Cluster Monte Carlo method with a conserved order parameter

L.A. Fernandez, V. Martin-Mayor, **David Yllanes**

Departamento de Física Teórica I
Universidad Complutense de Madrid

Leipzig, November 27th 2009

Nuclear Physics B **807**, 424-454 (2009)
Physical Review E **80**, 015701(R) (2009)

Introduction (I)

Motivation

- All known Monte Carlo methods suffer critical or exponential slowing down when applied to important problems:
  Gauge theories, structural glasses, spin glasses, protein folding,. . .

Introduction (I)

Motivation

- All known Monte Carlo methods suffer critical or exponential slowing down when applied to important problems:
  Gauge theories, structural glasses, spin glasses, protein folding,...
- Only for very few systems can one define an efficient cluster method that eliminates this problem.

Introduction (I)

Motivation

- All known Monte Carlo methods suffer critical or exponential slowing down when applied to important problems:
  Gauge theories, structural glasses, spin glasses, protein folding,. . .
- Only for very few systems can one define an efficient cluster method that eliminates this problem.
- On the other hand, considering global conservation laws is often useful (e.g., micromagnetical ensembles).

Introduction (I)

Motivation

- All known Monte Carlo methods suffer critical or exponential slowing down when applied to important problems:
  Gauge theories, structural glasses, spin glasses, protein folding,. . .
- Only for very few systems can one define an efficient cluster method that eliminates this problem.
- On the other hand, considering global conservation laws is often useful (e.g., micromagnetical ensembles).
- Combining cluster methods with fixed-magnetisation ensembles was soon identified as a key (but probably intractable) challenge.

Introduction (I)

Motivation

- All known Monte Carlo methods suffer critical or exponential slowing down when applied to important problems:
  Gauge theories, structural glasses, spin glasses, protein folding,...
- Only for very few systems can one define an efficient cluster method that eliminates this problem.
- On the other hand, considering global conservation laws is often useful (e.g., micromagnetical ensembles).
- Combining cluster methods with fixed-magnetisation ensembles was soon identified as a key (but probably intractable) challenge.
- We present a working cluster algorithm with a globally conserved order parameter.

Introduction (I)

Motivation

- All known Monte Carlo methods suffer critical or exponential slowing down when applied to important problems:
  Gauge theories, structural glasses, spin glasses, protein folding,. . .
- Only for very few systems can one define an efficient cluster method that eliminates this problem.
- On the other hand, considering global conservation laws is often useful (e.g., micromagnetical ensembles).
- Combining cluster methods with fixed-magnetisation ensembles was soon identified as a key (but probably intractable) challenge.
- We present a working cluster algorithm with a globally conserved order parameter.
- We work in the Tethered Monte Carlo framework (introduced at CompPhys08), an (almost) fixed-magnetisation ensemble.

Introduction (II)

Main features

- Tethered Ensemble: original d.o.f. + Gaussian *magnetostat*:
  - Micromagnetic ensemble: fixed β and order parameter (m).
  - Tethered ensemble: fixed β and $\hat{m} = m +$ [Gaussian bath].
  - Related to *Creutz's microcanonical demon*. Main differences:
    - Continuous demons, coupled to m rather than to energy.

Introduction (II)

Main features

- Tethered Ensemble: original d.o.f. + Gaussian *magnetostat*:
  - Micromagnetic ensemble: fixed β and order parameter (m).
  - Tethered ensemble: fixed β and $\hat{m} = m +$ [Gaussian bath].
  - Related to *Creutz's microcanonical demon*. Main differences:
    - Continuous demons, coupled to m rather than to energy.
    - Extensive number of demons, which are eventually integrated out.
- Independent simulations at fixed $\hat{m}$. Later, reconstruction of *canonical* effective potential Ω_N.

Introduction (II)

Main features

- Tethered Ensemble: original d.o.f. + Gaussian *magnetostat*:
  - Micromagnetic ensemble: fixed β and order parameter (m).
  - Tethered ensemble: fixed β and $\hat{m} = m +$ [Gaussian bath].
  - Related to *Creutz's microcanonical demon*. Main differences:
    - Continuous demons, coupled to m rather than to energy.
    - Extensive number of demons, which are eventually integrated out.
- Independent simulations at fixed $\hat{m}$. Later, reconstruction of *canonical* effective potential Ω_N.
- Canonical averages are recovered for any value of the external field h without new simulations.
- Local algorithm (e.g. Metropolis) straightforward.
  - No critical slowing down for magnetic observables.
  - Other quantities have typical $z = 2$ behavior.

Introduction (II)

Main features

- Tethered Ensemble: original d.o.f. + Gaussian *magnetostat*:
  - Micromagnetic ensemble: fixed β and order parameter (m).
  - Tethered ensemble: fixed β and $\hat{m} = m +$ [Gaussian bath].
  - Related to *Creutz's microcanonical demon*. Main differences:
    - Continuous demons, coupled to m rather than to energy.
    - Extensive number of demons, which are eventually integrated out.
- Independent simulations at fixed $\hat{m}$. Later, reconstruction of *canonical* effective potential Ω_N.
- Canonical averages are recovered for any value of the external field h without new simulations.
- Local algorithm (e.g. Metropolis) straightforward.
  - No critical slowing down for magnetic observables.
  - Other quantities have typical $z = 2$ behavior.
- Here we implement a Swendsen-Wang update scheme and present our results for the $D = 2, 3$ Ising model.

Notations: Ising model

- Standard benchmark for MC simulation methods.
- Partition function and main observables ($N = L^D$):

$$Z = \sum_{\{\sigma_{\boldsymbol{x}}\}} \exp\Big[\beta \sum_{\langle \boldsymbol{x},\boldsymbol{y}\rangle} \sigma_{\boldsymbol{x}}\sigma_{\boldsymbol{y}} + h \sum_{\boldsymbol{x}} \sigma_{\boldsymbol{x}}\Big], \qquad \sigma_{\boldsymbol{x}} = \pm 1,$$

$$E = Ne = -\sum_{\langle \boldsymbol{x},\boldsymbol{y}\rangle} \sigma_{\boldsymbol{x}}\sigma_{\boldsymbol{y}}, \qquad M = Nm = \sum_{\boldsymbol{x}} \sigma_{\boldsymbol{x}}.$$

- We denote canonical averages by $\langle \cdots \rangle_\beta$:

$$C = N[\langle e^2\rangle_\beta - \langle e\rangle_\beta^2], \qquad \chi = N[\langle m^2\rangle_\beta - \langle m\rangle_\beta^2].$$

The Tethered Ensemble (I)

- Canonical pdf for order parameter ($h = 0$),

$$p_1(m) = \frac{1}{Z} \sum_{\{\sigma_{\boldsymbol{x}}\}} \exp[-\beta E] \delta\Big(m - \sum_i \sigma_i / N \Big)$$

The Tethered Ensemble (I)

- Canonical pdf for order parameter ($h = 0$),

$$p_1(m) = \frac{1}{Z} \sum_{\{\sigma_{\boldsymbol{x}}\}} \exp[-\beta E] \delta\Big(m - \sum_i \sigma_i / N\Big)$$

- Extend configuration space with N decoupled Gaussian demons

$$Z = \int_{-\infty}^{\infty} \prod_{i=1}^{N} \mathrm{d}\eta_i \sum_{\{\sigma_{\boldsymbol{x}}\}} \exp\Big[-\beta E - \sum_i \phi_i^2/2\Big], \qquad R = Nr = \sum_i \phi_i^2/2.$$

- r (almost) Gaussian distributed, $r = \frac{1}{2} + \frac{\zeta}{\sqrt{N}}$, $|\zeta| \sim 1$.

The Tethered Ensemble (I)

- Canonical pdf for order parameter ($h = 0$),

$$p_1(m) = \frac{1}{Z} \sum_{\{\sigma_{\boldsymbol{x}}\}} \exp[-\beta E] \delta\Big(m - \sum_i \sigma_i / N\Big)$$

- Extend configuration space with N decoupled Gaussian demons

$$Z = \int_{-\infty}^{\infty} \prod_{i=1}^{N} \mathrm{d}\eta_i \sum_{\{\sigma_{\boldsymbol{x}}\}} \exp\Big[-\beta E - \sum_i \phi_i^2/2\Big], \qquad R = Nr = \sum_i \phi_i^2/2.$$

- r (almost) Gaussian distributed, $r = \frac{1}{2} + \frac{\zeta}{\sqrt{N}}$, $|\zeta| \sim 1$.
- Let $\hat{m} = m + r$. Its pdf is a convolution (m and r independent) $\rightarrow$ $p(\hat{m} = m + \frac{1}{2})$ is a *smoothing* of $p_1(m)$.

The Tethered Ensemble (I)

- Canonical pdf for order parameter ($h = 0$),

$$p_1(m) = \frac{1}{Z} \sum_{\{\sigma_{\boldsymbol{x}}\}} \exp[-\beta E] \delta\Big(m - \sum_i \sigma_i / N\Big)$$

- Extend configuration space with N decoupled Gaussian demons

$$Z = \int_{-\infty}^{\infty} \prod_{i=1}^{N} \mathrm{d}\eta_i \sum_{\{\sigma_{\boldsymbol{x}}\}} \exp\Big[-\beta E - \sum_i \phi_i^2/2\Big], \qquad R = Nr = \sum_i \phi_i^2/2.$$

- r (almost) Gaussian distributed, $r = \frac{1}{2} + \frac{\zeta}{\sqrt{N}}$, $|\zeta| \sim 1$.
- Let $\hat{m} = m + r$. Its pdf is a convolution (m and r independent) $\rightarrow p(\hat{m} = m + \frac{1}{2})$ is a *smoothing* of $p_1(m)$.
- A smooth $p(\hat{m})$ has an effective potential $\Omega_N(\hat{m}, \beta)$

$$p(\hat{m}) = \frac{1}{Z} \int_{-\infty}^{\infty} \prod_{i=1}^{N} \mathrm{d}\eta_i \sum_{\{\sigma_{\boldsymbol{x}}\}} \mathrm{e}^{-\beta E - \sum_i \frac{\eta_i^2}{2}} \delta\Big(\hat{m} - m - \sum_i \frac{\eta_i^2}{2N}\Big) = \mathrm{e}^{N\Omega_N(\hat{m},\beta)}$$

The tethered ensemble (II)

- Integrating demons out in the *constrained* (fixed $\hat{m}$) partition function → tethered expectation values:

$$\langle O \rangle_{\hat{m},\beta} = \frac{\sum_{\{\sigma_{\boldsymbol{x}}\}} O(\hat{m};\{\sigma_{\boldsymbol{x}}\})\,\omega(\beta,\hat{m},N;\{\sigma_{\boldsymbol{x}}\})}{\sum_{\{\sigma_{\boldsymbol{x}}\}} \omega(\beta,\hat{m},N;\{\sigma_{\boldsymbol{x}}\})},$$

The tethered ensemble (II)

- Integrating demons out in the *constrained* (fixed $\hat{m}$) partition function → tethered expectation values:

$$\langle O \rangle_{\hat{m},\beta} = \frac{\sum_{\{\sigma_{\boldsymbol{x}}\}} O(\hat{m};\{\sigma_{\boldsymbol{x}}\})\,\omega(\beta,\hat{m},N;\{\sigma_{\boldsymbol{x}}\})}{\sum_{\{\sigma_{\boldsymbol{x}}\}} \omega(\beta,\hat{m},N;\{\sigma_{\boldsymbol{x}}\})},$$

$$\omega(\beta,\hat{m},N;\{\sigma_{\boldsymbol{x}}\}) = \mathrm{e}^{-\beta E + M - N\hat{m}}(\hat{m}-m)^{(N-2)/2}\,\theta(\hat{m}-m).$$

The tethered ensemble (II)

- Integrating demons out in the *constrained* (fixed $\hat{m}$) partition function → tethered expectation values:

$$\langle O \rangle_{\hat{m},\beta} = \frac{\sum_{\{\sigma_{\boldsymbol{x}}\}} O(\hat{m};\{\sigma_{\boldsymbol{x}}\})\,\omega(\beta,\hat{m},N;\{\sigma_{\boldsymbol{x}}\})}{\sum_{\{\sigma_{\boldsymbol{x}}\}} \omega(\beta,\hat{m},N;\{\sigma_{\boldsymbol{x}}\})},$$

$$\omega(\beta,\hat{m},N;\{\sigma_{\boldsymbol{x}}\}) = \mathrm{e}^{-\beta E + M - N\hat{m}}(\hat{m}-m)^{(N-2)/2}\,\theta(\hat{m}-m).$$

- The *canonical* Ω_N follows from Fluctuation-Dissipation

$$\hat{h}(\hat{m};\{\sigma_{\boldsymbol{x}}\}) = -1 + \frac{N/2-1}{\hat{M}-M} \quad \Longrightarrow \quad \langle \hat{h} \rangle_{\hat{m},\beta} = \frac{\partial \Omega_N(\hat{m},\beta)}{\partial \hat{m}}.$$

The tethered ensemble (II)

- Integrating demons out in the *constrained* (fixed $\hat{m}$) partition function → tethered expectation values:

$$\langle O \rangle_{\hat{m},\beta} = \frac{\sum_{\{\sigma_{\boldsymbol{x}}\}} O(\hat{m}; \{\sigma_{\boldsymbol{x}}\}) \omega(\beta, \hat{m}, N; \{\sigma_{\boldsymbol{x}}\})}{\sum_{\{\sigma_{\boldsymbol{x}}\}} \omega(\beta, \hat{m}, N; \{\sigma_{\boldsymbol{x}}\})},$$

$$\omega(\beta, \hat{m}, N; \{\sigma_{\boldsymbol{x}}\}) = \mathrm{e}^{-\beta E + M - N\hat{m}} (\hat{m} - m)^{(N-2)/2}\, \theta(\hat{m} - m).$$

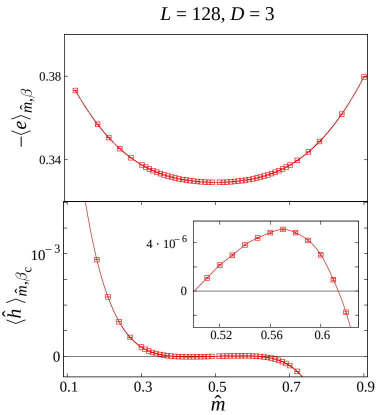
- The *canonical* Ω_N follows from Fluctuation-Dissipation

$$\hat{h}(\hat{m}; \{\sigma_{\boldsymbol{x}}\}) = -1 + \frac{N/2 - 1}{\hat{M} - M} \quad \Longrightarrow \quad \langle \hat{h} \rangle_{\hat{m},\beta} = \frac{\partial \Omega_N(\hat{m}, \beta)}{\partial \hat{m}}.$$

- Tethered mean values $\langle O \rangle_{\hat{m},\beta}$ ↔ canonical mean values $\langle O \rangle_\beta(h)$, for any external field h:

$$\langle O \rangle_\beta(h) = \int \mathrm{d}\hat{m}\; \langle O \rangle_{\hat{m},\beta} \exp[N(\Omega_N(\hat{m}, \beta) + h\hat{m})].$$

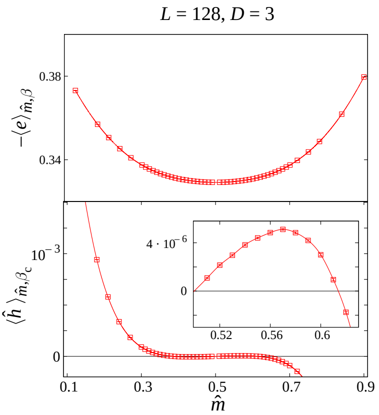
Numerical methods



Steps

1. Select a mesh of $\hat{m}$ values.

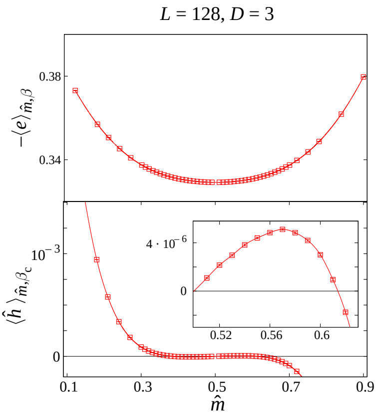
Numerical methods



Steps

1. Select a mesh of $\hat{m}$ values.
2. Independent simulation for each $\hat{m}$. Get $\langle O \rangle_{\hat{m},\beta}$.

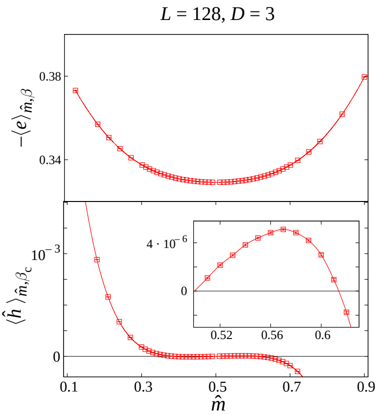
Numerical methods



Steps

1. Select a mesh of $\hat{m}$ values.
2. Independent simulation for each $\hat{m}$. Get $\langle O \rangle_{\hat{m},\beta}$.
3. $\langle O \rangle_{\hat{m},\beta}$ smooth functions of $\hat{m}$ → interpolate (e.g. spline).

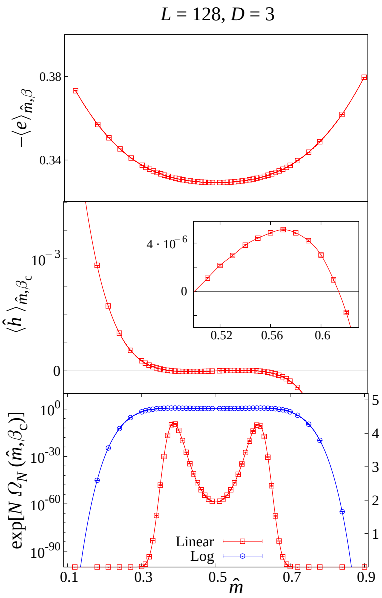
Numerical methods



Steps

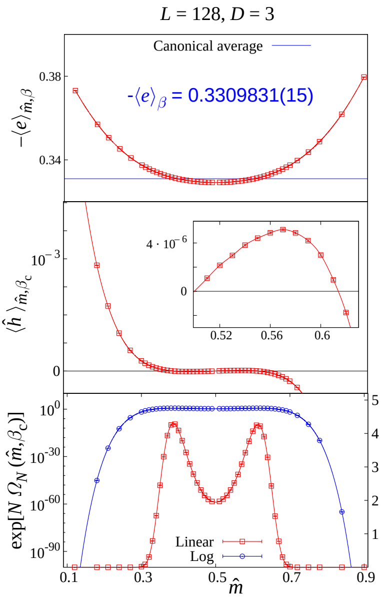
1. Select a mesh of $\hat{m}$ values.
2. Independent simulation for each $\hat{m}$. Get $\langle O \rangle_{\hat{m},\beta}$.
3. $\langle O \rangle_{\hat{m},\beta}$ smooth functions of $\hat{m}$ $\rightarrow$ interpolate (e.g. spline).
4. Numerical integration of $\langle \hat{h} \rangle_{\hat{m},\beta}$ yields $\Omega_N(\hat{m},\beta)$.

Numerical methods



Steps

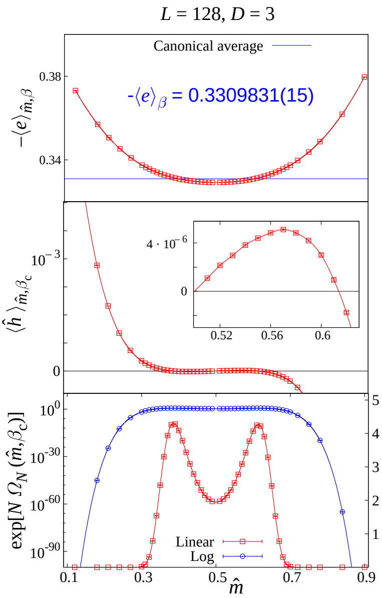
1. Select a mesh of $\hat m$ values.
2. Independent simulation for each $\hat m$. Get $\langle O\rangle_{\hat m,\beta}$.
3. $\langle O\rangle_{\hat m,\beta}$ smooth functions of $\hat m$ → interpolate (e.g. spline).
4. Numerical integration of $\langle \hat h\rangle_{\hat m,\beta}$ yields $\Omega_N(\hat m,\beta)$.
5. Reconstruct canonical $\langle O\rangle_\beta$ from $p(\hat m)$ and $\langle O\rangle_{\hat m,\beta}$.



Steps

1. Select a mesh of $\hat m$ values.
2. Independent simulation for each $\hat m$. Get $\langle O\rangle_{\hat m,\beta}$.
3. $\langle O\rangle_{\hat m,\beta}$ smooth functions of $\hat m$ → interpolate (e.g. spline).
4. Numerical integration of $\langle \hat h\rangle_{\hat m,\beta}$ yields $\Omega_N(\hat m,\beta)$.
5. Reconstruct canonical $\langle O\rangle_\beta$ from $p(\hat m)$ and $\langle O\rangle_{\hat m,\beta}$.
6. Statistical errors: jackknife.

Numerical methods



Steps

1. Select a mesh of $\hat{m}$ values.
2. Independent simulation for each $\hat{m}$. Get $\langle O \rangle_{\hat{m},\beta}$.
3. $\langle O \rangle_{\hat{m},\beta}$ smooth functions of $\hat{m}$ → interpolate (e.g. spline).
4. Numerical integration of $\langle \hat{h} \rangle_{\hat{m},\beta}$ yields $\Omega_N(\hat{m},\beta)$.
5. Reconstruct canonical $\langle O \rangle_{\beta}$ from $p(\hat{m})$ and $\langle O \rangle_{\hat{m},\beta}$.
6. Statistical errors: jackknife.
7. Systematic errors: refine $\hat{m}$ grid.

The cluster update algorithm (I)

- We can follow the Fortuin-Kasteleyn construction.
- Introduce bond-occupation variables n_{xy} $(= 0, 1)$:

$$\mathrm{e}^{\beta(\sigma_x \sigma_y - 1)} = \sum_{n_{xy}=0,1} [(1-p)\delta_{n_{xy},0} + p\delta_{\sigma_x,\sigma_y}\, \delta_{n_{xy},1}], \quad p = 1-\mathrm{e}^{-2\beta}.$$

these bonds form N_C clusters.

The cluster update algorithm (I)

- We can follow the Fortuin-Kasteleyn construction.
- Introduce bond-occupation variables n_{xy} $(= 0, 1)$:

$$\mathrm{e}^{\beta(\sigma_x \sigma_y - 1)} = \sum_{n_{xy}=0,1} [(1-p)\delta_{n_{xy},0} + p\delta_{\sigma_x,\sigma_y}\,\delta_{n_{xy},1}], \quad p = 1 - \mathrm{e}^{-2\beta}.$$

these bonds form N_C clusters.

- Combine this identity with $\omega(\beta, \hat{m}; \{\sigma_x\})$:

$$\omega(\beta, \hat{m}; \{\sigma_{\boldsymbol{x}}\}) = \mathrm{e}^{-\beta E + M - N\hat{m}} (\hat{m} - m)^{(N-2)/2}\, \theta(\hat{m} - m).$$

The cluster update algorithm (I)

- We can follow the Fortuin-Kasteleyn construction.
- Introduce bond-occupation variables n_{xy} $(= 0, 1)$:

$$\mathrm{e}^{\beta(\sigma_x \sigma_y - 1)} = \sum_{n_{xy}=0,1} [(1-p)\delta_{n_{xy},0} + p\delta_{\sigma_x,\sigma_y}\, \delta_{n_{xy},1}], \quad p = 1 - \mathrm{e}^{-2\beta}.$$

  these bonds form N_C clusters.
- Combine this identity with $\omega(\beta, \hat{m}; \{\sigma_x\})$:

$$\omega(\beta, \hat{m}; \{\sigma_{\boldsymbol{x}}\}) = \mathrm{e}^{-\beta E + M - N\hat{m}} (\hat{m} - m)^{(N-2)/2}\, \theta(\hat{m} - m).$$

- We have the following conditional probabilities:
  - Given $\{\sigma_x\}$, just as in the canonical case, bonds are independent and n_{xy} is 1 with probability $p\delta_{\sigma_x,\sigma_y}$.

The cluster update algorithm (I)

- We can follow the Fortuin-Kasteleyn construction.
- Introduce bond-occupation variables n_{xy} $(= 0, 1)$:

$$\mathrm{e}^{\beta(\sigma_x\sigma_y-1)} = \sum_{n_{xy}=0,1} [(1-p)\delta_{n_{xy},0}+p\delta_{\sigma_x,\sigma_y}\,\delta_{n_{xy},1}], \quad p = 1-\mathrm{e}^{-2\beta}.$$

  these bonds form N_C clusters.
- Combine this identity with $\omega(\beta,\hat{m};\{\sigma_x\})$:

$$\omega(\beta,\hat{m};\{\sigma_{\boldsymbol{x}}\}) = \mathrm{e}^{-\beta E+M-N\hat{m}}(\hat{m}-m)^{(N-2)/2}\,\theta(\hat{m}-m).$$

- We have the following conditional probabilities:
  - Given $\{\sigma_x\}$, just as in the canonical case, bonds are independent and n_{xy} is 1 with probability $p\delta_{\sigma_x,\sigma_y}$.
  - Given $\{n_{xy}\}$, the spins within cluster i are equal to $S_i = \pm 1$. Not all $\{S_i\}$ configurations have the same probability:

$$p(\{S_i\}) \propto \mathrm{e}^{M-\hat{M}}(\hat{M}-M)^{(N-2)/2}\,\theta(\hat{M}-M).$$

The cluster update algorithm (II)

a) Cluster tracing

b) Cluster flipping:

The cluster update algorithm (II)

a) Cluster tracing (as in canonical SW)

b) Cluster flipping:

The cluster update algorithm (II)

a) Cluster tracing (as in canonical SW)

b) Cluster flipping: It is impossible to consider all 2^{N_C} cluster orientations

The cluster update algorithm (II)

a) Cluster tracing (as in canonical SW)

b) Cluster flipping: We perform a dynamic MC

1. Select $N'_C \ll N_C$ clusters (for instance $N'_C \approx 5$).

The cluster update algorithm (II)

a) Cluster tracing (as in canonical SW)

b) Cluster flipping: We perform a dynamic MC

1. Select $N'_C \ll N_C$ clusters (for instance $N'_C \approx 5$).
2. Heat bath among the $2^{N'_C}$ conf. with $N_C - N'_C$ clusters fixed.

The cluster update algorithm (II)

a) Cluster tracing (as in canonical SW)

b) Cluster flipping: We perform a dynamic MC

1. Select $N'_C \ll N_C$ clusters (for instance $N'_C \approx 5$).
2. Heat bath among the $2^{N'_C}$ conf. with $N_C - N'_C$ clusters fixed.
3. Take measurements.

Steps 1–3 are repeated N_{REP} times, without retracing the clusters.

The cluster update algorithm (II)

a) Cluster tracing (as in canonical SW)

b) Cluster flipping: We perform a dynamic MC

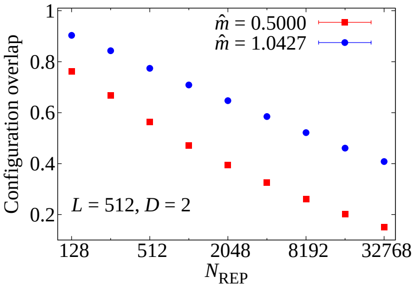
1. Select $N_C' \ll N_C$ clusters (for instance $N_C' \approx 5$).
2. Heat bath among the $2^{N_C'}$ conf. with $N_C - N_C'$ clusters fixed.
3. Take measurements.

Steps 1–3 are repeated N_{REP} times, without retracing the clusters.

- We consider the overlap

$$o = \frac{\langle \sum_{\boldsymbol{x}} [\sigma_{\boldsymbol{x}}^{t=0} \sigma_{\boldsymbol{x}}^{t=N_{\text{rep}}} - \langle m \rangle^2_{\hat{m},\beta}] \rangle_{\hat{m},\beta}}{N(1 - \langle m \rangle^2_{\hat{m},\beta})},$$

- Our N_{REP}: Tracing time $\approx$ Flipping time



The cluster update algorithm (II)

a) Cluster tracing (as in canonical SW)

b) Cluster flipping: We perform a dynamic MC

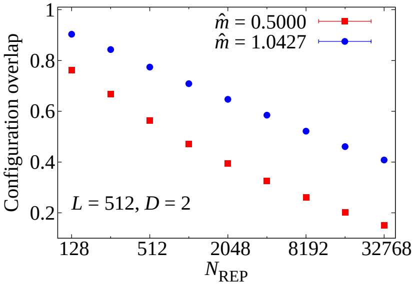
1. Select $N'_C \ll N_C$ clusters (for instance $N'_C \approx 5$).
2. Heat bath among the $2^{N'_C}$ conf. with $N_C - N'_C$ clusters fixed.
3. Take measurements.

Steps 1–3 are repeated N_{REP} times, without retracing the clusters.

- We consider the overlap

$$o = \frac{\langle \sum_{\boldsymbol{x}} [\sigma_{\boldsymbol{x}}^{t=0} \sigma_{\boldsymbol{x}}^{t=N_{\text{rep}}} - \langle m \rangle_{\hat{m},\beta}^2] \rangle_{\hat{m},\beta}}{N(1 - \langle m \rangle_{\hat{m},\beta}^2)},$$

- Our N_{REP}: Tracing time $\approx$ Flipping time
- Measuring $\hat{h}$ at each of the N_{REP} steps reduces errors by up to a factor 25.



Results at the critical point

Integrated autocorrelation times

- The slowest observable is E.
- τ_{int} largest at $\hat{m} = 0.5$ ($m \simeq 0$).

Results at the critical point

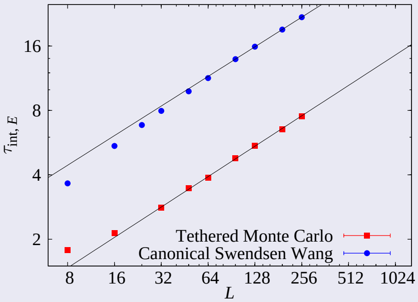
Integrated autocorrelation times

- The slowest observable is E.
- τ_{int} largest at $\hat{m} = 0.5$ ($m \simeq 0$).
- We fit to $\tau_{\text{int}} = AL^z$:
  - $D = 2 \longrightarrow z = 0.241(7)$

Results at the critical point

Integrated autocorrelation times

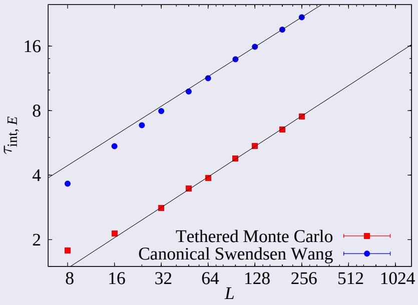
- The slowest observable is E.
- τ_{int} largest at $\hat{m} = 0.5$ ($m \simeq 0$).
- We fit to $\tau_{\text{int}} = AL^z$:
  - $D = 2 \longrightarrow z = 0.241(7)$
  - $D = 3 \longrightarrow z = 0.472(8)$



Results at the critical point

Integrated autocorrelation times

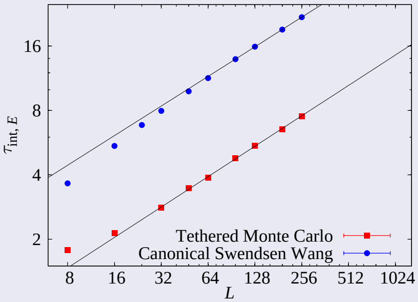
- The slowest observable is E.
- τ_{int} largest at $\hat{m} = 0.5$ ($m \simeq 0$).
- We fit to $\tau_{\text{int}} = AL^z$:
  - $D = 2 \longrightarrow z = 0.241(7)$
  - $D = 3 \longrightarrow z = 0.472(8)$
- $D = 3$, z compatible with that of canonical Swendsen-Wang (data from Ossola and Sokal, 2001)



Results at the critical point

Integrated autocorrelation times

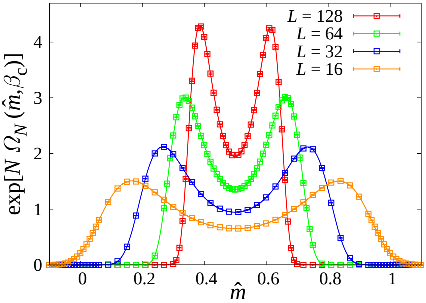
- The slowest observable is E.
- τ_{int} largest at $\hat{m} = 0.5$ ($m \simeq 0$).
- We fit to $\tau_{\text{int}} = AL^z$:
  - $D = 2 \longrightarrow z = 0.241(7)$
  - $D = 3 \longrightarrow z = 0.472(8)$
- $D = 3$, z compatible with that of canonical Swendsen-Wang (data from Ossola and Sokal, 2001)



Canonical averages for $L = 128$, $D = 3$

	MCS	$-\langle e \rangle_\beta$	C	χ	ξ
SW	48×10^6	0.3309822(16)	22.155(18)	21193(13)	82.20(3)
TMC	50×10^6	0.3309831(15)	22.174(13)	21202(13)	82.20(5)

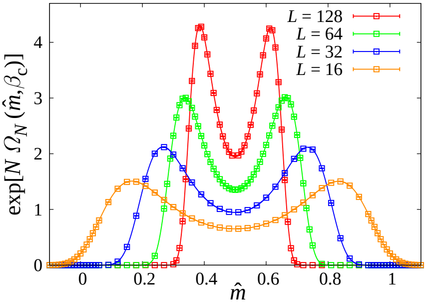
A funny way of computing the anomalous dimension



L	$\hat{m}_{\text{peak}} - \frac{1}{2}$
48	0.18956(4)
64	0.16341(4)
96	0.13240(4)
128	0.114083(24)
192	0.09246(4)
256	0.07959(12)
η	0.0360(7)

- $p(\hat{m}, \beta_{\text{c}}, L) = L^{\frac{\beta}{\nu}} f\left(L^{\frac{\beta}{\nu}}(\hat{m} - \frac{1}{2})\right)$

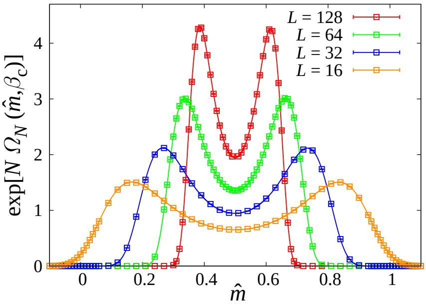
A funny way of computing the anomalous dimension



L	$\hat{m}_{\text{peak}} - \frac{1}{2}$
48	0.18956(4)
64	0.16341(4)
96	0.13240(4)
128	0.114083(24)
192	0.09246(4)
256	0.07959(12)
η	0.0360(7)

- $p(\hat{m}, \beta_c, L) = L^{\frac{\beta}{\nu}} f\left(L^{\frac{\beta}{\nu}}(\hat{m} - \frac{1}{2})\right) \Longrightarrow \hat{m}_{\text{peak}} - \frac{1}{2} = AL^{-(\eta+D-2)/2} + \ldots$

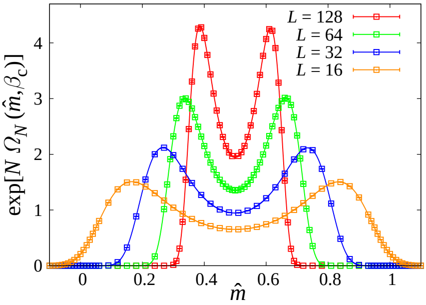
A funny way of computing the anomalous dimension



L	$\hat{m}_{\text{peak}} - \frac{1}{2}$
48	0.18956(4)
64	0.16341(4)
96	0.13240(4)
128	0.114083(24)
192	0.09246(4)
256	0.07959(12)
η	0.0360(7)

- $p(\hat{m}, \beta_c, L) = L^{\frac{\beta}{\nu}} f\left(L^{\frac{\beta}{\nu}}(\hat{m} - \frac{1}{2})\right) \Longrightarrow \hat{m}_{\text{peak}} - \frac{1}{2} = AL^{-(\eta+D-2)/2} + \ldots$
- Finding maxima numerically ill conditioned.

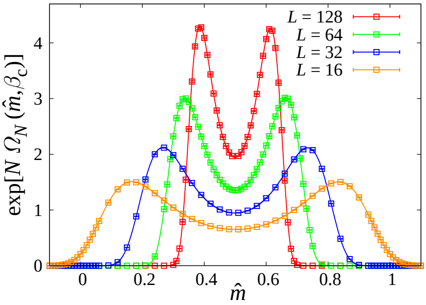
A funny way of computing the anomalous dimension



L	$\hat{m}_{\text{peak}} - \frac{1}{2}$
48	0.18956(4)
64	0.16341(4)
96	0.13240(4)
128	0.114083(24)
192	0.09246(4)
256	0.07959(12)
η	0.0360(7)

- $p(\hat{m},\beta_c,L) = L^{\frac{\beta}{\nu}} f\left(L^{\frac{\beta}{\nu}}(\hat{m}-\frac{1}{2})\right) \Longrightarrow \hat{m}_{\text{peak}} - \frac{1}{2} = AL^{-(\eta+D-2)/2} + \ldots$
- Finding roots is OK: $0 = \langle \hat{h} \rangle_{\hat{m}_{\text{peak}},\beta_c}$

A funny way of computing the anomalous dimension



L	$\hat{m}_{\text{peak}} - \frac{1}{2}$
48	0.18956(4)
64	0.16341(4)
96	0.13240(4)
128	0.114083(24)
192	0.09246(4)
256	0.07959(12)
η	0.0360(7)

- $p(\hat{m},\beta_c,L) = L^{\frac{\beta}{\nu}} f\left(L^{\frac{\beta}{\nu}}(\hat{m}-\frac{1}{2})\right) \Longrightarrow \hat{m}_{\text{peak}} - \frac{1}{2} = AL^{-(\eta+D-2)/2} + \ldots$
- Finding roots is OK: $0 = \langle \hat{h} \rangle_{\hat{m}_{\text{peak}},\beta_c}$
- Previous determinations for $D = 3$:
  - HT-expansion: $\eta = 0.03639(15)$ (Campostrini et al., 2002).
  - MC + perfect action: $\eta = 0.0362(8)$ (Hasenbusch, 2001).

Conclusions and Outlook

- We presented Tethered Monte Carlo,
  a general strategy to compute the canonical effective potential.

Conclusions and Outlook

- We presented Tethered Monte Carlo,
  a general strategy to compute the canonical effective potential.
- Promising when suffering from large tunnelling barriers
  associated to the order parameter.

Conclusions and Outlook

- We presented Tethered Monte Carlo,
  a general strategy to compute the canonical effective potential.
- Promising when suffering from large tunnelling barriers associated to the order parameter.
- Metropolis simulation straightforward and suffers no critical slowing down for magnetic observables.

Conclusions and Outlook

- We presented Tethered Monte Carlo, a general strategy to compute the canonical effective potential.
- Promising when suffering from large tunnelling barriers associated to the order parameter.
- Metropolis simulation straightforward and suffers no critical slowing down for magnetic observables.
- It is possible to define a cluster update method:
  - Even for the computation of canonical averages, it is as efficient as SW in $D = 3$.
  - It can work in a magnetic field without new simulations.
  - New possibilities (e.g., precise computation of η from $p(\hat{m})$).

Conclusions and Outlook

- We presented Tethered Monte Carlo,
  a general strategy to compute the canonical effective potential.
- Promising when suffering from large tunnelling barriers associated to the order parameter.
- Metropolis simulation straightforward and suffers no critical slowing down for magnetic observables.
- It is possible to define a cluster update method:
  - Even for the computation of canonical averages,
    it is as efficient as SW in $D = 3$.
  - It can work in a magnetic field without new simulations.
  - New possibilities (e.g., precise computation of η from $p(\hat{m})$).
- Current and future work
  - Diluted Antiferromagnet in a Field (TMC + Metropolis).
  - Condensation transition (TMC + cluster).