ETH zürich
Institute for Theoretical Physics
High Performance Computational Physics

Correlated Cluster Algorithms

Thea Budde, Marina Krstić Marinković, Joao C. Pinto Barros
SIGN 2025, January 21st

Introduction

Goal: A cluster algorithm for $S = 1/2$ Quantum Link Models

We got: A new type of cluster algorithm with more applications

J. Pinto Barros, **TB**, M. Kristc Marinkovic, arXiv:2402.01039

TB, M. Kristc Marinkovic, J. Pinto Barros, in preparation

Outline

1. Cluster Algorithms
2. Correlated Cluster Algorithm for canonical ensembles
3. Correlated Cluster Algorithm for Abelian Gauge Theories
4. Sign Problems

Outline

1. Cluster Algorithms

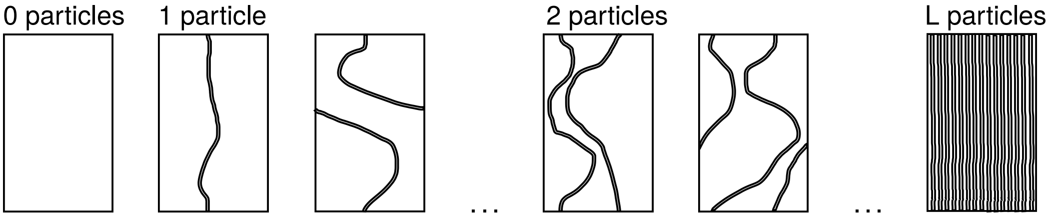
2. Correlated Cluster Algorithm for canonical ensembles

3. Correlated Cluster Algorithm for Abelian Gauge Theories

4. Sign Problems

Worldline Formulation

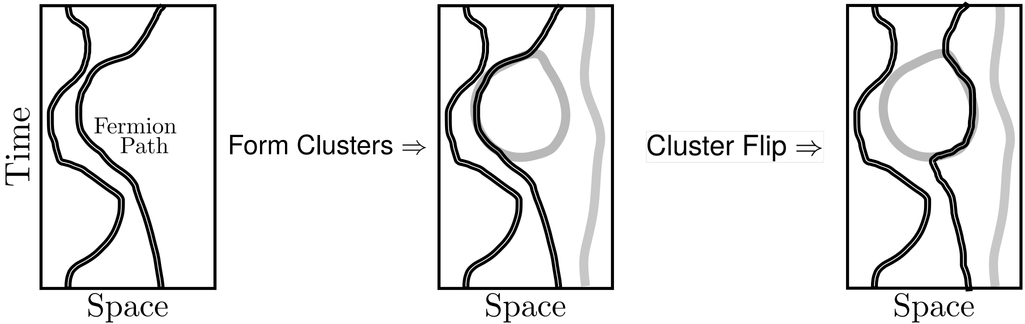
$$\langle \mathcal{O} \rangle_\beta = \frac{\mathrm{Tr}\left(\mathcal{O} e^{-\beta H}\right)}{\mathrm{Tr}\left(e^{-\beta H}\right)} = \sum_{C \text{ worldline configuration}} O(C)\, p(C)$$



Cluster Algorithms

Sample worldline configurations with MCMC with non-local updates

$\rightarrow$ Very efficient in simulating theories for which they have been formulated



U. Wolff, Phys. Rev. Lett. 62, 361 (1989)

N. Prokof’ev and B. Svistunov, Phys. Rev. Lett. 87, 160601 (2001)

Meron Cluster Algorithm

We will be extending the algorithm that simulates the Hamiltonian

$$H = -t \sum_i c_{i+1}^\dagger c_i + c_i^\dagger c_{i+1} + 2\left(c_i^\dagger c_i - \frac{1}{2}\right)\left(c_{i+1}^\dagger c_{i+1} - \frac{1}{2}\right)$$

for a chain of spinless fermions

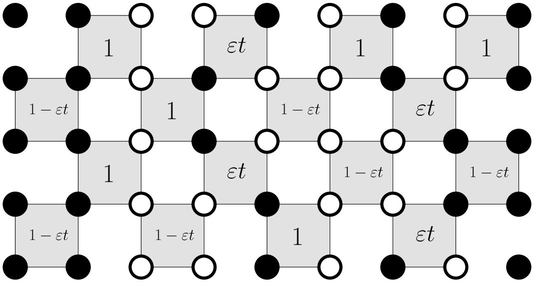
S. Chandrasekharan, U.J. Wiese, Phys.Rev.Lett. 83 (1999) 3116-3119

S. Chandrasekharan, J. Cox, J. C. Osborn and U.J. Wiese NPB 673 405 (2003)

Meron Cluster Algorithm

$$Z = \mathrm{Tr}(e^{-\beta H}) = \sum_i \langle n_i | e^{-\beta H} | n_i \rangle \overset{\text{Trotter}}{\approx} \sum_i \langle n_i | \left(e^{-\frac{\beta}{N} H_{\text{even}}} e^{-\frac{\beta}{N} H_{\text{odd}}} \right)^N | n_i \rangle$$

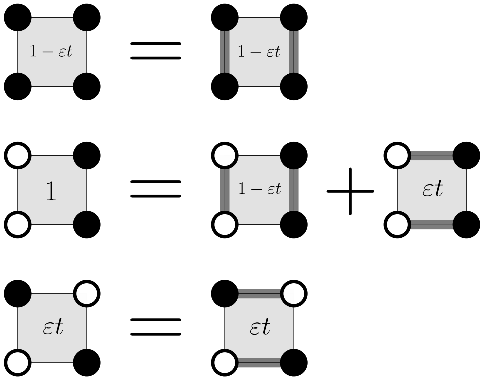
$$= \sum_{n_0, n_1, n_2, n_3, \ldots} \langle n_0 | e^{-\epsilon H_{\text{even}}} | n_1 \rangle \langle n_1 | e^{-\epsilon H_{\text{odd}}} | n_2 \rangle \langle n_2 | \ldots | n_0 \rangle$$



S. Chandrasekharan, U.J. Wiese, Phys.Rev.Lett. 83 (1999) 3116-3119

Meron Cluster Algorithm

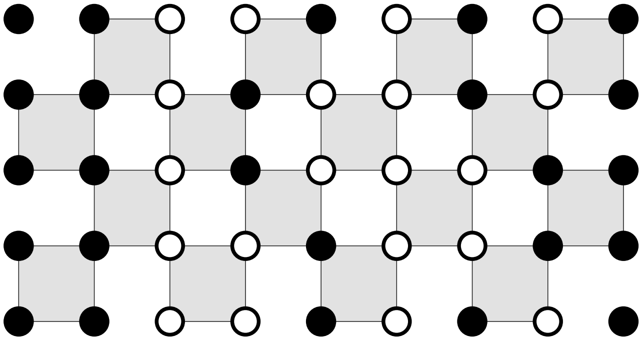
Add additional degrees of freedom: break-ups



S. Chandrasekharan, U.J. Wiese, Phys.Rev.Lett. 83 (1999) 3116-3119

Meron Cluster Algorithm

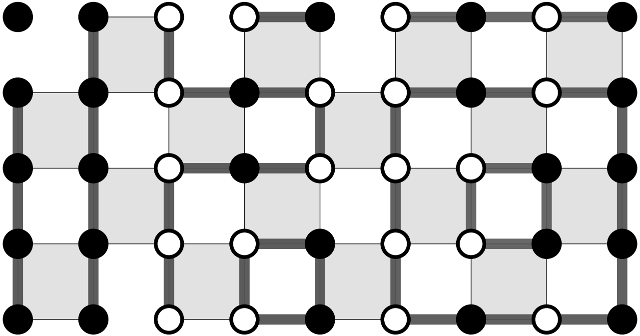
Place breakups with probabilities according to the plaquette weight



S. Chandrasekharan, U.J. Wiese, Phys.Rev.Lett. 83 (1999) 3116-3119

Meron Cluster Algorithm

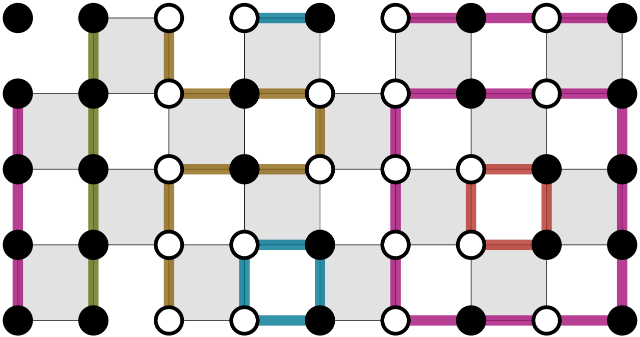
Place breakups with probabilities according to the plaquette weight



S. Chandrasekharan, U.J. Wiese, Phys.Rev.Lett. 83 (1999) 3116-3119

Meron Cluster Algorithm

Place breakups with probabilities according to the plaquette weight ⇒ Clusters



S. Chandrasekharan, U.J. Wiese, Phys.Rev.Lett. 83 (1999) 3116-3119

Meron Cluster Algorithm

A cluster flip gives a configuration with equal weight

$\rightarrow$ Always accepted

S. Chandrasekharan, U.J. Wiese, Phys.Rev.Lett. 83 (1999) 3116-3119

Meron Cluster Algorithm

A cluster flip gives a configuration with equal weight

$\rightarrow$ Always accepted

Flip each cluster with $p = \frac{1}{2}$

$\rightarrow$ Almost uncorrelated with the previous configuration
$\rightarrow$ Ergodic

There is still a sign problem left to address

S. Chandrasekharan, U.J. Wiese, Phys.Rev.Lett. 83 (1999) 3116-3119

Meron Cluster Algorithm

A cluster flip gives a configuration with equal weight

$\rightarrow$ Always accepted

Flip each cluster with $p = \frac{1}{2}$

$\rightarrow$ Almost uncorrelated with the previous configuration
$\rightarrow$ Ergodic

There is still a sign problem left to address

What if we only want a subset of configurations?

S. Chandrasekharan, U.J. Wiese, Phys.Rev.Lett. 83 (1999) 3116-3119

Outline

1. Cluster Algorithms

2. Correlated Cluster Algorithm for canonical ensembles

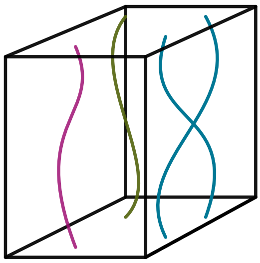
3. Correlated Cluster Algorithm for Abelian Gauge Theories

4. Sign Problems

The canonical ensemble

Fix the number of fermions:

- Flipping cluster i adds n_i fermions



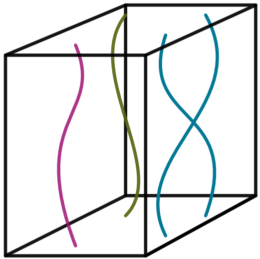
The canonical ensemble

Fix the number of fermions:

- Flipping cluster i adds n_i fermions

- We need

$$\sum_{\text{flipped clusters}} n_i = 0$$



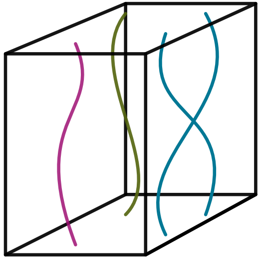
The canonical ensemble

Fix the number of fermions:

- Flipping cluster i adds n_i fermions
- We need

$$\sum_{\text{flipped clusters}} n_i = 0$$

- Each valid set must be generated with equal probability



Restricting to the canonical ensemble

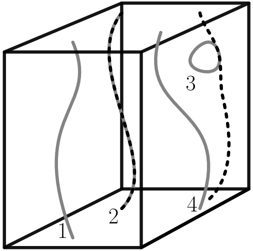
Calculate the number of possibilities iteratively:

$$A_i(N) = A_{i-1}(N) + A_{i-1}(N - n_i)$$

Use them to determine the probabilities to flip:

$$p_i = \frac{A_{i-1}(N_i - n_i)}{A_i(N_i)}$$

with $N_0 = 0$ and $N_i = \begin{cases} N_{i-1} + n_i & \text{if cluster } i \text{ got flipped,} \\ N_{i-1} & \text{otherwise.} \end{cases}$



n_i \ $\sum n_i$	-2	-1	0	1	2
1			1	1	
-1					
0					
1					

Restricting to the canonical ensemble

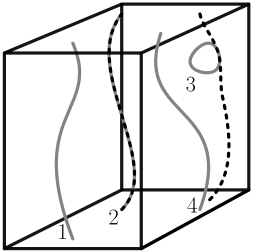
Calculate the number of possibilities iteratively:

$$A_i(N) = A_{i-1}(N) + A_{i-1}(N - n_i)$$

Use them to determine the probabilities to flip:

$$p_i = \frac{A_{i-1}(N_i - n_i)}{A_i(N_i)}$$

with $N_0 = 0$ and $N_i = \begin{cases} N_{i-1} + n_i & \text{if cluster } i \text{ got flipped,} \\ N_{i-1} & \text{otherwise.} \end{cases}$



n_i \ $\sum n_i$	-2	-1	0	1	2
1			1	1	
-1		1	1		
0					
1					

Restricting to the canonical ensemble

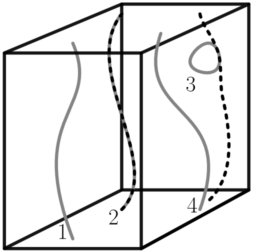
Calculate the number of possibilities iteratively:

$$A_i(N) = A_{i-1}(N) + A_{i-1}(N - n_i)$$

Use them to determine the probabilities to flip:

$$p_i = \frac{A_{i-1}(N_i - n_i)}{A_i(N_i)}$$

with $N_0 = 0$ and $N_i = \begin{cases} N_{i-1} + n_i & \text{if cluster } i \text{ got flipped,} \\ N_{i-1} & \text{otherwise.} \end{cases}$



n_i \ $\sum n_i$	-2	-1	0	1	2
1			1	1	
-1		1	$1+1$	1	
0					
1					

Restricting to the canonical ensemble

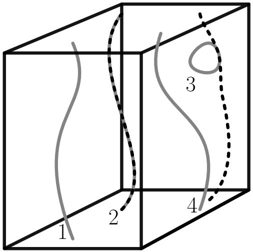
Calculate the number of possibilities iteratively:

$$A_i(N) = A_{i-1}(N) + A_{i-1}(N - n_i)$$

Use them to determine the probabilities to flip:

$$p_i = \frac{A_{i-1}(N_i - n_i)}{A_i(N_i)}$$

with $N_0 = 0$ and $N_i = \begin{cases} N_{i-1} + n_i & \text{if cluster } i \text{ got flipped,} \\ N_{i-1} & \text{otherwise.} \end{cases}$



n_i \ $\sum n_i$	-2	-1	0	1	2
1			1	1	
-1		1	2	1	
0					
1					

Restricting to the canonical ensemble

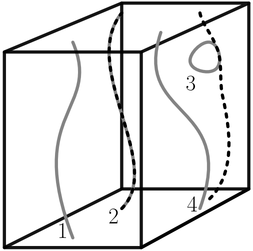
Calculate the number of possibilities iteratively:

$$A_i(N) = A_{i-1}(N) + A_{i-1}(N - n_i)$$

Use them to determine the probabilities to flip:

$$p_i = \frac{A_{i-1}(N_i - n_i)}{A_i(N_i)}$$

with $N_0 = 0$ and $N_i = \begin{cases} N_{i-1} + n_i & \text{if cluster } i \text{ got flipped,} \\ N_{i-1} & \text{otherwise.} \end{cases}$



n_i \ $\sum n_i$	-2	-1	0	1	2
1			1	1	
-1		1	2	1	
0		2	4	2	
1					

Restricting to the canonical ensemble

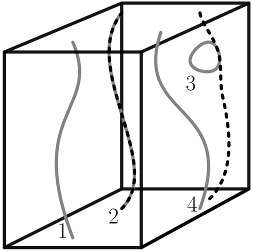
Calculate the number of possibilities iteratively:

$$A_i(N) = A_{i-1}(N) + A_{i-1}(N - n_i)$$

Use them to determine the probabilities to flip:

$$p_i = \frac{A_{i-1}(N_i - n_i)}{A_i(N_i)}$$

with $N_0 = 0$ and $N_i = \begin{cases} N_{i-1} + n_i & \text{if cluster } i \text{ got flipped,} \\ N_{i-1} & \text{otherwise.} \end{cases}$



n_i \ $\sum n_i$	-2	-1	0	1	2
1			1	1	
-1		1	2	1	
0		2	4	2	
1		2	6	6	2

Restricting to the canonical ensemble

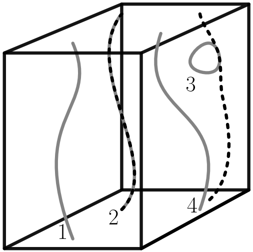
Calculate the number of possibilities iteratively:

$$A_i(N) = A_{i-1}(N) + A_{i-1}(N - n_i)$$

Use them to determine the probabilities to flip:

$$p_i = \frac{A_{i-1}(N_i - n_i)}{A_i(N_i)}$$

with $N_0 = 0$ and $N_i = \begin{cases} N_{i-1} + n_i & \text{if cluster } i \text{ got flipped,} \\ N_{i-1} & \text{otherwise.} \end{cases}$



n_i \ $\sum n_i$	-2	-1	0	1	2
1			1	1	
-1		1	2	1	
0		2	4	2	
1		2	6	6	2

Restricting to the canonical ensemble

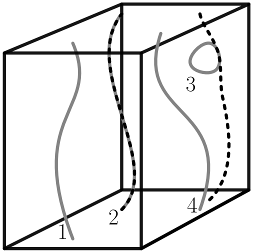
Calculate the number of possibilities iteratively:

$$A_i(N) = A_{i-1}(N) + A_{i-1}(N - n_i)$$

Use them to determine the probabilities to flip:

$$p_i = \frac{A_{i-1}(N_i - n_i)}{A_i(N_i)}$$

with $N_0 = 0$ and $N_i = \begin{cases} N_{i-1} + n_i & \text{if cluster } i \text{ got flipped,} \\ N_{i-1} & \text{otherwise.} \end{cases}$



n_i \ $\sum n_i$	-2	-1	0	1	2
1			1	1	
-1		1	2	1	
0		2	4	2	
1		2	6	6	2

Restricting to the canonical ensemble

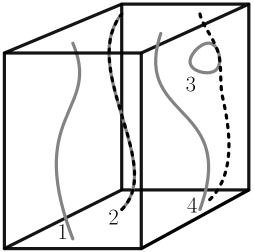
Calculate the number of possibilities iteratively:

$$A_i(N) = A_{i-1}(N) + A_{i-1}(N - n_i)$$

Use them to determine the probabilities to flip:

$$p_i = \frac{A_{i-1}(N_i - n_i)}{A_i(N_i)}$$

with $N_0 = 0$ and $N_i = \begin{cases} N_{i-1} + n_i & \text{if cluster } i \text{ got flipped,} \\ N_{i-1} & \text{otherwise.} \end{cases}$



n_i \ $\sum n_i$	-2	-1	0	1	2
1			1	1	
-1		1	2	1	
0		2	4	2	
1		2	6	6	2

Restricting to the canonical ensemble

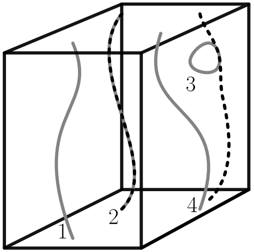
Calculate the number of possibilities iteratively:

$$A_i(N) = A_{i-1}(N) + A_{i-1}(N - n_i)$$

Use them to determine the probabilities to flip:

$$p_i = \frac{A_{i-1}(N_i - n_i)}{A_i(N_i)}$$

with $N_0 = 0$ and $N_i = \begin{cases} N_{i-1} + n_i & \text{if cluster } i \text{ got flipped,} \\ N_{i-1} & \text{otherwise.} \end{cases}$



n_i \ $\sum n_i$	-2	-1	0	1	2
1			1	1	
−1		1	2	1	
0		2	4	2	
1		2	6	6	2

Restricting to the canonical ensemble

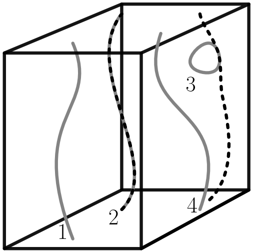
Calculate the number of possibilities iteratively:

$$A_i(N) = A_{i-1}(N) + A_{i-1}(N - n_i)$$

Use them to determine the probabilities to flip:

$$p_i = \frac{A_{i-1}(N_i - n_i)}{A_i(N_i)}$$

with $N_0 = 0$ and $N_i = \begin{cases} N_{i-1} + n_i & \text{if cluster } i \text{ got flipped,} \\ N_{i-1} & \text{otherwise.} \end{cases}$



n_i \ $\sum n_i$	-2	-1	0	1	2
1			1	1	
-1		1	2	1	
0		2	4	2	
1		2	6	6	2

Restricting to the canonical ensemble

Calculate the number of possibilities iteratively:

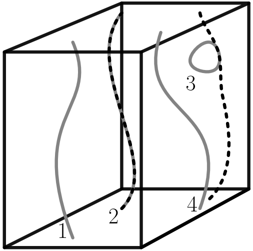
$$A_i(N) = A_{i-1}(N) + A_{i-1}(N - n_i)$$

Use them to determine the probabilities to flip:

$$p_i = \frac{A_{i-1}(N_i - n_i)}{A_i(N_i)}$$

with $N_0 = 0$ and $N_i = \begin{cases} N_{i-1} + n_i & \text{if cluster } i \text{ got flipped,} \\ N_{i-1} & \text{otherwise.} \end{cases}$

Runtime: $\mathcal{O}\left(n_{\text{clusters}} \max(\sum n_i)\right) = \mathcal{O}(\beta V^2)$



n_i \ $\sum n_i$	-2	-1	0	1	2
1			1	1	
-1		1	2	1	
0		2	4	2	
1		2	6	6	2

Outline

The Schwinger Model in the Hamiltonian Formalism

Discretized Schwinger model with a Thirring term and staggered fermions:

$$H = \sum_n \underbrace{-t\left(c_n^\dagger U_n c_{n+1} + h.c.\right) + 2t\left(\hat{n}_n - \frac{1}{2}\right)\left(\hat{n}_{n+1} - \frac{1}{2}\right)}_{\text{gauged spinless fermions}} + \underbrace{m(-1)^n c_n^\dagger c_n}_{\text{mass}} + \underbrace{g\left(E_n + \frac{\theta}{2\pi}\right)^2}_{\text{gauge term with topological } \theta \text{ angle}}$$

Here, U_n is a raising operator for the gauge field E_n living on the links

Tensor Networks: e.g. M.C. Bañuls, K. Cichy, J.I. Cirac and K. Jansen - JHEP11 (2013) 158

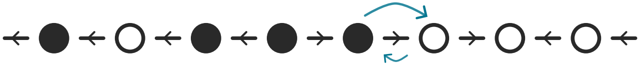
Quantum Simulations: e.g. O. Kaikov, T. Saporiti, V. Sazonov, M. Tamaazousti - arXiv:2407.09224 (2024)

The Schwinger Model in the Hamiltonian Formalism

Discretized Schwinger model with a Thirring term and staggered fermions:

$$H = \sum_n \underbrace{-t\left(c_n^\dagger U_n c_{n+1} + h.c.\right) + 2t\left(\hat{n}_n - \frac{1}{2}\right)\left(\hat{n}_{n+1} - \frac{1}{2}\right)}_{\text{gauged spinless fermions}} + \underbrace{m(-1)^n c_n^\dagger c_n}_{\text{mass}} + \underbrace{g\left(E_n + \frac{\theta}{2\pi}\right)^2}_{\text{gauge term with topological } \theta \text{ angle}}$$

Here, U_n is a raising operator for the gauge field E_n living on the links



Tensor Networks: e.g. M.C. Bañuls, K. Cichy, J.I. Cirac and K. Jansen - JHEP11 (2013) 158

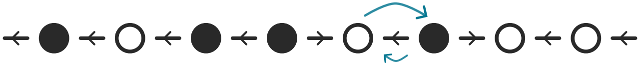
Quantum Simulations: e.g. O. Kaikov, T. Saporiti, V. Sazonov, M. Tamaazousti - arXiv:2407.09224 (2024)

The Schwinger Model in the Hamiltonian Formalism

Discretized Schwinger model with a Thirring term and staggered fermions:

$$H = \sum_n \underbrace{-t\left(c_n^\dagger U_n c_{n+1} + h.c.\right) + 2t\left(\hat{n}_n - \frac{1}{2}\right)\left(\hat{n}_{n+1} - \frac{1}{2}\right)}_{\text{gauged spinless fermions}} + \underbrace{m(-1)^n c_n^\dagger c_n}_{\text{mass}} + \underbrace{g\left(E_n + \frac{\theta}{2\pi}\right)^2}_{\text{gauge term with topological } \theta \text{ angle}}$$

Here, U_n is a raising operator for the gauge field E_n living on the links



Tensor Networks: e.g. M.C. Bañuls, K. Cichy, J.I. Cirac and K. Jansen - JHEP11 (2013) 158

Quantum Simulations: e.g. O. Kaikov, T. Saporiti, V. Sazonov, M. Tamaazousti - arXiv:2407.09224 (2024)

The Schwinger Model in the Hamiltonian Formalism

This has an Abelian local symmetry $[G_n, H] = 0$

$$G_n = \underbrace{c_n^\dagger c_n + (1 - (-1)^n)/2}_{\text{charge } \rho_n} - \underbrace{E_n - E_{n-1}}_{\nabla E_n} = \rho_n - \nabla E_n$$

We focus on the sector $G_n \left|\psi\right\rangle = 0$, so states that obey Gauss's law

The Schwinger Model in the Hamiltonian Formalism

This has an Abelian local symmetry $[G_n, H] = 0$

$$G_n = \underbrace{c_n^\dagger c_n + (1 - (-1)^n)/2}_{\text{charge } \rho_n} - \underbrace{E_n - E_{n-1}}_{\nabla E_n} = \rho_n - \nabla E_n$$

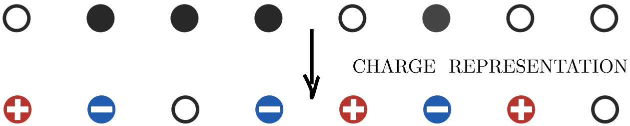
We focus on the sector $G_n \left|\psi\right\rangle = 0$, so states that obey Gauss's law

The Schwinger Model in the Hamiltonian Formalism

This has an Abelian local symmetry $[G_n, H] = 0$

$$G_n = \underbrace{c_n^\dagger c_n + (1 - (-1)^n)/2}_{\text{charge } \rho_n} - \underbrace{E_n - E_{n-1}}_{\nabla E_n} = \rho_n - \nabla E_n$$

We focus on the sector $G_n |\psi\rangle = 0$, so states that obey Gauss's law

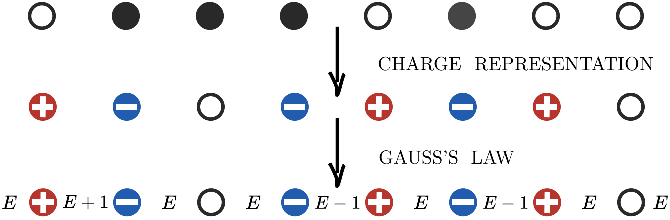


The Schwinger Model in the Hamiltonian Formalism

This has an Abelian local symmetry $[G_n, H] = 0$

$$G_n = \underbrace{c_n^\dagger c_n + (1-(-1)^n)/2}_{\text{charge } \rho_n} - \underbrace{E_n - E_{n-1}}_{\nabla E_n} = \rho_n - \nabla E_n$$

We focus on the sector $G_n |\psi\rangle = 0$, so states that obey Gauss's law



A Cluster Algorithm for the Schwinger Model

Idea: Sample QLMs as a subset of Meron Cluster Algorithm configurations

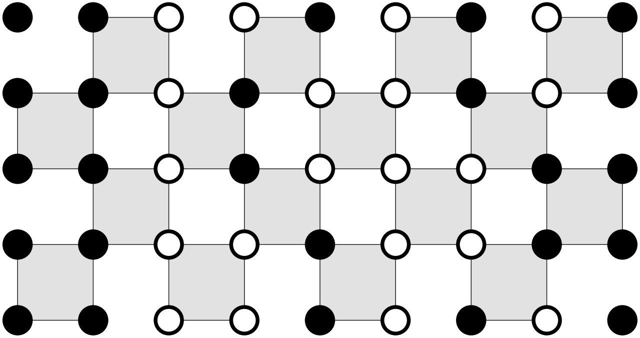
Alternative Cluster Algorithm approach with emergent gauge symmetry:

J. Frank, E. Huffman, S. Chandrasekharan - Physics Letters B (2020)

D. Banerjee, and Emilie Huffman - Phys. Rev. D 109, L031506 (2024)

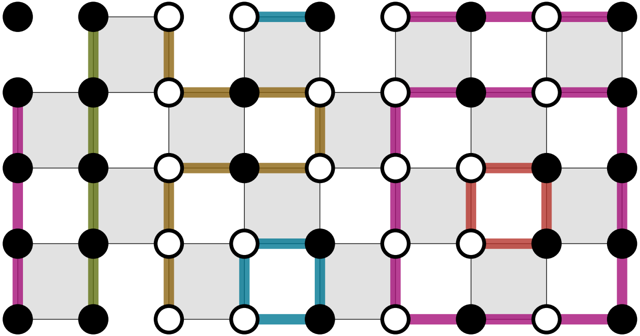
Mapping Fermion Paths to QLMs

Take Fermion Configuration



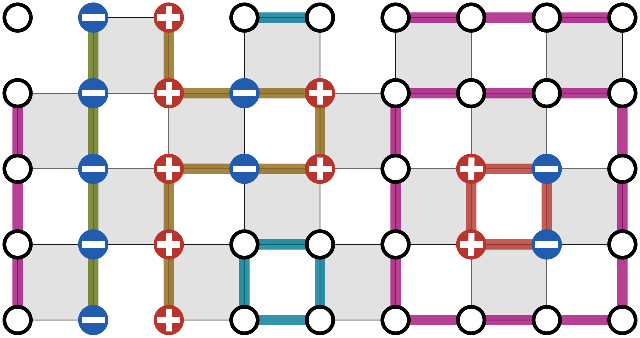
Mapping Fermion Paths to QLMs

Place break-ups as usual



Mapping Fermion Paths to QLMs

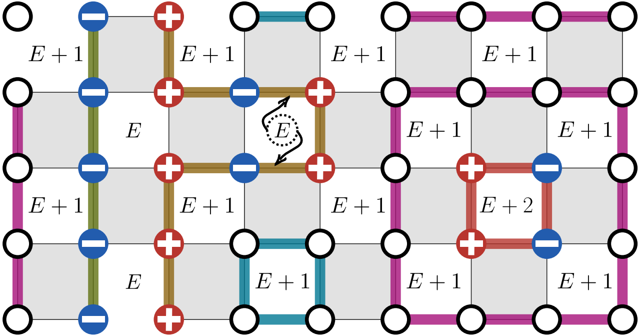
Translate to charge representation



⇒ Clusters flip between all charged and neutral

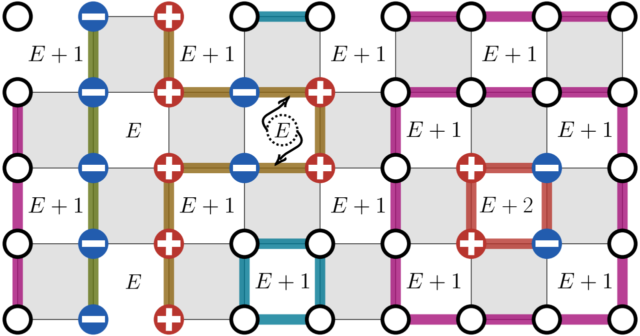
Mapping Fermion Paths to QLMs

Calculate links with Gauss's law



Mapping Fermion Paths to QLMs

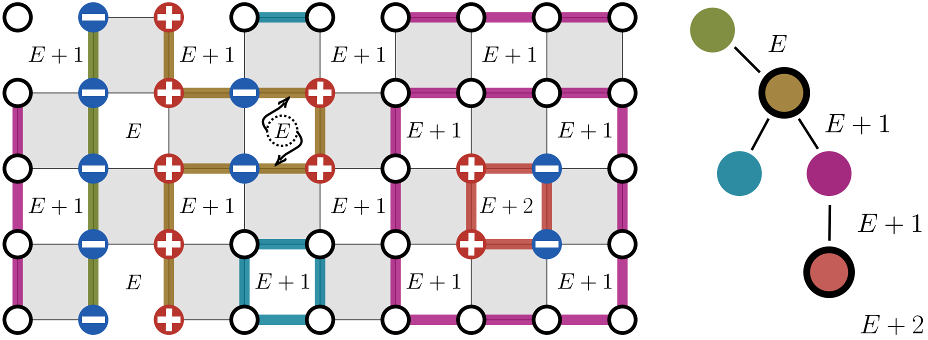
Calculate links with Gauss's law



$\Rightarrow$ Fields only change when crossing a charged cluster

Mapping Fermion Paths to QLMs

Order neighboring clusters into a tree



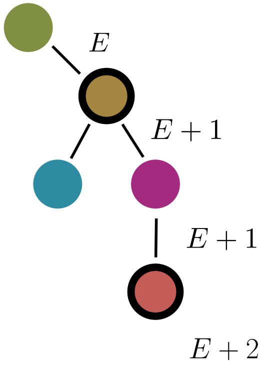
Use Correlated Cluster Algorithm

1. Count number of ways to get each field value L from bottom up

$$A_i(L) = \sum_{j \in \mathcal{C}_i} \left(A_j(L) + A_j(L - l_i) \right)$$

2. Calculate flip probabilities from the top down

$$p_i = \frac{\sum_{j \in \mathcal{C}_i} A_j(L - l_i)}{A_i(L_i)}$$



Local terms in the Hamiltonian can also be added

$$A_i(L_i) = a_i\left(L_i\right) \sum_{j \in G(i)} A_j\left(L_i\right) + a_i'\left(L_i\right) \sum_{j \in G(i)} A_j\left(L_i - l_i\right)$$

Simulating the Schwinger Model

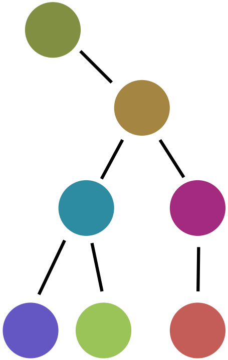
- Possible only if we have a finite number of field values
- Different formulations (QLM, TLM, $\mathbb{Z}_n$) are possible
- For S possible link values, this is scales with $\mathcal{O}(L\beta S^2)$

  $\rightarrow$ Efficient algorithm even with a θ-term

Generalizations of this approach

The prerequisites for this method are:

- The configurations can be mapped onto a subset of an existing cluster algorithm
- The dependence of the clusters is non-cyclic



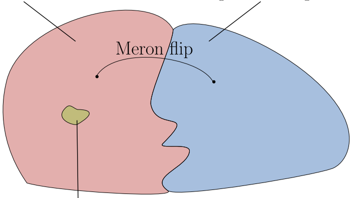
Outline

1. Cluster Algorithms
2. Correlated Cluster Algorithm for canonical ensembles
3. Correlated Cluster Algorithm for Abelian Gauge Theories
4. Sign Problems

Solving Sign Problems with Meron Cluster Algorithms

The Meron Cluster Algorithm solves the fermionic sign problem:

- Flipping certain clusters (Merons) changes the sign of the configuration
- All configurations with a Meron cancel

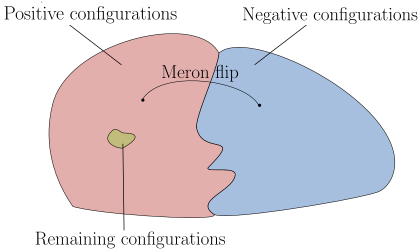


S. Chandrasekharan, U.J. Wiese, Phys.Rev.Lett. 83 (1999) 3116-3119

Solving Sign Problems with Meron Cluster Algorithms

The Meron Cluster Algorithm solves the fermionic sign problem:

- Flipping certain clusters (Merons) changes the sign of the configuration
- All configurations with a Meron cancel

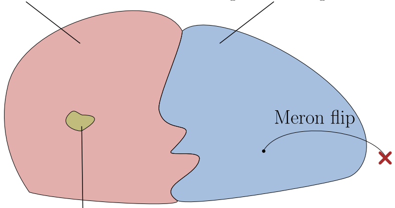


$\Rightarrow$ Avoid Merons with accept/reject step when updating the break-ups

S. Chandrasekharan, U.J. Wiese, Phys.Rev.Lett. 83 (1999) 3116-3119

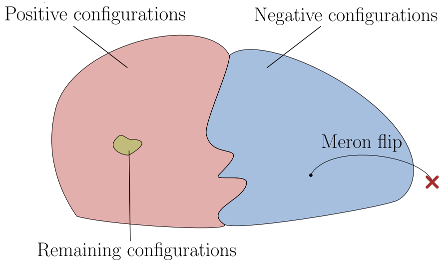
Sign problems in Correlated Cluster Algorithms

The configuration where the meron is flipped could now be illegal



Sign problems in Correlated Cluster Algorithms

The configuration where the meron is flipped could now be illegal



- Use improved estimator for the sign: $\bar{\sigma} = \frac{1}{N} \sum_{\text{legal flips}} \sigma$
- Will likely help a lot, but it could still give zero or negative contributions

Sign problems in Correlated Cluster Algorithms

- This formulation of (1+1)d Abelian gauge theories has no severe sign problem
- Open question: sign problems that occur when studying canonical ensembles in higher dimensions

Conclusion

- Correlated Cluster Algorithms extend the applicability of cluster algorithms
- They can be used to simulate Abelian gauge theories in $1+1$d and canonical ensembles
- They can potentially solve new sign problems

Conclusion

- Correlated Cluster Algorithms extend the applicability of cluster algorithms
- They can be used to simulate Abelian gauge theories in $1+1$d and canonical ensembles
- They can potentially solve new sign problems

Marina Krstić Marinković

Joao Pinto Barros

Thank you