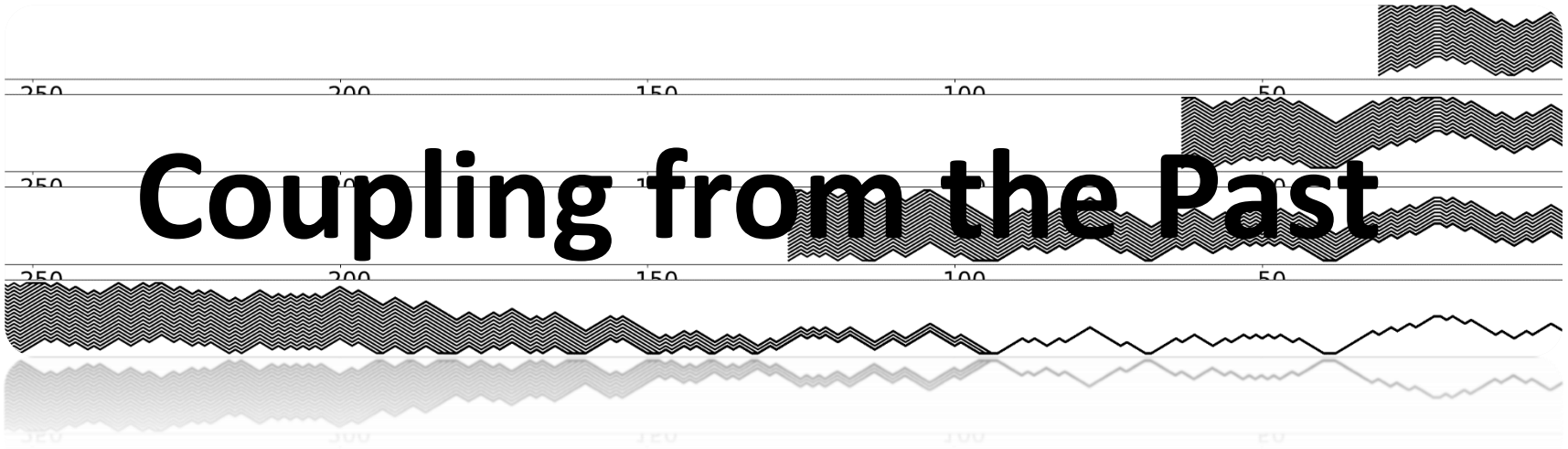


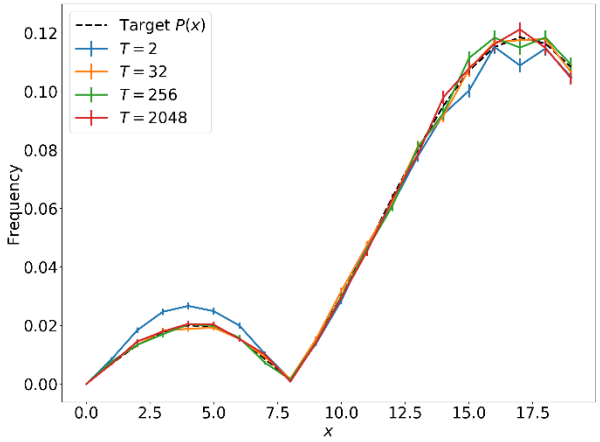
Coupling from the Past

Algorithm Interest Group
presentation by Eli Chertkov

What does CFTP do?

Generates *perfect* samples from a Markov chain!

Markov chains are often used to sample desired probability distributions, say $P(x)$.



The samples $x_1, x_2, \ldots$ generated by a Markov chain are correlated in time. If we draw samples every T time steps, then we are not sampling $P(x)$ exactly. If T is large enough, larger than what is called the **mixing time**, then the Markov chain has "forgotten its initial conditions" and the samples are uncorrelated and drawn from the correct distribution.

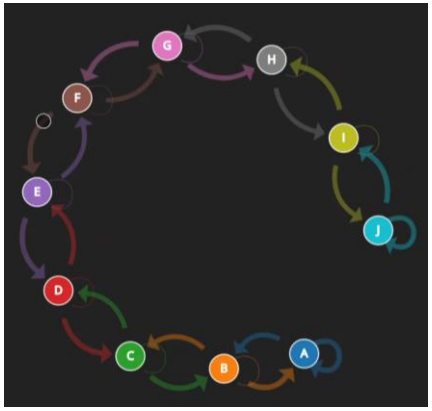
Also called:

- Perfect Sampling, Exact Sampling, or Perfect Simulation
- Wilson-Propp algorithm

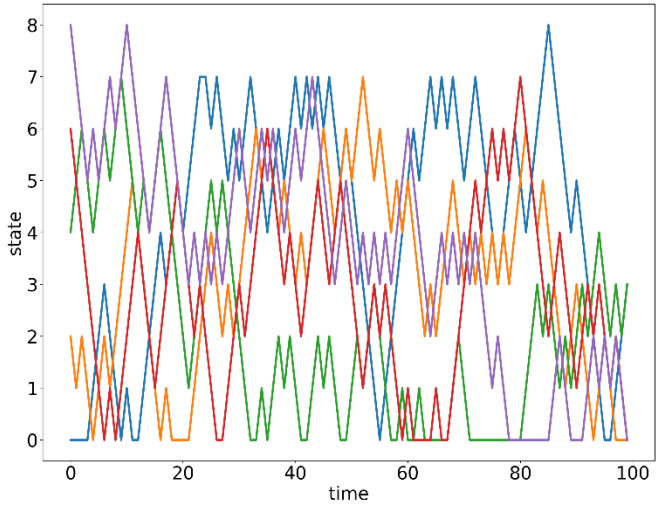
Markov chain sampling

A Markov chain is a memoryless random process. At each time step, a state jumps randomly to another state in a way that depends only on the value of the current state.

Toy model Markov chain



$N = 10$ states: $x = A, B, C, \ldots, I$
Arrows indicate the transition probability $P(x, y)$



Examples of a few random Markov chains evolving over time.

Source: setosa.io/markov

Markov chain sampling (cont)

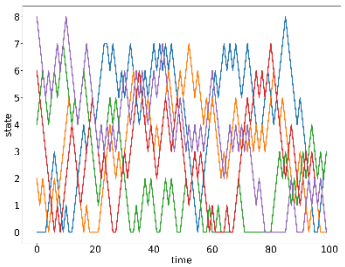
You can think of a Markov chain's time evolution as being governed by applying a random function at each time step

$x_t = f(x_{t-1}, r_t)$ where r_t is a set of random numbers

$Prob[y = f(x, r)] = P(x, y) =$ Markov chain transition probability from x to y

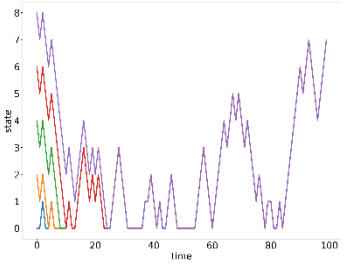
Toy model Markov chain

$$f(x,r) = \begin{cases} \min(x+1, N) & when\ r < 0.45 \\ x & when\ r < 0.46 \\ \max(x-1, 1) & otherwise \end{cases}$$



Each Markov chain has its own random numbers.

Coalescence:
After some time, all of the states converge to a single state.

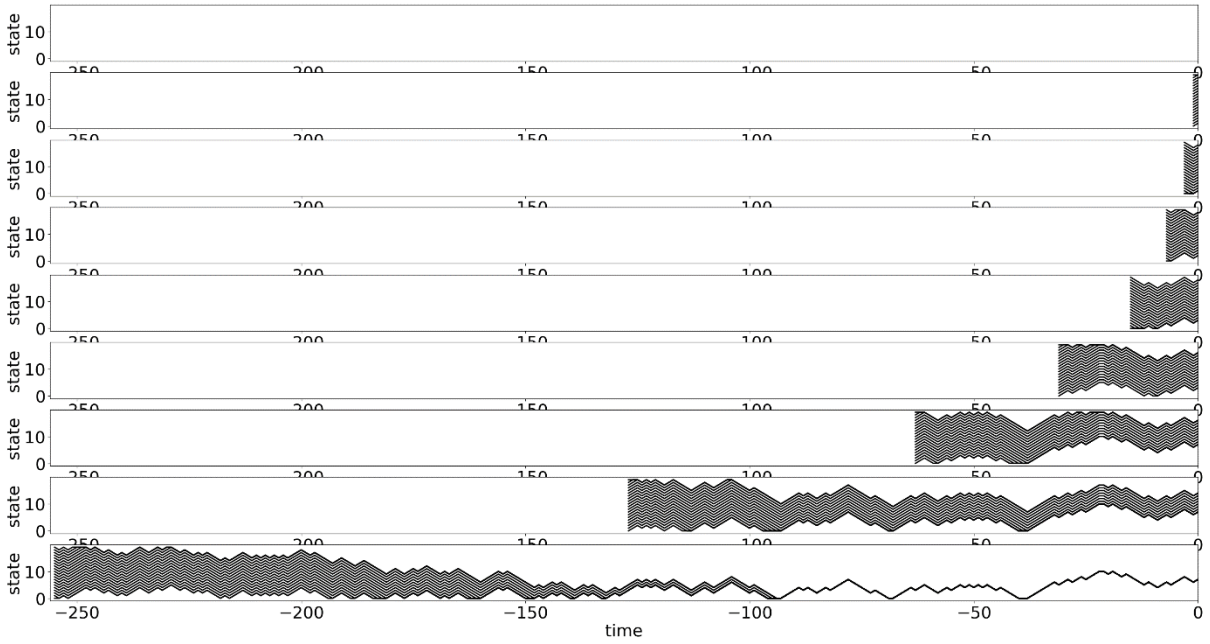


Each Markov chain has the same random numbers.

Results on toy model

Main idea of CFTP:

go back in time until coalescence occurs

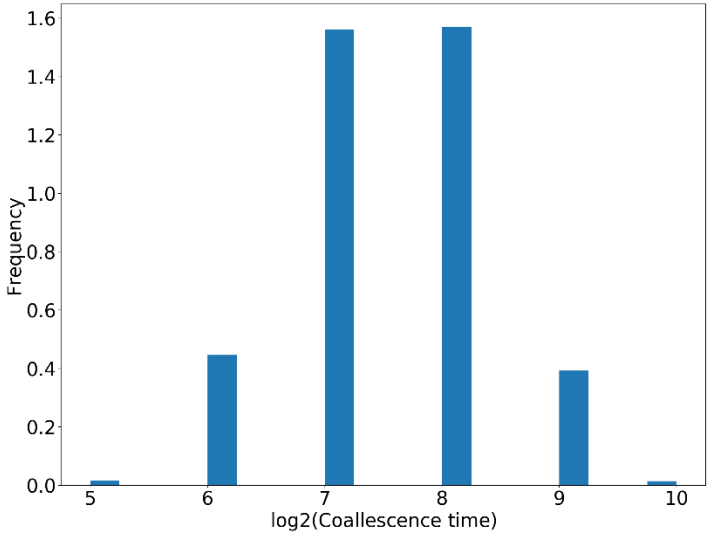


This is a perfect sample

To determine the coalescence time, you restart at times $-T = -1, -2, -4, -8, -16, \ldots$ and run the Markov chain forward from $t = -T$ to $t = 0$ until you observe the Markov chains converge to a unique state.

Results on toy model

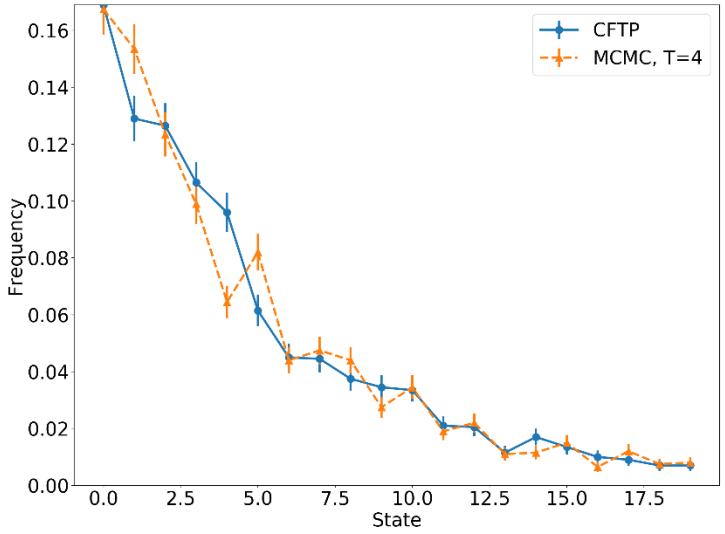
Coalescence times

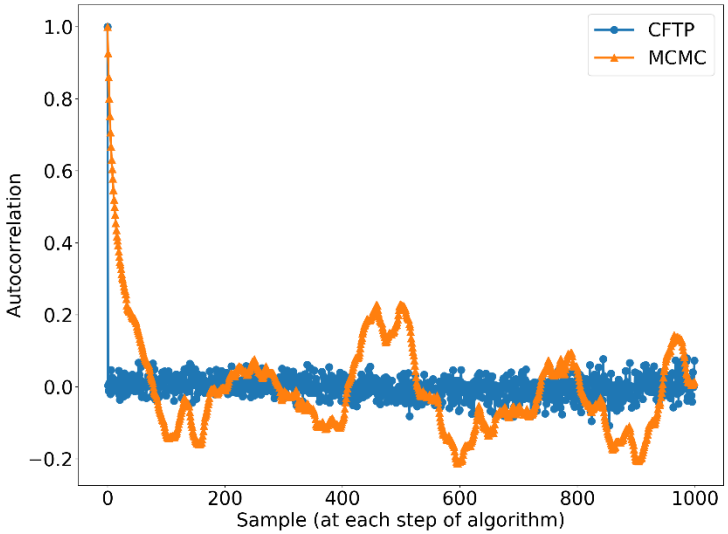


The runtime of the algorithm is variable. You have to wait until you find a perfect sample. Generally, you expect the coalescence time to be $O(4M)$, where M is the **mixing time** of the Markov chain. Altogether, the algorithm runtime is then $O(4MN)$ where N is the number of states.

Results on toy model

Comparison to usual MCMC-style sampling



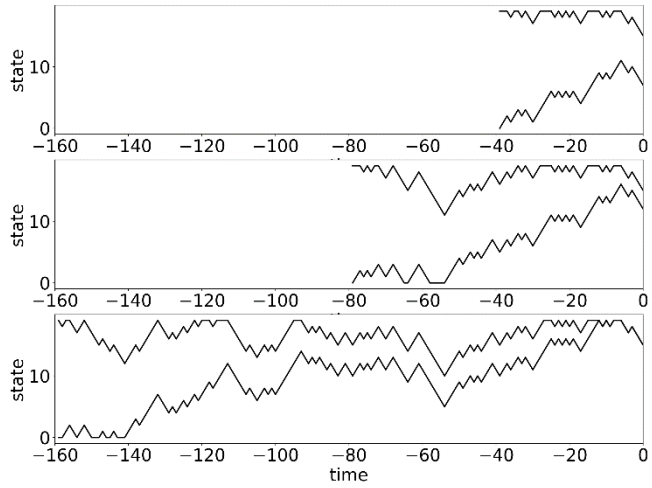


Samples drawn from the Markov chain every T steps are correlated, while the CFTP samples are completely uncorrelated.

Results on toy model

Monotonic CFTP

Partial order defined on states: $x \leq y$



Monotonic coupling $f(x, r)$

$$x \leq y \Rightarrow f(x, r) \leq f(y, r)\ \forall r$$

If your **states** can be **partially ordered** and your **couplings are monotonic**, then the algorithm can be made MUCH more efficient. Instead of checking that *all* states coalesce, you can just check whether the "top" and "bottom" states coalesce. In this case, the run time is $O(4Mh)$ where h is the "height", or the longest distance between the top and bottom states.

MCMC on 2D Ising model

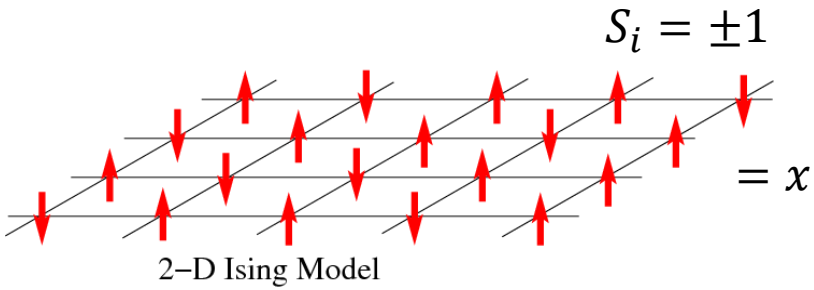
An important application of Markov chains is for doing statistical physics.

The Ising model is a simplified model of a ferromagnet made of up and down spins:

Energy: $E(x) = -J \sum_{\langle i,j \rangle} S_i S_j$

Partition function: $Z = \sum_x e^{-\beta E(x)}$

Boltzmann dist.: $P(x) = \frac{1}{Z} e^{-\beta E(x)}$



You can sample a MC to compute thermal averages of observables:

$$\langle O \rangle = \sum_x O(x)P(x) \approx \frac{1}{N_s} \sum_{i=1}^{N_s} O(x_i)$$

Gibbs sampling (heat-bath algorithm) MC

For each time t, go from state x_{t-1} to a new state x_t by

Picking a random spin i *using random number r_1*

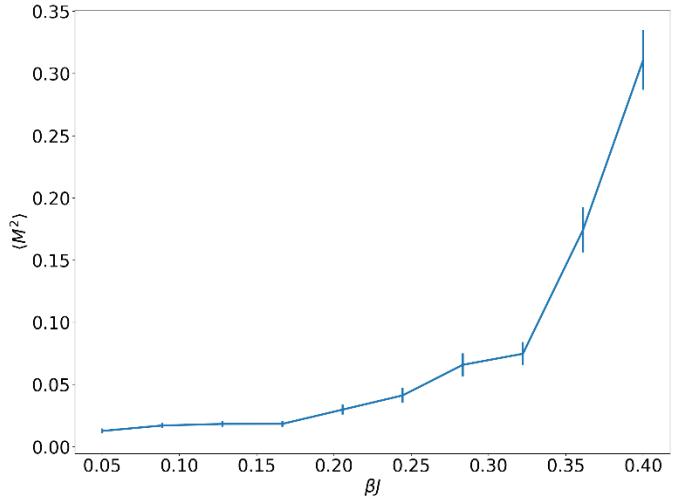
Setting it to $S_i = +1$ with prob. $P_+ = \frac{\lambda}{\lambda + \lambda^{-1}}$, where $\lambda = e^{\beta J \sum_{j \in N(i)} S_j}$ *using random number r_2*

Setting it to $S_i = -1$ otherwise

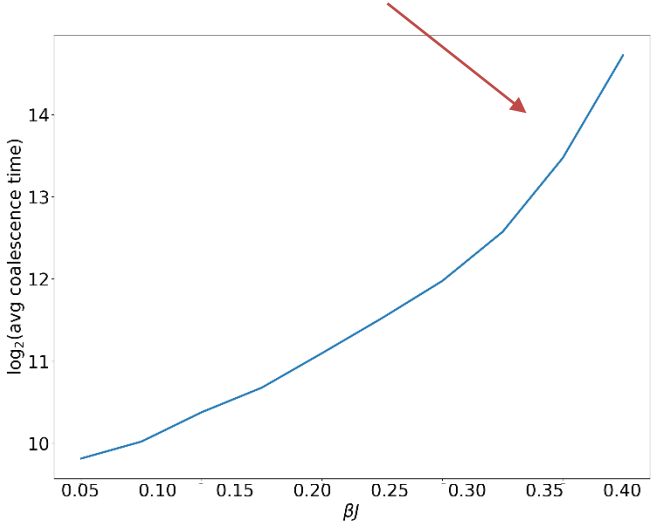
} Defines an $f(x, r)$ that is **monotonic**

CFTP MCMC on 2D Ising

Ferromagnetic Ising model on 10×10 square lattice



Critically slows down near the phase transition at $\beta J \approx 0.45$

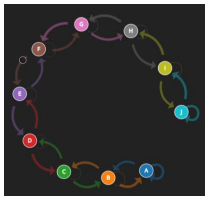


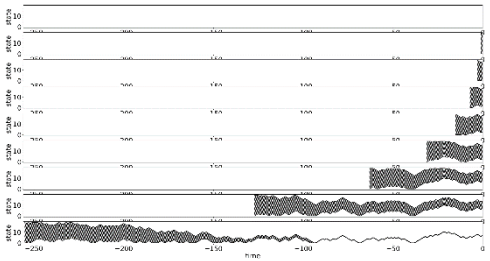
The lesson here is that CFTP doesn't save you if you design a poorly mixing Markov chain.

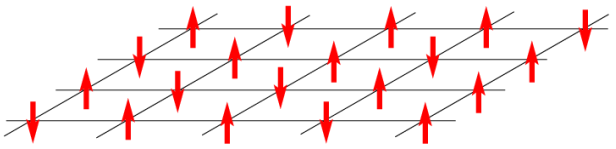
However, if you do have a well-mixing monotonic coupling MC, then you can basically run CFTP for no extra cost.

Summary

- CFTP is a method for generating perfect samples from Markov chains.
- Pros:
  - provides perfect samples
  - provides estimate of mixing time
  - can be applied to interesting problems, like Ising model
- Cons:
  - efficient implementations are limited in applicability (monotonic couplings and extensions)
  - slow when mixing time is large







References

- Alistair Sinclair, CS294: Markov Chain Monte Carlo: Foundations & Applications, Fall 2009 lecture notes. https://people.eecs.berkeley.edu/~sinclair/cs294/n9.pdf
- *Information Theory, Inference, and Learning Algorithms* by David MacKay.