Algorithms for lattice QCD I

Stefan Schaefer

NIC, DESY

Kolkata Lattice Gauge Theory School

Algorithms in Lattice QCD

Definition Wilson'74

Using a discrete space-time QCD can be defined.

Computational method

Path integral $\rightarrow$ high dimensional integral

Computation of integrals by Monte Carlo.

- E.g. hadron masses, decay constants, ...

Algorithms

Many calculations computationally very expensive.

- Massively parallel computers employed.

Need to work on simulation setup, algorithms,...

- Many choices.
- Physics understanding $\leftrightarrow$ Numerical setup.

Algorithms in Lattice QCD

Field update

Integral over field variables $\rightarrow$ **sum** over field configurations

- How to efficiently generate these fields.
- Transformation of action into form amenable to numerical treatment.
- Numerical solution of diff. eq.

Solution of Dirac equation

Needed in field update and fermionic observables.

Out-of-the-box algorithms, e.g., the conjugate gradient, perform badly
$\rightarrow$ Need to take physics into account

Outline

Monday

Introduction

Markov Chain Monte Carlo

The HMC algorithm

Tuesday

MD integrators

Fermions in QCD simulations

Mass preconditioning

Outline

Wednesday

Other fermion methods (RHMC, DD-HMC)

Solution of the Dirac equation

Thursday

Solution of the Dirac equation II
Local deflation

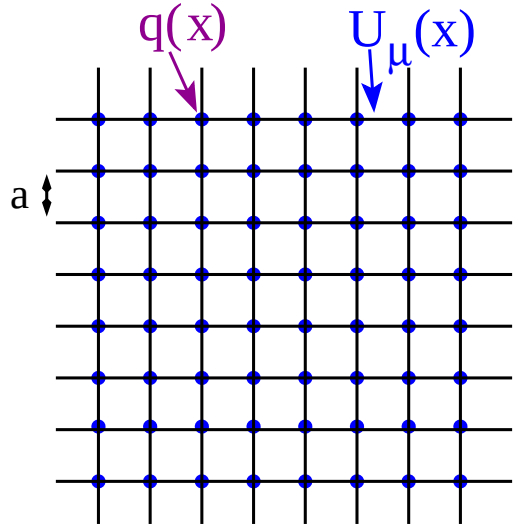
Methods to compute hadron observables

Friday

Autcorrelations

General reading:
M. Lüscher
Computational strategies in lattice QCD
Les Houches 2009

Goal



Computation of path integral

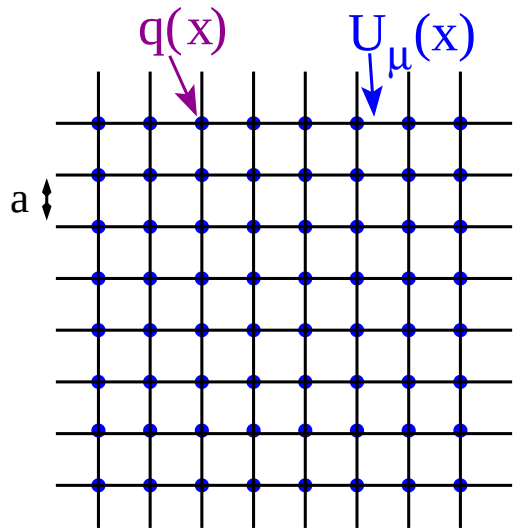
$$\langle A \rangle = \frac{1}{Z} \int \prod_{x,\mu} dU_{x,\mu} e^{-S[U]} \, A[U]$$

with

$$Z = \prod_{x,\mu} dU_{x,\mu} e^{-S[U]}$$

One SU(3) integration variable for each link.

Goal



$$\langle A \rangle = \frac{1}{Z} \int \prod_{x,\mu} dU_{x,\mu} e^{-S[U]} \, A[U]$$

One SU(3) integration variable for each link.

128×64^3 lattice $\rightarrow 1.3 \cdot 10^8$ links

Classical numerical quadrature would need $N^{\#\text{variables}}$ function evaluations

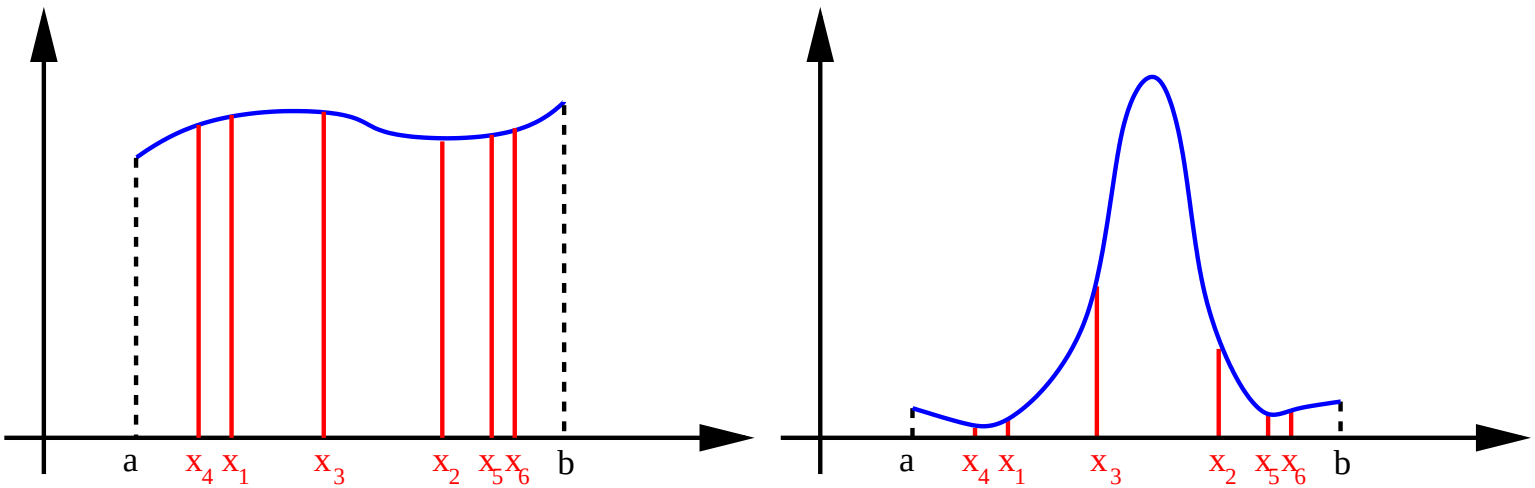
Monte Carlo

General idea of Monte-Carlo integration

$$\frac{1}{b-a}\int_a^b \mathrm{d}x\, f(x) \approx \frac{1}{N}\sum_{i=1}^{N} f(x_i)$$

with randomly chosen points x_i in the integration region

- good idea, if $f(x)$ approximately constant
  $\Rightarrow$ small fluctuations in $f(x_i)$.
- For a given realization of the N points x_i, this is an **unbiased estimator** of the integral



Error of a MC simulation

$$\tilde{F}_j = \frac{1}{N}\sum_{i=1}^{N} f(x_i^j)$$

Index j labels the repetition of the "experiment".

Unbiased = gives correct result *on average*

$$F = \langle\langle \tilde{F} \rangle\rangle$$

$\langle\langle \cdot \rangle\rangle$ average over realizations of the x_i

Typical deviation $\rightarrow$variance *of this estimator*.

$$\begin{aligned}\langle\langle (\tilde{F} - F)^2 \rangle\rangle &= \frac{1}{N^2}\sum_{ik} \langle\langle F_i F_k \rangle\rangle - F^2 \\ &= \frac{1}{N}(\sum_{i} \langle\langle \tilde{(F_i)}^2 \rangle\rangle - F^2) = \mathbf{var}(F)/N\end{aligned}$$

Error decreases as $1/\sqrt{N}$

Note of caution

This theoretical analysis assumes knowlege of two quanitites

$$\bar{F} = \int dx\, f(x) \qquad \text{and} \qquad \mathrm{var(f)} = \int dx\, (f(x) - \bar{f})^2$$

What you get from the Monte Carlo are **estimators** of these quantities.

The analysis is correct for $N \to \infty$; ∞ is a large number.

These estimators might have significant errors, which are hard to get from the MC. You might also just have been unlucky.

To a certain extend, practical Monte Carlo is an art and requires careful inspection of the results.

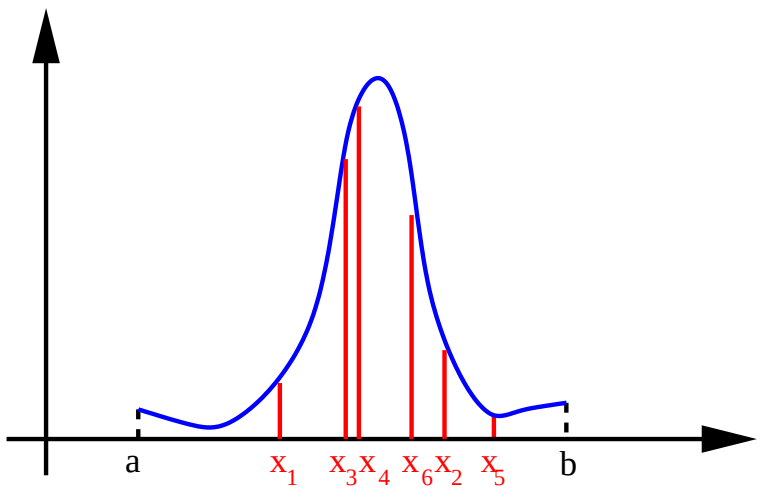
Importance sampling

Estimator correct up to $\sqrt{\mathrm{var}(f)/N} \rightarrow$ reduce variance.

$$\int \mathrm{d}x\, f(x) = \int \rho(x)\mathrm{d}x\, \frac{f(x)}{\rho(x)} = \left[\frac{1}{N}\sum_{i=1}^{N} \frac{f(x_i)}{\rho(x_i)}\right] (1 + \mathcal{O}(N^{-1/2}))$$

with points x_i chosen according to ρ.

- Choose points accoring to probablity distribution similar to function to be integrated
- optimal, if distribution $\propto |f(x)|$ ...need $\int dx|f(x)|$



Markov Chain Monte Carlo

Problem of "straight" Monte Carlo is to find a *normalized* probability density $P(x)$

Solution

Use a method which only needs **relative** probabilites

Construct a sequence of points

$$x_1 \to x_2 \to x_3 \to \cdots \to x_N$$

using a transition probability $T(x_{i+1} \leftarrow x_i)$

Analysis using arguments of a N sets of such chains.

Properties of T

For any given pair of points x_1 and x_2

$$T(x_2 \leftarrow x_1)$$

with the following properties

(A) **Stability**

$$P(x') = \int dx\, P(x)\, T(x' \leftarrow x)$$

(B) **Normalization**

$$\int dx'\, T(x' \leftarrow x) = 1$$

(C) **Ergodicity**

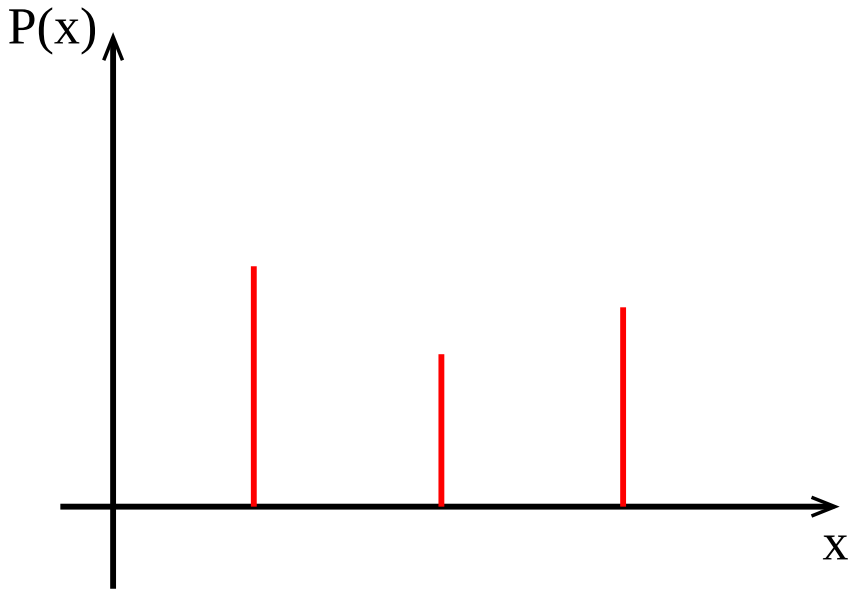
$T(x' \leftarrow x) > 0$ for each pair x, x'

No reference to absolute normalization of P

Analysis of MCMC

For the sake of simplicity, consider discrete state space.
Integrals $\longrightarrow$ sums.

Example for a single variable with three possible values:



$T(x' \leftarrow x)$ is a matrix acting in the space of states.
For the example a 3×3 matrix, because x can take 3 values.

A probability distribution is a normalized vector in this space.

Analysis of MCMC

Imagine having an ensemble of points x_i distributed according to $P(x)$.

Condition (A) reads

$$P = T\,P$$

$\rightarrow P$ is eigenvector of T with eigenvalue 1.

Theorem of Frobenius–Perron

For a matrix with the properties A–C the following holds

- There is exactly one eigenvalue $\lambda = 1$.
- All eigenvalues λ have $|\lambda| \leq 1$.

Convergence of Markov Process

Given any starting distribution P_0 repeated application of T leads to exponential convergence to desired distribution

$$\begin{aligned} T^n P_0 &= \sum_{i=0}^{N} \lambda_i^n \left(\Psi_i, P_0\right) \Psi_i \\ &= \sum_{i=0}^{N} e^{\log |\lambda_i| n} \left(\Psi_i, P_0\right) \frac{\lambda_i}{|\lambda_i|} \Psi_i \\ &\propto P + \mathrm{O}(e^{-n/\tau_1}) \end{aligned}$$

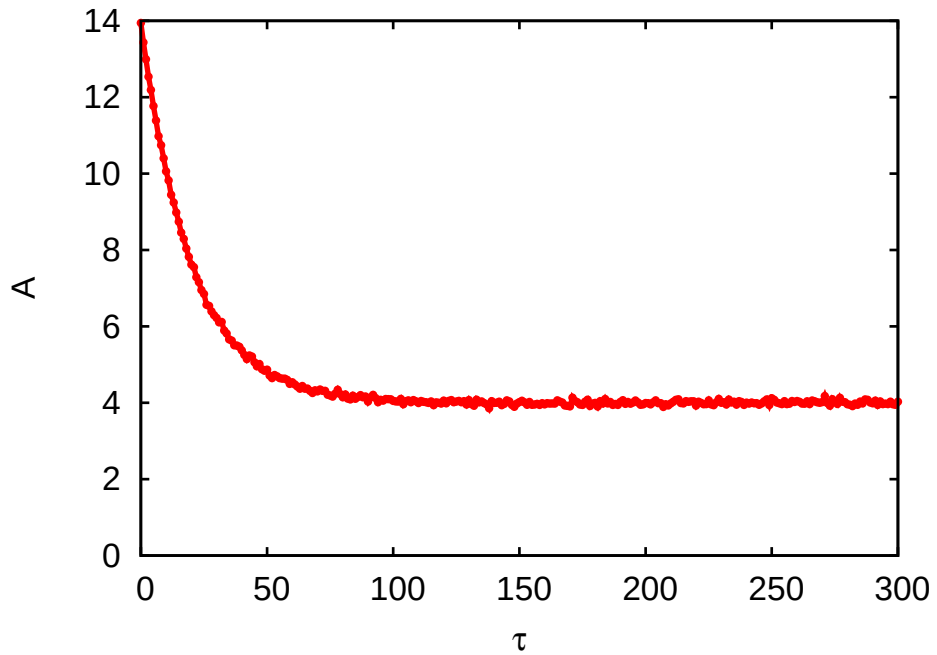
$\tau_i = 1/\log|\lambda_i|$ are the exponential autocorrelation times

A nicer interpretation in terms of single exponentials can be given with detailed balance, see later.

Practical MCMC

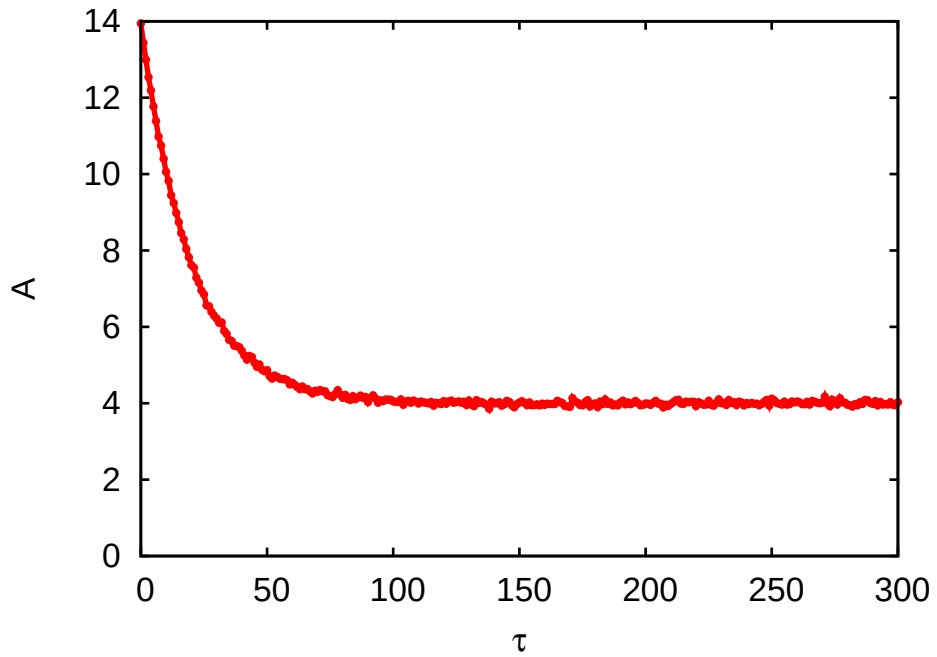
Start with one (or a few) points
Averages of Monte Carlo time

$$\langle f \rangle = \frac{1}{N} \sum_{i=1}^{N} f(x_i)$$



Cut away first *thermalization* phase.

Time evolution



$$\langle f \rangle = \frac{1}{N} \sum_{t=1}^{N} f(x_i)(1 + \mathcal{O}(1/\sqrt{N}))$$

At the beginning large contributions from $\Psi_{i>0}$.
Formally, they are supressed by $1/N$, but they are best not considered.
A valid simulation must have $n \gg 1/\tau_1$

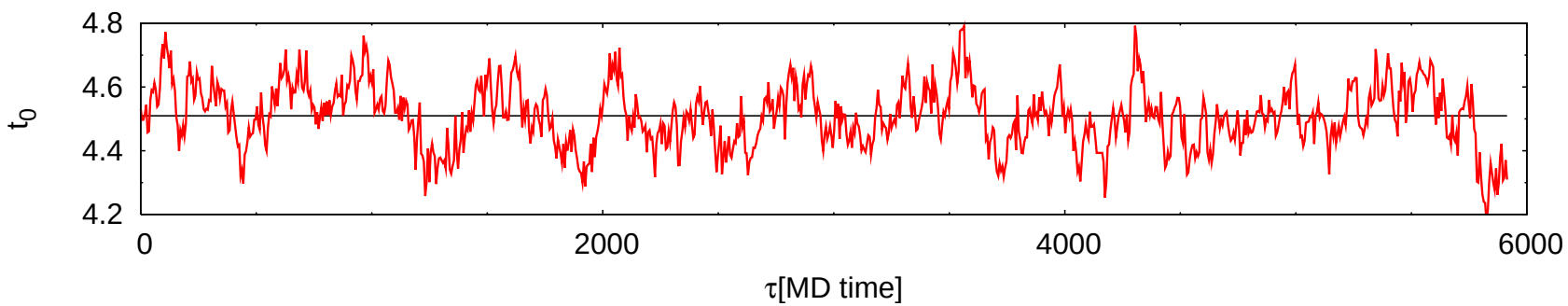
Autocorrelations

There is a price to pay:
Subsequent points x_i/configurations are in general not independent.

$$\langle\langle f(x_i) f(x_{i+1})\rangle\rangle \neq 0$$

This needs to be taken into account in the *analysis* of the data.

It also can make simulations exceedingly expensive.



Construction of T

Constructing a valid T seems a daunting task.

Metropolis–Hastings method Metropolis et al'53

- Symmetric proposal (trans. prob. w/ $P_0(x)$ =const)

$$W(x' \leftarrow x) = W(x \leftarrow x')$$

- Acceptance step

$$P_{acc}(x', x) = \min[1, \frac{P(x')}{P(x)}]$$

Next point is x' with propability P_{acc}, else x

Total transition propability

$$T(x' \leftarrow x) = W(x' \leftarrow x)P_{acc}(x', x) + \delta_{xx'} \sum_{\tilde{x}} (1 - P_{acc}(\tilde{x}, x))W(\tilde{x} \leftarrow x)$$

Proof

$$T(x' \leftarrow x) = W(x' \leftarrow x)P_{acc}(x', x) + \delta_{xx'}[\sum_{\tilde{x}}(1 - P_{acc}(\tilde{x}, x))W(\tilde{x} \leftarrow x)]$$

(A) **Stability**

$$\sum_x P(x)\, T(x' \leftarrow x)$$

$$= \sum_x W(x' \leftarrow x)P(x)P_{\text{acc}}(x', x) + P(x') \sum_{\tilde{x}}(1 - P_{acc}(\tilde{x}, x))W(\tilde{x} \leftarrow x)]$$

$$= \sum_x W(x \leftarrow x')P(x')P_{\text{acc}}(x, x') + P(x')(1 - \sum_{\tilde{x}} P_{acc}(\tilde{x}, x)W(\tilde{x} \leftarrow x)] = P(x')$$

(B) **Normalization**

$$\sum_{x'} T(x' \leftarrow x) = 1$$

Practical implementation

We want to generate the sequence

$$x_1 \to x_2 \to x_3 \to \cdots$$

and arrived at x_j

- Make a proposal y according to probability $W(y \leftarrow x_j)$
- Compute P_{acc} and draw random number $0 \leq r < 1$

$$x_{j+1} = \begin{cases} y & \text{if } r < P_{\text{acc}} \\ x_j & \text{else} \end{cases}$$

Good proposal
- Easy to generate
- High acceptance propability

Generic issues of the method

The step size can be rather limited

Small steps

$P(x') \approx P(x) \rightarrow P_{\mathrm{acc}}$ high
But it take many steps to sample the whole integration space
Large autocorrelations

Large steps

Small autocorrelations
Feasible only if an update leading to reasonable P_{acc} can be found
Can be difficult to compute proposal.

In high dimensional spaces, it can be very difficult to argue, what "large step" and "small step" even mean.

Summary of part I

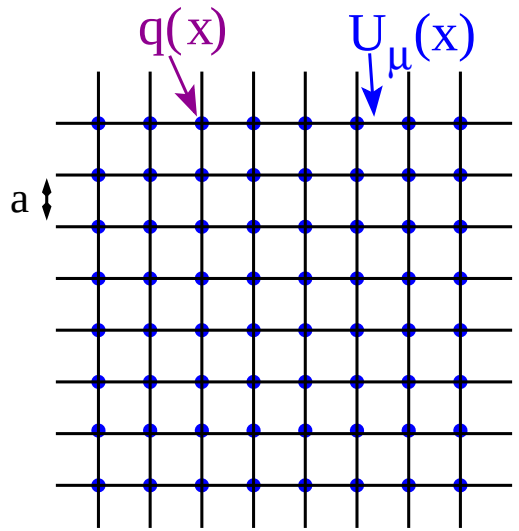
Monte Carlo is a method to numerically compute high-dimensional integrals.

Integral $\rightarrow$ average over sample points.

Sample points need to be chosen in region of high probability.

Problem of constructing a normalized distribution
$\rightarrow$ Markov Chain Monte Carlo
No normalized distribution needed.
Deal with autocorrelations.

Back to QCD



One point $x \rightarrow$ one value for each link variable
$\rightarrow$ one field configuration

Instead of x we will therefore now use U which are in SU(3).

$$\langle A \rangle = \frac{1}{Z} \int \prod_{x,\mu} dU_{x,\mu} e^{-S[U]} \, A[U]$$

with

$$Z = \prod_{x,\mu} dU_{x,\mu} e^{-S[U]}$$

Normalized probability density

$$P[U] = \frac{1}{Z} e^{-S[U]}$$

Ratios in probabilities

$\rightarrow$

Need to evaluate *differences* in the action $S[U]$

While this choice of $P[U]$ seems natural, it is not unique

Basically two types of algorithms

Single link updates

Of the $4V$ links, only one is changed at a time.

In each step $S[U] - S[U']$
Possible if this is an $\mathrm{O}(1)$ operation.

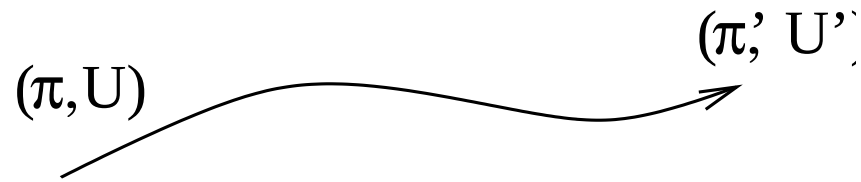
With dynamical fermions, estimating change in action is a global $\mathrm{O}(V)$ operation

Method of choice in pure gauge theory.
No systematic study.

Basically two types of algorithms

Updates based on Molecular Dynamics

Based on ideas from classical mechanics.



Field configuration position
Introduce momenta $\rightarrow$ equations of motion.

Updates keep propbability constant (micro-canonical)

Solves problem of finding a good proposal in Metropolis-Hastings procedure

Extended field space

$$Z = \int [dU][d\pi]\, e^{-\frac{1}{2}(\pi,\pi) - S[U]}$$

Expectation values of observables $A[U]$
remain the same.

Momenta $\pi = \pi^a T^a \in \mathbf{su}(N)$, $\pi^a \in \mathbb{R}$

$$(\pi, \pi) = \sum_{x,\mu} \pi^a_{x,\mu} \pi^a_{x,\mu}$$

Updates

Make updates in this extended phase space.
$\Rightarrow$ updates for U fields.

Molecular dynamics

Essential update step for the gauge fields: $(\pi, U) \to (\pi', U')$

Hamilton's equations of motion

Hamiltonian

$$H[\pi, U] = \frac{1}{2}(\pi, \pi) + S[U]$$

E.o.m.

$$\dot{U}_{x,\mu} = \pi_{x,\mu} U_{x,\mu}$$

$$\dot{\pi}_{x,\mu} = -F_{x,\mu} \ , \qquad F^a_{x,\mu} = \frac{\partial S(e^\omega U)}{\partial \omega^a(x,\mu)}$$

Molecular dynamics

By **Liouville's theorem**, the classical dynamics

$$(\pi, U) \to (\pi', U')$$

maps areas of equal likelihood into eachother.

Energy conservation

$$\frac{d}{d\tau} H = 0$$

Boltzmann factor e^{-H} is constant.

Conservation of phase space

An exact solution of the E.o.m is a valid update.

Fundamentally different from Metropolis–Hastings

In practice: integration errors

Momentum Heatbath

Refresh momenta π (Gaussian random numbers)

Molecular Dynamics

Solve numerically MD equations for some MC time τ (trajectory) deriving from Hamiltonian $H = \frac{1}{2}(\pi, \pi) + S[U]$.

Acceptance Step

Correcting for inaccuracies in integration.

Metropolis

Different viewpoint:

The molecular dynamics as the symmetric proposal in Metropolis.

This solves the problem of the inexact integration.

Need a **symplectic integrator**, i.e. area conserving and reversible.

Reversibility haunts computer implementations.
No good theory for this.

Acceptance step

Molecular dynamics $(\pi, U) \to (\bar{\pi}, \bar{U})$

$$P_{\text{acc}} = \min(1, e^{-(H(\bar{\pi},\bar{U}) - H(\pi,U))})$$

Exact solution of MD equations has $\Delta H = 0$
$\Rightarrow$ always accepted

ΔH needs to be $O(1)$ for good acceptance.
Difficult to achieve on large volume.

Updates

$$Z = \int [dU][d\pi]\, e^{-\frac{1}{2}(\pi,\pi) - S[U]}$$

Momenta: Heatbath

- $(\pi, \pi) = \sum_{x,\mu} |\pi^a_{x,\mu}|^2$
  π^a are Gaussian random numbers.
- Normalization is known.

Just for π this is the optimal update
no correlation to previous config.
Also true for combined system? (Kramers' rule,...)

Comments

Momentum heat-bath is the only source of randomness.

Makes algorithm ergodic.

Problems with ergodicity from $S = \infty$ surfaces.

The original hope was that a trajectory consititutes a macroscopic update.

Free field theory

$$\ddot{U} = \dot{\pi} = \frac{\delta S}{\delta U}$$

$\rightarrow$ Monte Carlo Time $\tau \propto 1/a$

In the classic algorithm trajectory length is scaled with $1/a$.

In number of updates, autocorrelations should stay the same.

M.Lüscher, S.S.'11

This does not apply in interacting theory
$\rightarrow$ Wednesday

Virotta,Sommer, S.S.'10

Still, rather long trajectories are a good idea, $\tau \sim 2$.

For exercises in ϕ^4 theory: S.S., Les Houches 2009, available at NIC@DESY website.