Critical Slowing Down and Cluster Updates in Monte Carlo Simulations

Ruben Andrist

Tutor: Dr. Munehisa Matsumoto

May 14th 2007

Motivation

- Existing Theory: Meanfield Theory
- Exact solutions by Onsager (Ising 1D & 2D)
- However: Comparison shows inaccuracy
- Need accurate predictions for unsolved problems

What are Monte Carlo Simulations?

- Numerical approach to make predictions
- Simulate stochastic processes using random numbers
- Find equilibrium state of the system
- Advantages and Disadvantages over physical experiments

Idea behind Monte Carlo Simulations

- Construct Markov Chain of states
- Next state generated from current one
- Ensure tendency towards equilibrium
- Relaxation to equilibrium after time

Historical Notes on Monte Carlo Simulations

- Laplace: Estimation of π using Needles
- N. Metropolis: Review of Monte Carlo simulations (1949)
- Has grown faster than the computer industry
- Simulations have become another way of research

Ising Model in two Dimensions

- Each spin has only two states: $\sigma \in \{\pm 1\}$
- Consider only nearest neighbour interaction
- Hamiltonian given by:

$$\mathcal{H} = -\mathcal{J} \sum_{NN} \sigma_i \sigma_j - H \sum_i \sigma_i$$

$\mathcal{J}$ – interaction constant, H – magnetic field.

Ising Model in two Dimensions

- Theory predicts $T_c = 2.269$
- Correlation length scales as $\xi \propto \varepsilon^{-\nu}$
  ξ – Correlation length, ν – Critical exponent,
  $\varepsilon = |1 - T/T_c|$ – Reduced temperature
- Note also, that the correlation time scales with $\tau \propto \xi^z$
  (z – dynamic critical exponent)

How to design a Monte Carlo algorithm?

- Must converge (to find equilibrium)
- Feasible with available technology
- Yield data within reasonable time

Conditions for Validity

- Ergodicity (condition for convergence)

  "Every point in phase space reachable"

- Detailed Balance relation

$$P_n^{eq} W_{n \to m} = P_m^{eq} W_{m \to n}$$

P_n – Probability to be in State n
$W_{n \to m}$ – Probability to get from State n to State m

How to Check for Correlation of the Data?

- Markov Chains $\Rightarrow$ Data is autocorrelated
- Do a Binning Analysis to check this
- Divide data into bins of increasing size
- While correlated, expect logarithmic increase

Local Updates

- Probability determined only by nearest neighbours
- Sites can be updated locally, one by one
- Several such updates construct a Markov Chain

Metropolis Algorithm

- N. Metropolis proposed the condition:

$$\text{if } (r < e^{-\Delta E/k_B T}) \text{ then flip}$$

$\Delta E = E_{after} - E_{before}$, r – random $\in [0, 1)$
k_B – Boltzmann constant, T – Temperature

- Note: if $\Delta E < 0$, the spin is always flipped

Ergodicity & Detailed Balance

- Ergodicity: choose sequence of random numbers
- Detailed Balance (sufficient but not necessary)

Visual Demonstration

- Dynamics at different temperatures
- Introducing a magnetic field
- First order phase transition when sweeping H
- Various cluster sizes at T_c

Measurements

- Measured property: Magnetisation per site

$$M_n = \frac{1}{\#Sites} \sum_i \sigma_i$$

- Calculated and averaged over time

$$\langle M \rangle = \frac{1}{\#States} \sum_n M_n$$

- Several runs with different random numbers averaged

Validity of the Data

- Measurements depend on initial state
- Allow time for relaxation to equilibrium
- Omit first $\mathcal{N}_0$ configurations

Critical Slowing Down

- Relaxation time depends on T
- At a certain T_c, it diverges
- Physical effect, not just seen in simulations

Why does Critical Slowing Down occur?

- Divergence associated with a long correlation length
- Local updates become very ineffective
- Shown by measuring M/t at different T

Relaxation from High Energy State

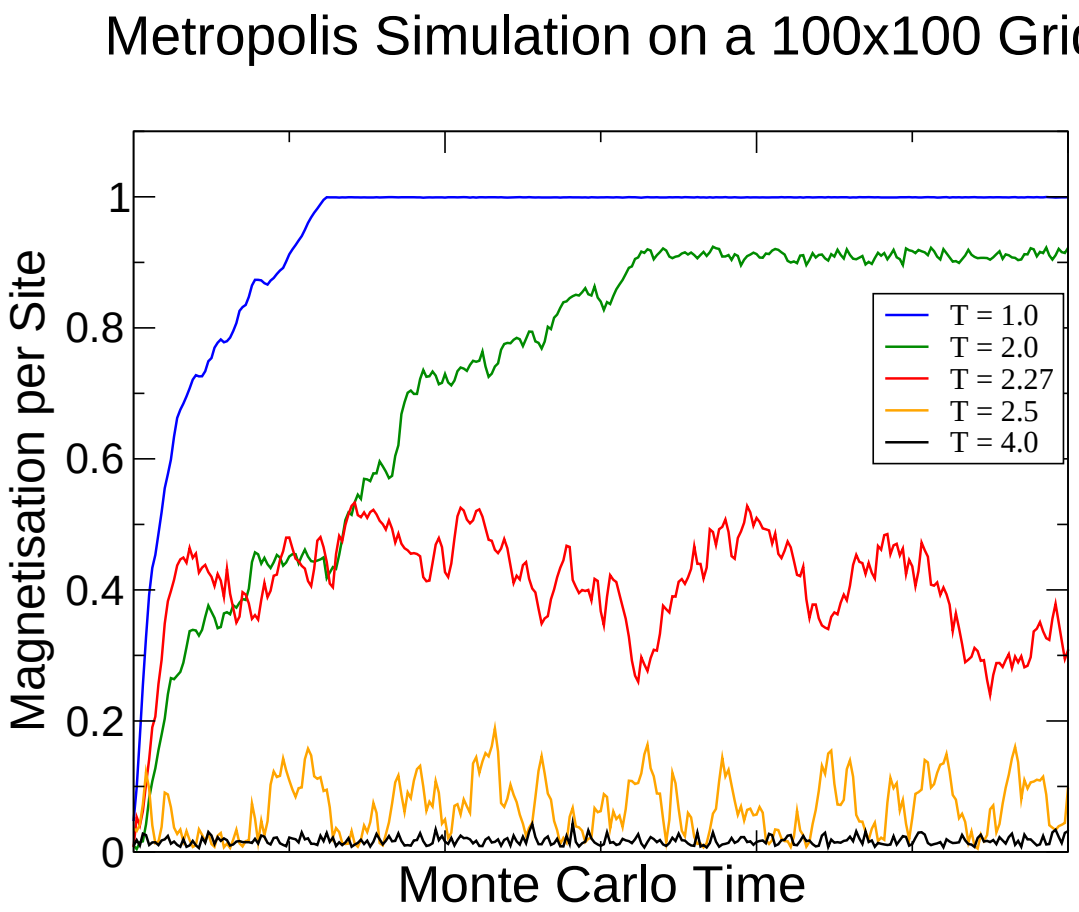

Figure: Relaxation from High Energy State under Metropolis Dynamics

Relaxation from Low Energy State

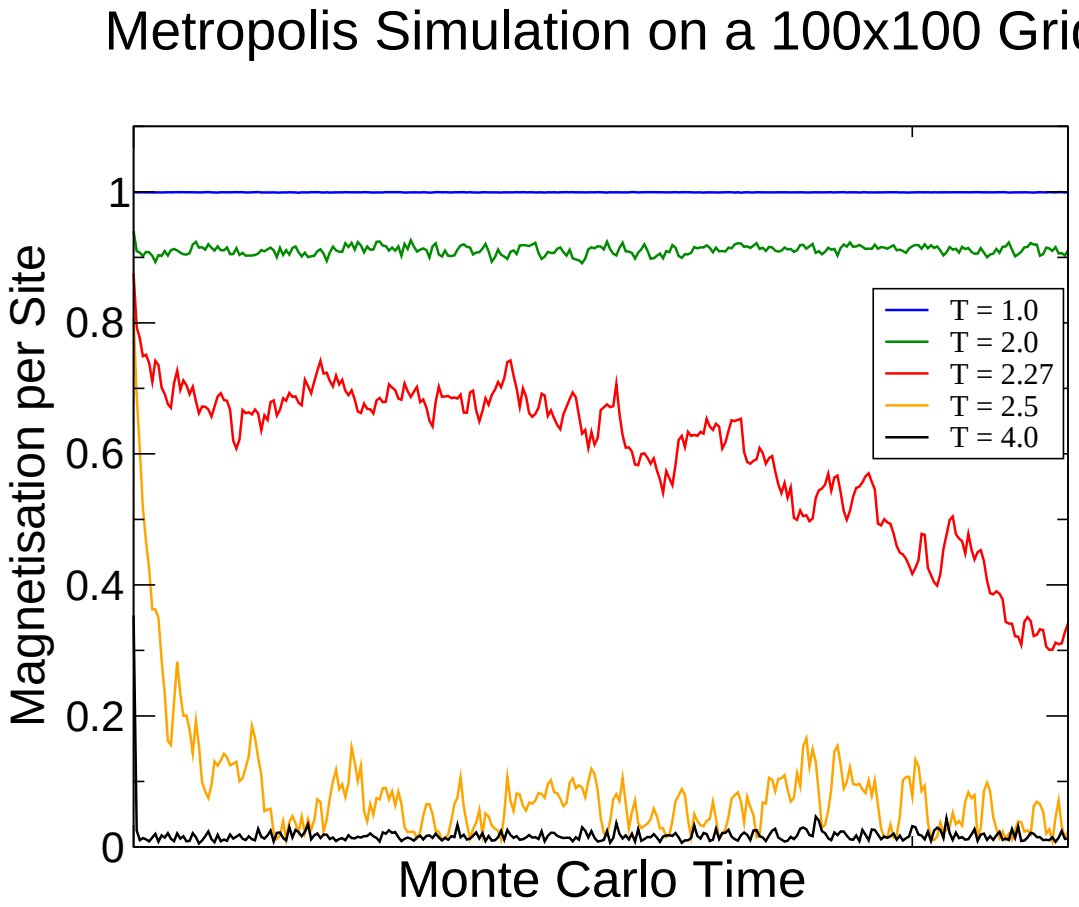

Figure: Relaxation from Low Energy State under Metropolis Dynamics

Consequences of Critical Slowing Down

- Very long relaxation time
- Most of the gathered data is discarded
- Difficulty to get reliable data around T_c

Initial Idea

- Want to update whole clusters
- How to identify these clusters?
- First steps by Kasteleyn and Fortuin
- Equivalent to the Percolation Problem

Idea behind Cluster Updating

- Percolation Problem has no Slowing Down
- Map NN Bonds onto an equivalent lattice
- Probability for NN to be in the same cluster:

$$p = 1 - e^{-\frac{2\mathcal{J}}{k_B T}\delta_{\sigma_i \sigma_j}}$$

Swendson–Wang method (1987)

- Iterate through all bonds, try placing them
- Identify all clusters
- Radomly assign a new value

Wolff method

- Clusters are gown from a single site (recursively)
- Only one cluster grown at a time
- Cluster is always flipped
- More efficient than the Swendson–Wang method

Visual Demonstration

- Show: Wolff method
- Note: Osciallation at low temperatures
- Sweep Temperature

Relaxation from High Energy State

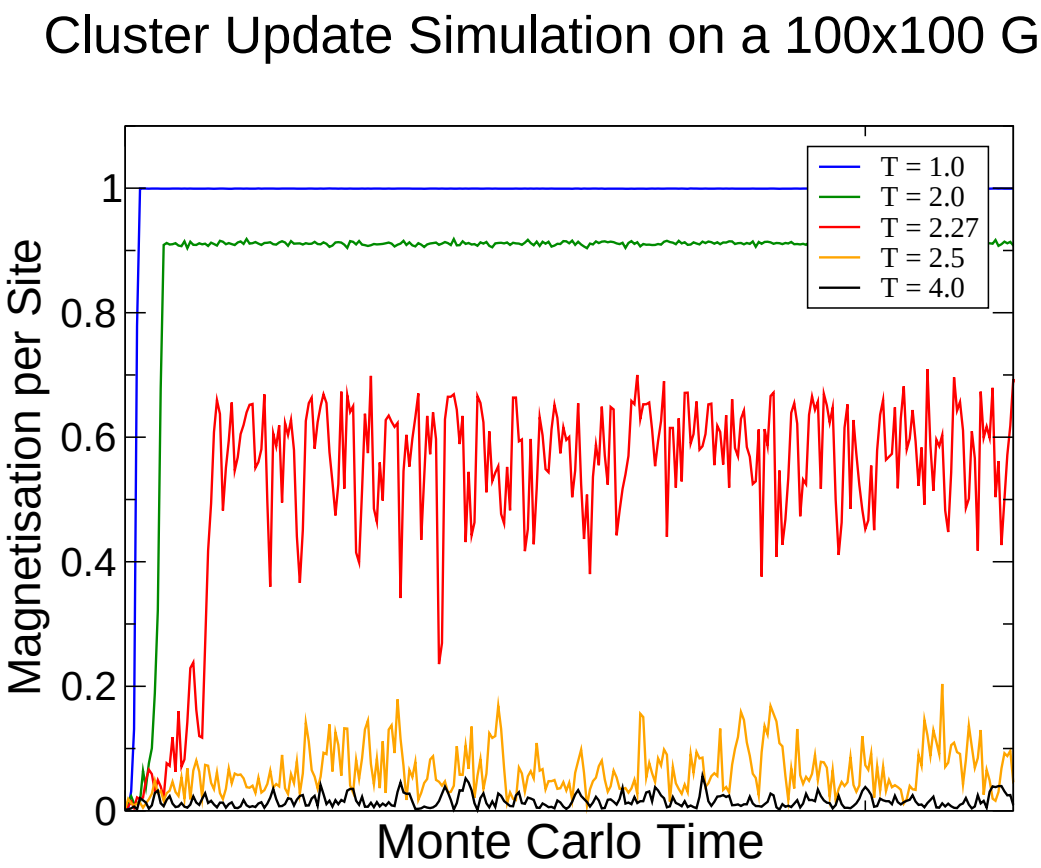

Figure: Relaxation from High Energy State width Cluster Updates

Relaxation from Low Energy State

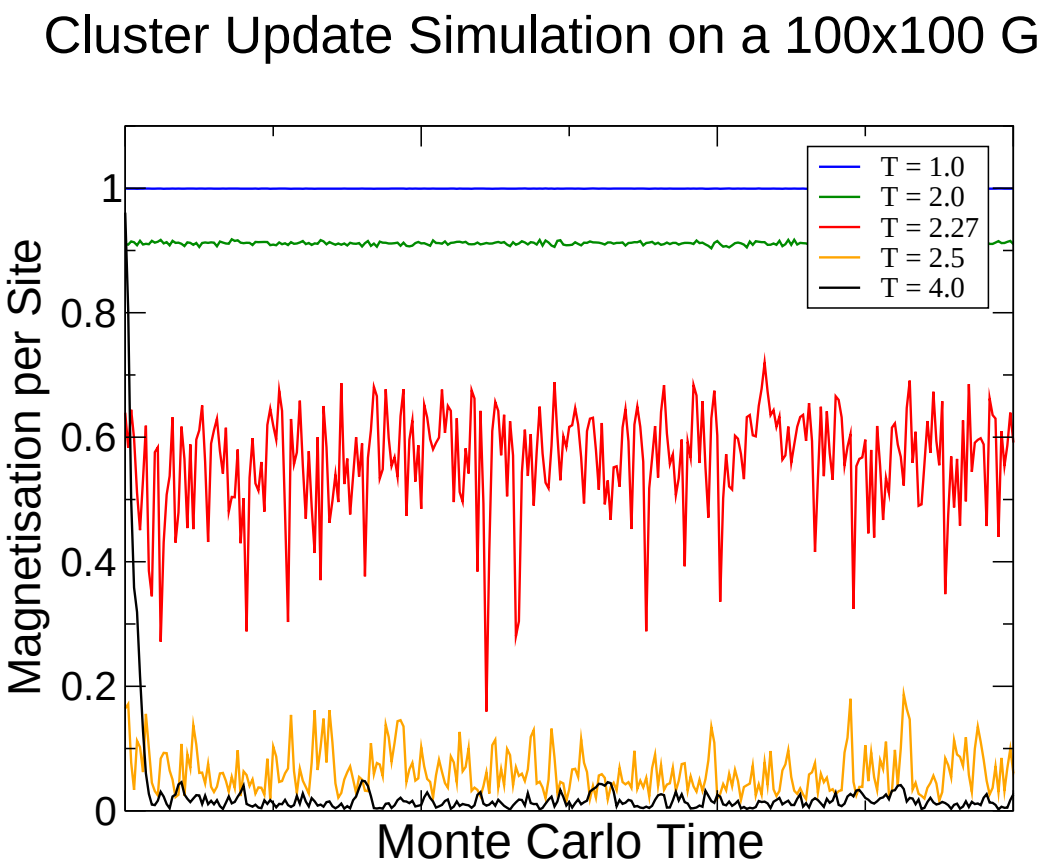

Figure: Relaxation from High Energy State width Cluster Updates

Relaxation compared to Local Updates

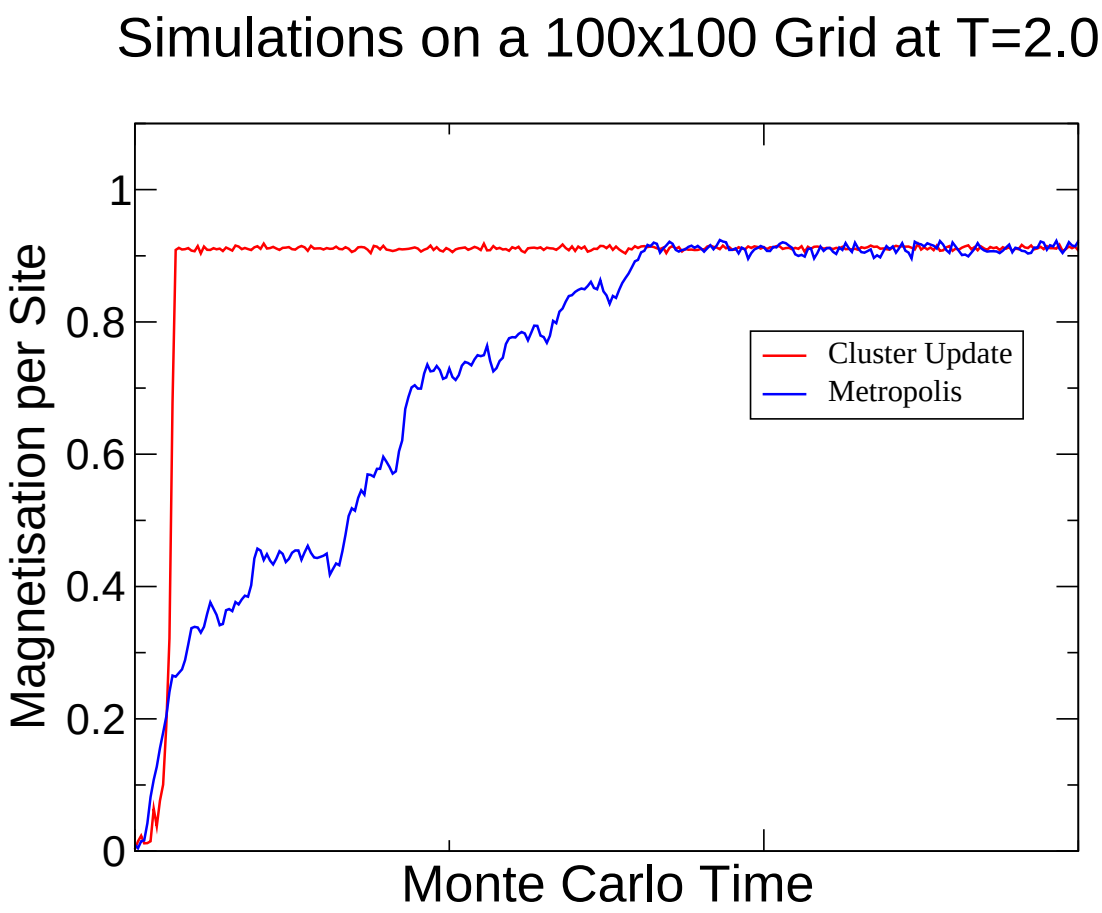

Figure: Comparison of Local and Cluster Update at $T = 2.0$

Relaxation compared to Local Updates

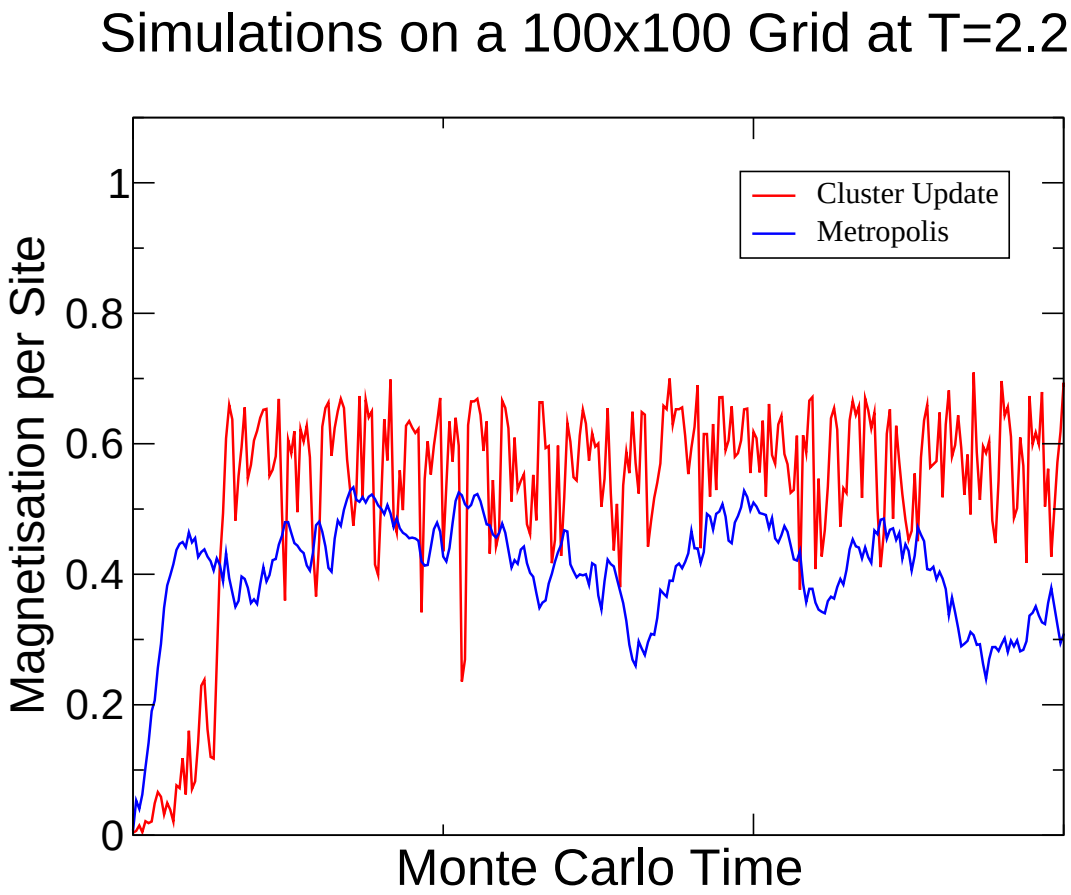

Figure: Comparison of Local and Cluster Update at $T = T_c$

Updates work nicely around T_c

- Fast Relaxation to equilibrium state
- At T_c all sizes of clusters coexist
- Cluster Update works nice at T_c
- Hence we can get reliable data

Drawbacks

- Performance worse than Metropolis away from T_c
- Cannot handle magnetic field (freezes at T_c)
- Not adequate to investigate dynamics

Conclusion

- Monte Carlo Simulations can provide good predictions
- Difficulty at T_c can be overcome
- Cluster Updates also have drawbacks

Sources

1. D. P. Landau & K. Binder:
   A Guide to Monte Carlo Simulations in Statistical Physics
   (Cambridge University Press, 2005)
2. K.Binder & D.W.Heermann:
   Monte Carlo Simulation in Statistical Physics (Springer, 2006)
3. http://montecarlo.csi.cuny.edu/umass/lectures.html (03/07)