Lecture#2
Hybrid Monte Carlo (HMC) algorithm

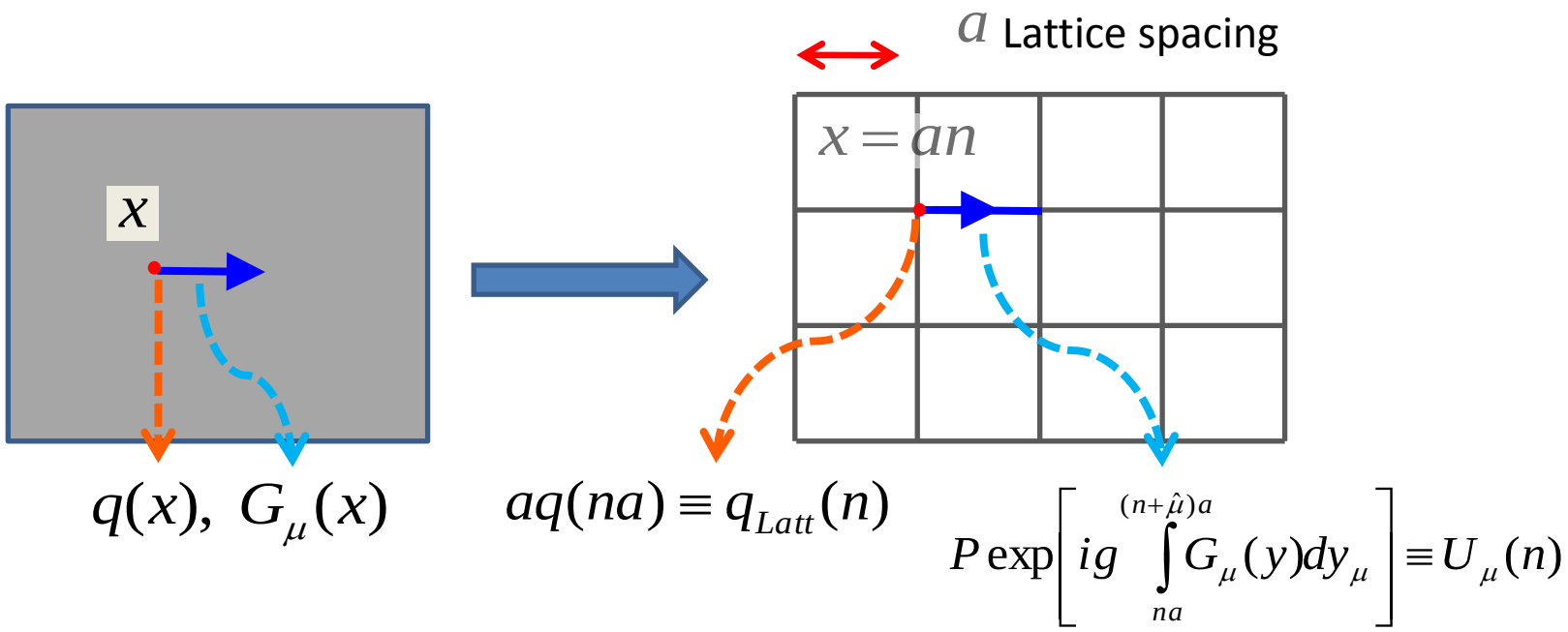
Ken-Ichi Ishikawa

Hiroshima Univ.

1. HMC in Lattice QCD

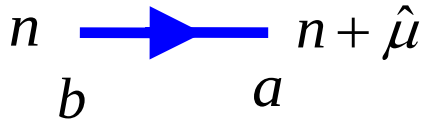
1-1 LQCD

- Gluon and quark fields are defined on a lattice.
- Quarks are described with fermionic spinor fields with color d.o.f.
- Gluons are described with bosonic color matrix fileds.

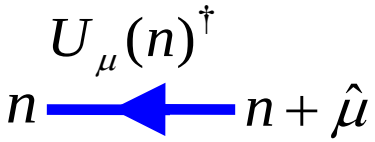


- Euclidean path-integral partition function

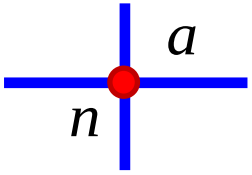
$$U_\mu(n) = \left(U_\mu(n)^{ab}\right)$$



Link field. SU(3) 3x3 unitary matrix.
Lattice vector field. Connects site n and $n+\hat{\mu}$.



$$q(n) = \left(q(n)_a\right)$$



Quark field. SU(3) fundamental rep.
lattice spinor field. Resides on site n.
This may have flavor index *f* by introducing several quarks. (u,d,c,s,t,b)

$$Z[V,\bar{\eta},\eta] = \int \prod_n d\bar{q}(n) dq(n) \prod_{n,\mu} dU_\mu(n) \exp\left[-S_{LQCD}[U,\bar{q},q] + V.U + \bar{q}.\eta + \bar{\eta}.q\right]$$

Local Gauge transformation.

$$q'(n) = \Omega(n) q(n)$$

$$U'_\mu(n) = \Omega(n) U_\mu(n) \Omega(n+\hat{\mu})$$

Local gauge transformation SU(3) group matrix.

$$\Omega(n)$$

Integration Measure and the action are defined to be gauge invariant.
SU(3) group invariant Harr measure for d*U*.

- Typical LQCD action. (K.G.Wilson 1974)

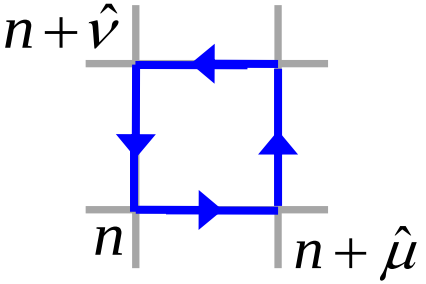
$$S_{LQCD}[U,\bar{q},q] = S_G[U] + S_Q[U,\bar{q},q]$$

$$S_G[U] = \beta \sum_n \sum_{\mu>\nu} \left(1 - \frac{1}{2N_c} \mathrm{Tr}\left[P_{\mu\nu}(n) + P_{\mu\nu}(n)^\dagger \right] \right)$$

Wilson Gauge action. (Gluon)

$$P_{\mu\nu}(n) \equiv U_\mu(n) U_\nu(n+\hat{\mu}) U_\mu(n+\hat{\nu})^\dagger U_\nu{}^\dagger(n)$$

Plaquette *P*
Field strength



$$\beta = \frac{2N_c}{g^2} = \frac{6}{g^2}$$

Inverse bare gauge coupling.

$$S_Q[U,\bar{q},q] = \sum_{f=u,d,s} \sum_{n,m} \bar{q}_f(n) D_f[U](n,m) \bar{q}_f(m)$$

Wilson Fermion action. (qaurks)

$$D_f[U](n,m) = \delta_{n,m} - \kappa_f \sum_{\mu=1}^{4} \left[(1-\gamma_\mu) U_\mu(n) \delta_{n+\hat{\mu},m} + (1+\gamma_\mu) U_\mu(n-\hat{\mu})^\dagger \delta_{n-\hat{\mu},m} \right]$$

Wilson-Dirac operator *D*.
Contains gauge covariant difference op.

$$\kappa_f = \frac{1}{2(am_f + 4)}$$

Hopping parameter κ
No chiral symmetry.

- Observables in LQCD
  - Weighted averaging for Hadronic opertors $O[U,\bar{q},q]$

$$\langle O[U,\bar{q},q]\rangle = \frac{1}{Z[0]}\int\prod_n d\bar{q}(n)dq(n)\prod_{n,\mu} dU_\mu(n) O[U,\bar{q},q]\exp\left[-S_{LQCD}[U,\bar{q},q]\right]$$

  - This integral contains Grassmann variables *q*. This cannot be treated numerically on computers.
  - Analytic integration on Grassmann variables *q* is possible when the action is bilinear of *q*. This is OK for LQCD.

$$S_Q[U,\bar{q},q] = \bar{q}\cdot D[U]\cdot q$$

$$\langle O[U,D[U]^{-1}]\rangle = \frac{1}{Z[0]}\int\prod_{n,\mu} dU_\mu(n) O[U,D[U]^{-1}]\prod_{f=u,d,s}\det[D_f[U]]\exp\left[-S_G[U]\right]$$

  - $q\bar{q}$ operators in the observables are all replaced to corresponding quark propagators $\left(D[U]\right)^{-1}$ [= inverse of Dirac operators].

$$\left\langle O[U, D[U]^{-1}] \right\rangle = \frac{1}{Z[0]} \int \prod_{n,\mu} dU_\mu(n) O[U, D[U]^{-1}] \prod_{f=u,d,s} \det[D_f[U]] \exp\left[-S_G[U]\right]$$

- Thus the average is obtained by integration on $\{U_\mu(n)\}$ with the weight

$$W[U] = Const \prod_{f=u,d,s} \det[D_f[U]] \exp\left[-S_G[U]\right]$$

- This weight is constructed to be real and non-negative. With the Wilson gauge action the exp weight is real and non-negative.
- The determinant of the Lattice Dirac operator appears in the weight. The determinant is also real in the standard setup. However it causes difficulty with negative/complexe values in some cases (finite density simulation).
- For Wilson Dirac fermions the determinant is not protected from negative value. However sufficiently large quark masses it is positive real.
- What we have to do is MCMC simulation with the weight W[U].

1-2 MCMC for the continuous state/probablity case.

- As seen before we need to generate ensemble for {U} with the weight

$$W[U] = Const \prod_{f=u,d,s} \det[D_f[U]] \exp\left[-S_G[U]\right]$$

- More precisely the probability density

$$W[U]\prod_{n,\mu} dU_\mu(n) = \text{Const} \times \left(\prod_{f=u,d,s} \det[D_f[U]] \exp\left[-S_G[U]\right] \right) \prod_{n,\mu} dU_\mu(n)$$

- Hybrid Monte Carlo (HMC) algorithm is used for this purpose.

$$\left\{U^{(1)}, U^{(2)}, U^{(3)}, \cdots, U^{(N)}\right\} \qquad \left\langle O[U, D[U]^{-1}] \right\rangle \approx \frac{1}{N} \sum_{j=1}^{N} O[U^{(j)}, D[U^{(j)}]^{-1}]$$

- To explain the basics on the HMC algorithm, we simplify the notation by introducing corrective real scalar coordinates $\vec{\phi} \leftarrow \{U\}$.

$$\left\{\vec{\phi}^{(1)}, \vec{\phi}^{(2)}, \vec{\phi}^{(3)}, \cdots, \vec{\phi}^{(N)}\right\} \qquad \left\langle O(\vec{\phi}) \right\rangle = \frac{1}{Z(\vec{0})} \int d\vec{\phi} O(\vec{\phi}) \exp\left[-S(\vec{\phi})\right] \approx \frac{1}{N} \sum_{j=1}^{N} O(\vec{\phi}^{(j)})$$

- The basic idea of HMC comes from statistical mechanics.

$$Z(\vec{0}) = \int d\vec{\phi} \exp\left[-S(\vec{\phi})\right]$$

$$\left\langle O(\vec{\phi})\right\rangle = \frac{1}{Z(\vec{0})}\int d\vec{\phi} O(\vec{\phi}) \exp\left[-S(\vec{\phi})\right]$$

- Constant expressed by Gaussian integral $C = \int d\vec{\pi} \exp\left[-\frac{\vec{\pi}^2}{2}\right]$

$\vec{\pi}$: real scalar coordinate having same d.o.f to $\vec{\phi}$

- Multiply this constant to the partition function

$$Z_{HMC} = Z(\vec{0})C = \int d\vec{\pi} d\vec{\phi} \exp\left[-\left(\frac{\vec{\pi}^2}{2} + S(\vec{\phi})\right)\right] = \int d\vec{\pi} d\vec{\phi} \exp\left[-H(\vec{\pi},\vec{\phi})\right]$$

$$\left\langle O(\vec{\phi})\right\rangle = \frac{1}{Z_{HMC}}\int d\vec{\pi} d\vec{\phi} O(\vec{\phi}) \exp\left[-H(\vec{\pi},\vec{\phi})\right]$$

- This is identical to the Canonical ensemble estimator mathematically.

2. Hybrid Monte Carlo (HMC)

$$Z_{HMC}(\vec{\eta}) = \int d\vec{\pi} d\vec{\phi} \exp\left[- H(\vec{\pi}, \vec{\phi}) + \vec{\eta} \cdot \vec{\phi} \right] \qquad H(\vec{\pi}, \vec{\phi}) = \frac{\vec{\pi}^2}{2} + S(\vec{\phi})$$

$$\left\langle O(\vec{\phi}) \right\rangle = \frac{1}{Z_{HMC}(\vec{0})} \int d\vec{\pi} d\vec{\phi} O(\vec{\phi}) \exp\left[- H(\vec{\pi}, \vec{\phi}) \right]$$

- We can use Metropolis algorithm to generate ensemble for random variables $\left(\vec{\pi}, \vec{\phi}\right)$.

$$\left\{ \left(\vec{\pi}^{(1)}, \vec{\phi}^{(1)}\right), \left(\vec{\pi}^{(2)}, \vec{\phi}^{(2)}\right), \left(\vec{\pi}^{(3)}, \vec{\phi}^{(3)}\right), \cdots, \left(\vec{\pi}^{(N)}, \vec{\phi}^{(N)}\right) \right\} \qquad w((\vec{\pi}, \vec{\phi})) \propto \exp\left[- H(\vec{\pi}, \vec{\phi}) \right]$$

$$\rho((\vec{\pi}, \vec{\phi}), (\vec{\pi}', \vec{\phi}')) = \min\left(1, \exp\left[- H(\vec{\pi}, \vec{\phi}) + H(\vec{\pi}', \vec{\phi}') \right]\right)$$

$$q((\vec{\pi}, \vec{\phi}) \,|\, (\vec{\pi}', \vec{\phi}')) = ? \qquad [\text{Candidate generation prob.}]$$

- For good Metropolis test acceptance rate, generating candidate state $(\vec{\pi}, \vec{\phi})$ requires small energy change

$$\Delta H \equiv H(\vec{\pi}, \vec{\phi}) - H(\vec{\pi}', \vec{\phi}') \approx 0$$

- We know that the canonical ensemble reproduces the identical prediciont to the Micro canonical ensemble in the thermodynamic limit.
- The average with the Micro canonical ensemble can be expressed as

$$\left\langle O(\vec{\phi})\right\rangle = \lim_{T\to\infty}\int_0^T d\tau O(\vec{\phi}^{(\tau)})$$

- Where τ is a fictitious time for micro canonical ensemble and $\vec{\pi}^{(\tau)}$ is treated as the canonical momentum to the $\vec{\phi}^{(\tau)}$.
- The dynamics of $\left(\vec{\pi}^{(\tau)},\vec{\phi}^{(\tau)}\right)$ is governed by the Hamitonian

$$H(\vec{\pi},\vec{\phi}) = \frac{\vec{\pi}}{2} + S(\vec{\phi})$$

- And equation of motion:

$$\frac{d}{d\tau}\vec{\phi}^{(\tau)} = \vec{\pi}^{(\tau)} = \frac{d}{d\vec{\pi}}H(\vec{\pi}^{(\tau)},\vec{\phi}^{(\tau)})$$
$$\frac{d}{d\tau}\vec{\pi}^{(\tau)} = -\frac{d}{d\vec{\phi}}S(\vec{\pi}^{(\tau)},\vec{\phi}^{(\tau)}) = -\frac{d}{d\vec{\phi}}H(\vec{\pi}^{(\tau)},\vec{\phi}^{(\tau)})$$

- We can use this property to generate candidate state $(\vec{\pi},\vec{\phi})$ from $(\vec{\pi}',\vec{\phi}')$.
  - Initial condition : $(\vec{\pi}^{(0)},\vec{\phi}^{(0)}) = (\vec{\pi}',\vec{\phi}')$
  - Fictitious time evolution via Hamiltonian eq. of motion : $(\vec{\pi}^{(\tau)},\vec{\phi}^{(\tau)}) \underset{\text{timeevoluve}}{\leftarrow} (\vec{\pi}^{(0)},\vec{\phi}^{(0)})$
  - Candidate for Metropolis test : $(\vec{\pi},\vec{\phi}) = (\vec{\pi}^{(\tau)},\vec{\phi}^{(\tau)})$

- HMC algorithm

(step 0) Given initial state $\vec{\phi}$.

(step 1) Generate initial momentum $\vec{\pi}$ with the probablity/Gaussian dist. $\exp\left(-\frac{\vec{\pi}^2}{2}\right)$

(step 2) Set initial state $(\vec{\pi}^{(0)}, \vec{\phi}^{(0)}) = (\vec{\pi}, \vec{\phi})$, where $\vec{\phi}$ is from latest ensemble.

(step 3) Generate candidate state $(\vec{\pi}^{(\tau)}, \vec{\phi}^{(\tau)})$ via integratin g Hamilton eq. of m.

for a fixed time τ. $\dot{\vec{\phi}} = \vec{\pi}, \dot{\vec{\pi}} = -\vec{\nabla} S(\vec{\phi}), \Leftarrow H(\vec{\pi}, \vec{\phi}) = \frac{\vec{\pi}^2}{2} + S(\vec{\phi}).$

(step 4) Do Metropolis test. If it is accepted $(\vec{\phi}^{(\tau)})$ is added to the ensemble.

If rejected $(\vec{\phi}^{(0)})$ is added to the ensemble.

(step 5) Goto step 1.

- In step 2, we can directly generate the fictitious momentum from Gaussian distribution as it should be.
- Step 3 requires the numerical integration on Hamilton eq. of m. This is called as the Molecular dynamics (MD) integration/evolution. This violates the energy conservation. Thus the Metropolis test is required.
- Step 3 corresponds to the candidate generation. The transition prob. for this candidate generation should be symmetric $q_{ij} = q_{ji}$ to satisfy the detailed balance (Lect#1). Thus the MD integrator should be revertible and at the end of time evolution the momentum should be reversed.
- In step 4, the momentum is not stored to the ensemble since it is not required for observables.

- The efficiency of the HMC algorithm depends on the numerical MD integrator performance.
  - Time reversible (and area preserving) integrator.
  - Smaller energy conservation violation with larger time step.
  - The most time consuming part of the MD integrator is the computation of force.

HMC = MD integrator + Metropolis algorithm

- We need better MD integration scheme.

3. MD integrators

- We need to integrate the following Hamiltonian equation of motion.

$$\frac{d}{d\tau}\vec{\phi}^{(\tau)} = \vec{\pi}^{(\tau)} = \frac{d}{d\vec{\pi}} H(\vec{\pi}^{(\tau)}, \vec{\phi}^{(\tau)})$$

$$\frac{d}{d\tau}\vec{\pi}^{(\tau)} = -\frac{d}{d\vec{\phi}} S(\vec{\pi}^{(\tau)}, \vec{\phi}^{(\tau)}) = -\frac{d}{d\vec{\phi}} H(\vec{\pi}^{(\tau)}, \vec{\phi}^{(\tau)})$$

$$H(\vec{\pi}, \vec{\phi}) = \frac{\vec{\pi}}{2} + S(\vec{\phi})$$

- Simple Runge-Kutta method is not time reversible. This cannot be used for HMC.

$$(\vec{\pi}^{(0)}, \vec{\phi}^{(0)}) \underset{\text{Runge-Kutta}}{\rightarrow} (\vec{\pi}^{(\tau)}, \vec{\phi}^{(\tau)})$$

$$(-\vec{\pi}^{(\tau)}, \vec{\phi}^{(\tau)}) \underset{\text{Runge-Kutta}}{\rightarrow} (\vec{\pi}^{(2\tau)}, \vec{\phi}^{(2\tau)})$$

$$(\vec{\pi}^{(0)}, \vec{\phi}^{(0)}) \neq (-\vec{\pi}^{(2\tau)}, \vec{\phi}^{(2\tau)})$$

$$q\left((-\vec{\pi}^{(\tau)}, \vec{\phi}^{(\tau)}) \,|\, (\vec{\pi}^{(0)}, \vec{\phi}^{(0)})\right) \neq q\left((\vec{\pi}^{(0)}, \vec{\phi}^{(0)}) \,|\, (-\vec{\pi}^{(\tau)}, \vec{\phi}^{(\tau)})\right)$$

- Properties of Hamiltonian dynamics.
  - Poisson brackets.

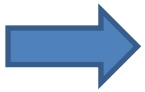
$$\{X,Y\} \equiv \frac{\partial X}{\partial \vec{\phi}} \cdot \frac{\partial Y}{\partial \vec{\pi}} - \frac{\partial X}{\partial \vec{\pi}} \cdot \frac{\partial Y}{\partial \vec{\phi}}, \qquad X,Y: \text{ functions of } (\vec{\pi},\vec{\phi})$$

$$\{X,Y\} = -\{Y,X\}$$
$$\{X,aY+bZ\} = a\{X,Y\} + b\{X,Z\}$$
$$\{X,\{Y,Z\}\} + \{Z,\{X,Y\}\} + \{Y,\{Z,X\}\} = 0$$

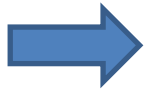
Defines Lie algebra

  - Define Liouville operator *L* : $iL_Y X \equiv \{X,Y\}$
  - Hamiltonian eq. of m.

$$\frac{d\vec{\phi}}{d\tau} = \{\vec{\phi},H\} = +\frac{dH}{d\vec{\pi}} = \vec{\pi}$$
$$\frac{d\vec{\pi}}{d\tau} = \{\vec{\pi},H\} = -\frac{dH}{d\vec{\phi}} = -\frac{dS}{d\vec{\phi}}$$



$$\frac{d\vec{\phi}}{d\tau} = iL_H\vec{\phi}$$
$$\frac{d\vec{\pi}}{d\tau} = iL_H\vec{\pi}$$



$$\frac{d}{d\tau}\begin{pmatrix}\vec{\phi}\\ \vec{\pi}\end{pmatrix} = iL_H\begin{pmatrix}\vec{\phi}\\ \vec{\pi}\end{pmatrix}$$

- Properties of Hamiltonian dynamics.
  - Formal solution of Hamilton equation.

$$\frac{d}{d\tau}\begin{pmatrix}\vec{\phi}\\ \vec{\pi}\end{pmatrix} = iL_H \begin{pmatrix}\vec{\phi}\\ \vec{\pi}\end{pmatrix} \quad \Rightarrow \quad \begin{pmatrix}\vec{\phi}\\ \vec{\pi}\end{pmatrix}(\tau) = \exp\left[\tau \times iL_H\right]\begin{pmatrix}\vec{\phi}\\ \vec{\pi}\end{pmatrix}(0)$$

  - Where $\exp(\tau \times iL_Y)X \equiv X + \tau \times iL_Y X + \frac{\tau^2}{2!}(iL_Y)^2 X + \frac{\tau^3}{3!}(iL_Y)^3 X + \ldots$

$$= X + \tau\{X,Y\} + \frac{\tau^2}{2!}\{\{X,Y\},Y\} + \frac{\tau^3}{3!}\{\{\{X,Y\},Y\}\} + \ldots$$

Similar formula for commutation bracket in Quantum mechanics. Algebra is same.

  - Finding a time reversible and area conserving MD integration integrator == finding simple approximation for time evolution operator $\exp\left[\tau \times iL_H\right]$.

- Approximation for the time evolution op.
  - Our Hamiltonian is composed of Kinetic term T and potential term S.

$$H(\vec{\pi},\vec{\phi}) = \frac{\vec{\pi}}{2} + S(\vec{\phi}) = T(\vec{\pi}) + V(\vec{\phi})$$

  - The Hamiltonian Lioubille operator can be decomposed to two parts.

$$iL_H X = \{X, H\} = \{X, T+V\} = \{X, T\} + \{X, V\} = iL_T X + iL_V X \qquad [iL_T, iL_V]X \neq 0$$

Note: L_V and L_T are not commutable.

  - Time evolution operator

$$\exp\left[\tau \times iL_H\right] = \exp\left[\tau \times \left(iL_T + iL_V\right)\right]$$

  - We can approximate this operator as

$$\exp\left[\tau \times iL_H\right] = \exp\left[\tau \times \left(iL_T + iL_V\right)\right]$$

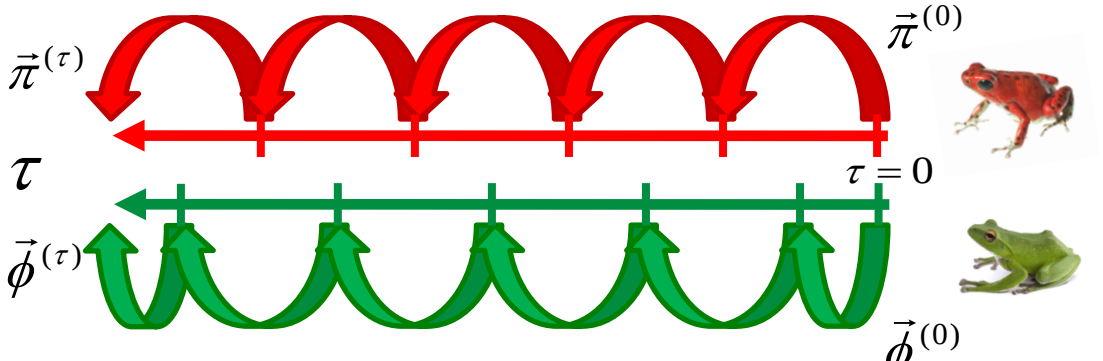
$$\approx \left[\exp\left[\frac{\tau}{2N_{MD}} \times iL_T\right] \exp\left[\frac{\tau}{N_{MD}} \times iL_V\right] \exp\left[\frac{\tau}{2N_{MD}} \times iL_T\right]\right]^{N_{MD}}$$

$e^{(A+B)\delta} \approx e^{A\delta/2} e^{B\delta} e^{A\delta/2}$ A symmetric decomposition of Matrix Exponential.

Leapfrog Scheme

- Properties of the Leapfrog Scheme
  - Leapfrog time evolution operator U_{LF} :

$$\exp\left[\tau \times iL_H\right] \approx U_{\mathrm{LF}}(\tau) \equiv \left[Q(\frac{\Delta\tau}{2}) P(\Delta\tau) Q(\frac{\Delta\tau}{2}) \right]^{N_{MD}}$$



  - The Kinetic and potential exponential operators are exactly computable and invertible.

$$\exp\left[\Delta\tau \times iL_T\right] \begin{pmatrix} \vec{\phi}^{(0)} \\ \vec{\pi}^{(t)} \end{pmatrix} \equiv Q(\Delta\tau) \begin{pmatrix} \vec{\phi}^{(0)} \\ \vec{\pi}^{(t)} \end{pmatrix} = \exp\left[\Delta\tau \times \vec{\pi}^{(t)} \cdot \frac{\partial}{\partial \vec{\phi}} \right] \begin{pmatrix} \vec{\phi}^{(0)} \\ \vec{\pi}^{(t)} \end{pmatrix} = \begin{pmatrix} \vec{\phi}^{(0)} + \Delta\tau \times \vec{\pi}^{(t)} \\ \vec{\pi}^{(t)} \end{pmatrix}$$

$$\exp\left[\Delta\tau \times iL_V\right] \begin{pmatrix} \vec{\phi}^{(t)} \\ \vec{\pi}^{(0)} \end{pmatrix} \equiv P(\Delta\tau) \begin{pmatrix} \vec{\phi}^{(t)} \\ \vec{\pi}^{(0)} \end{pmatrix} = \exp\left[\Delta\tau \times \vec{F}(\vec{\phi}^{(t)}) \cdot \frac{\partial}{\partial \vec{\pi}} \right] \begin{pmatrix} \vec{\phi}^{(t)} \\ \vec{\pi}^{(0)} \end{pmatrix} = \begin{pmatrix} \vec{\phi}^{(t)} \\ \vec{\pi}^{(0)} + \Delta\tau \times \vec{F}(\vec{\phi}^{(t)}) \end{pmatrix}$$

$$Q(\tau)Q(-\tau) = 1 \rightarrow Q(-\tau) = Q(\tau)^{-1}, \quad P(\tau)P(-\tau) = 1 \rightarrow P(-\tau) = P(\tau)^{-1}$$

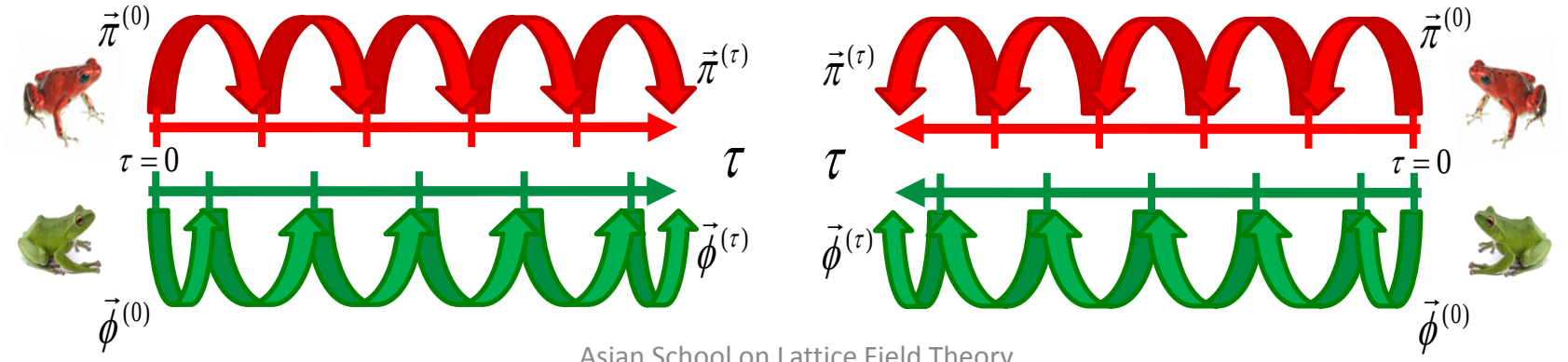
- Properties of the Leapfrog Scheme
  - Thus the Leapfrog operator U_{LF} is invertible.

$$U_{\mathrm{LF}}(\tau) \equiv \left[Q(\frac{\Delta\tau}{2}) P(\Delta\tau) Q(\frac{\Delta\tau}{2}) \right]^{N_{MD}}$$

$$Q(\tau)Q(-\tau) = 1 \rightarrow Q(-\tau) = Q(\tau)^{-1}, \quad P(\tau)P(-\tau) = 1 \rightarrow P(-\tau) = P(\tau)^{-1}$$

$$\begin{aligned} U_{\mathrm{LF}}(\tau)U_{\mathrm{LF}}(-\tau) &= \left[Q(\frac{\Delta\tau}{2}) P(\Delta\tau) Q(\frac{\Delta\tau}{2}) \right]^{N_{MD}} \left[Q(-\frac{\Delta\tau}{2}) P(-\Delta\tau) Q(-\frac{\Delta\tau}{2}) \right]^{N_{MD}} \\ &= \left[Q(\frac{\Delta\tau}{2}) P(\Delta\tau) Q(\frac{\Delta\tau}{2}) \right]^{N_{MD}} \left[Q(\frac{\Delta\tau}{2})^{-1} P(\Delta\tau)^{-1} Q(\frac{\Delta\tau}{2})^{-1} \right]^{N_{MD}} \\ &= 1 \end{aligned}$$

Half time step evolution at first and last on ϕ is important to ensure the reversibility.



- Properties of the Leapfrog Scheme
  - The leapfrog time evolution operator is one of the approximation for true time evolution op.
  - This operator violates energy conservation low.
  - We can analyze the accuracy of the approximation using the Backer-Campbell-Hausdorff formula for matrix exponential.

$$e^{A}e^{B} \approx e^{A+B+\frac{1}{2}[A,B]+\frac{1}{12}([A,[A,B]]+[B[B,A]])\cdots}$$: B.C.H. formula

$$\exp\left[\tau \times iL_H\right] \approx U_{\mathrm{LF}}(\tau) = \left[e^{\frac{\Delta\tau}{2}\times iL_T} e^{\Delta\tau \times iL_V} e^{\frac{\Delta\tau}{2}\times iL_T} \right]^{N_{MD}} = \exp\left[\tau \times iL_{H(\mathrm{Shadow})}\right]$$

  - Here we defined a Shadow Hamiltonian $H^{(\mathrm{Shadow})}$. The discrepancy from the true Hamiltonian is

$$H^{(\mathrm{Shadow})} = H - \frac{1}{24}\left(\{T,\{T,V\}\}+2\{V,\{T,V\}\}\right)\Delta\tau^2 + O\left(\Delta\tau^4\right)$$

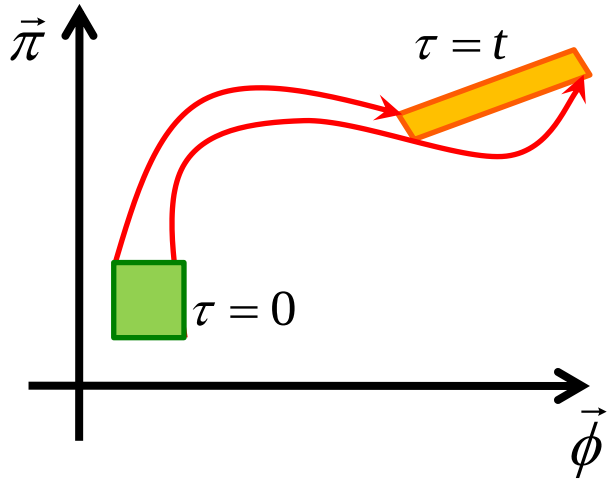
Problem 1

  - We expect $O(\Delta\tau^2)$ violation on the energy conservation low. This affects the Metropolis test acceptance rate directly.

- Area preservation of the Leapfrog Scheme
  - The candidate generation should preserve integration measure in the continuous case to preserve the transition probability. This means

Known as Liouville's theorem in analytical mechanics

$$d\vec{\pi}d\vec{\phi}\Big|_{\tau=0} = d\vec{\pi}d\vec{\phi}\Big|_{\tau=t}$$



  - The area preservation holds for the leapfrog scheme. To prove this property, it is sufficient to consider the time evolution for a single time step.

$$\begin{pmatrix} \vec{\phi}^{(\Delta\tau)} \\ \vec{\pi}^{(\Delta\tau)} \end{pmatrix} = e^{\frac{\Delta\tau}{2}\times iL_T} e^{\Delta\tau\times iL_V} e^{\frac{\Delta\tau}{2}\times iL_T} \begin{pmatrix} \vec{\phi}^{(0)} \\ \vec{\pi}^{(0)} \end{pmatrix}$$

  - The Jacobian of this transformation is proved to be 1.

Problem 2

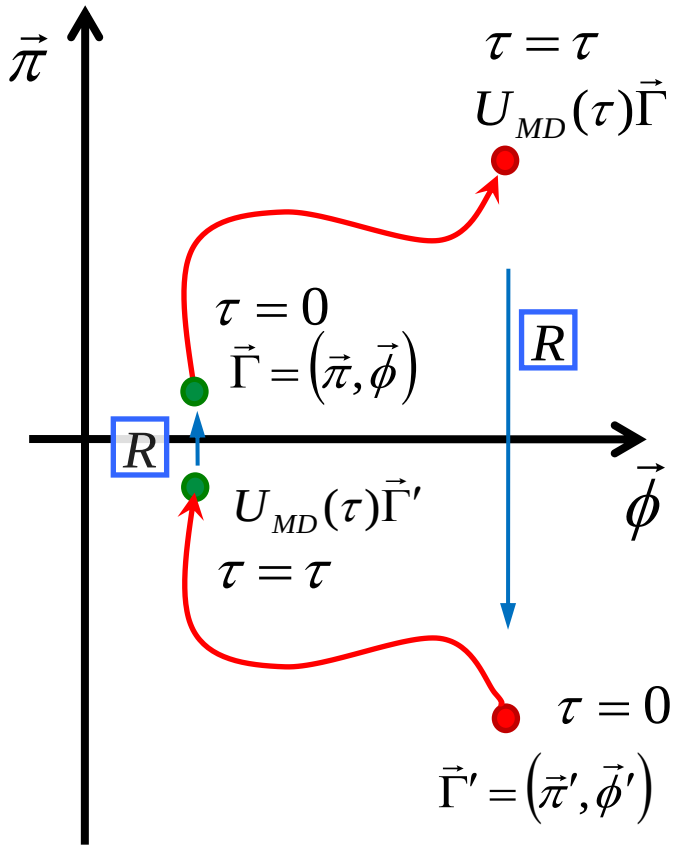
- Time reversal and momentum flip in the MD integrators
  - The MD integrators should be constructed to be time reversible and area preserving.
  - We can integrate back on the same trajectory by flipping the Momentum.
  - This means

$$\vec{\Gamma} = (\vec{\pi}, \vec{\phi}) \qquad \vec{\Gamma}' = (\vec{\pi}', \vec{\phi}')$$

$$\vec{\Gamma}' = RU_{MD}(\tau)\vec{\Gamma} \quad \Leftrightarrow \quad RU_{MD}(\tau)\vec{\Gamma}' = \vec{\Gamma}$$

$$R: \text{momentum flip op.} \quad R(\vec{\pi}, \vec{\phi}) = (-\vec{\pi}, \vec{\phi})$$

  - In the phase space.



Note: The operation of *R* is usually irrelevant for *H* value.
The *R* operations are often omitted in the HMC algorithm.

4. Properties of the HMC algorithm

- Acceptance rate
  - The leapfrog scheme has a $O(\Delta\tau^2)$ error.
  - Rejection occurs in the Metropolis test. How large/small is it?

$$\vec{\Gamma} = (\vec{\pi}, \vec{\phi})$$

  - Consider the partition function: $$Z_{HMC}(\vec{0}) = \int d\vec{\Gamma} \exp\left[-H(\vec{\Gamma})\right]$$
  - A MD integrator+momentum flip moves/maps $\vec{\Gamma} = (\vec{\pi}, \vec{\phi})$ to $\vec{\Gamma}' = (\vec{\pi}', \vec{\phi}')$ as

$$\vec{\Gamma}' = RU_{MD}(\tau)\vec{\Gamma}$$

  - Considering this MD evolution as a variable change for the partition function, we have

$$\boxed{\left\langle e^{-\Delta H} \right\rangle = 1}$$

Expectation value of the exponential of the Energy conservation violation should be 1 for area preserving MD integrator.

$\because$

$$\int d\vec{\Gamma} \exp\left[-H(\vec{\Gamma})\right] = \int d\vec{\Gamma}' \exp\left[-H(\vec{\Gamma}')\right] = \int d(RU_{MD}\vec{\Gamma}) \exp\left[-H(RU_{MD}\vec{\Gamma})\right]$$
$$= \int d\vec{\Gamma} \exp\left[-H(RU_{MD}\vec{\Gamma})\right] = \int d\vec{\Gamma} \exp\left[-H(RU_{MD}\vec{\Gamma}) + H(\vec{\Gamma})\right] \exp\left[-H(\vec{\Gamma})\right]$$
$$= \int d\vec{\Gamma} \exp\left[-\Delta H\right] \exp\left[-H(\vec{\Gamma})\right]$$

- Acceptance rate of the HMC Metropolis test
  - From $\left\langle e^{-\Delta H} \right\rangle = 1$, we can estimate the acceptance rate by assuming a Gaussian distribution for $\Delta H = x$ (as a random variable).

$$1 = \left\langle e^{-x} \right\rangle = \int_{-\infty}^{\infty} e^{-x} \frac{e^{-\frac{(x-\mu)^2}{2\sigma^2}}}{\sqrt{2\pi\sigma^2}} = e^{-\mu+\frac{\sigma^2}{2}} \Rightarrow \mu = \frac{\sigma^2}{2}$$

  - From the problem #(5) of Lecture 1 we can estimate the averatged acceptance rate as

$$\left\langle P_{acc} \right\rangle = \int_{-\infty}^{\infty} \min(1, e^{-\Delta H}) \frac{1}{\sqrt{4\pi\mu}} e^{-\frac{(\Delta H-\mu)^2}{4\mu}} = \mathrm{erfc}\left(\frac{\sqrt{\Delta H}}{2} \right)$$

  - For the Leapfrog MD integrator we have $\Delta H \propto \Delta\tau^2$ and

$$\left\langle \Delta H \right\rangle = \frac{Var(\Delta H)}{2} = \frac{1}{2}\left(\left\langle \Delta H^2 \right\rangle - \left\langle \Delta H \right\rangle^2 \right) \propto \Delta\tau^4$$

  - Thus

$$\left\langle P_{acc} \right\rangle = \mathrm{erfc}\left(\frac{Const \times (\Delta\tau)^2}{2} \right)$$

- HMC transition probability for $\vec{\phi} \leftarrow \vec{\phi}'$

$$P_{\mathrm{HMC}}(\vec{\phi} \mid \vec{\phi}') = u(\vec{\phi} \mid \vec{\phi}') + (1 - r(\vec{\phi}'))\delta(\vec{\phi} \mid \vec{\phi}')$$

$$u(\vec{\phi} \mid \vec{\phi}') = \int d\vec{\pi} \left(\int \left[\rho((\vec{\pi}, \vec{\phi}) \mid (\vec{\pi}', \vec{\phi}'))q((\vec{\pi}, \vec{\phi}) \mid (\vec{\pi}', \vec{\phi}'))e^{-\frac{\vec{\pi}'}{2}} \right] d\vec{\pi}' \right)$$

$$r(\vec{\phi}') = \int d\vec{\phi} u(\vec{\phi} \mid \vec{\phi}')$$

π' is generated with Gaussian.
π and π' are integrated out since they are not measured.

$$\rho((\vec{\pi}, \vec{\phi}), (\vec{\pi}', \vec{\phi}')) = \min\left(1, e^{-H(\vec{\pi}, \vec{\phi}) + H(\vec{\pi}', \vec{\phi}')}\right)$$

$$q((\vec{\pi}, \vec{\phi}) \mid (\vec{\pi}', \vec{\phi}')) = \delta((\vec{\pi}, \vec{\phi}) - RU_{MD}{}^{(\tau)}(\vec{\pi}', \vec{\phi}'))$$

Metropolis test probability

Candidate generation probability
This is deterministic via MD integration.

$$P_{\mathrm{HMC}}(\vec{\phi}' \mid \vec{\phi}) e^{-S(\vec{\phi})} = P_{\mathrm{HMC}}(\vec{\phi} \mid \vec{\phi}') e^{-S(\vec{\phi}')}$$

Problem 3

Satisfies the detailed balance condition

5. Examples

(1) Gaussian distribution with HMC

$$Z(\eta) = \int d\phi \exp\left[-S(\phi) + \eta\phi\right], \qquad S(\phi) = \omega^2 \frac{\phi^2}{2}$$

$$\left\langle O(\phi) \right\rangle = \frac{1}{Z(0)} \int d\phi O(\phi) \exp\left[-S(\phi)\right]$$

– transformation to the HMC weight.

$$Z_{\mathrm{HMC}}(\eta) = \int d\pi d\phi \exp\left[-H(\pi,\phi) + \eta\phi\right]$$

$$H(\pi,\phi) = \frac{\pi^2}{2} + S(\phi) = \frac{\pi^2}{2} + \omega^2 \frac{\phi^2}{2}$$

$$\left\langle O(\phi) \right\rangle = \frac{1}{Z_{\mathrm{HMC}}(0)} \int d\pi d\phi O(\phi) \exp\left[-H(\pi,\phi)\right]$$

– MD system becomes a Harmonic oscillator $H(\pi,\phi) = \frac{\pi^2}{2} + \omega^2 \frac{\phi^2}{2}$

(1) Gaussian distribution with HMC

- MD system becomes a Harmonic oscillator

$$H(\pi,\phi)=\frac{\pi^2}{2}+\omega^2\frac{\phi^2}{2}$$

- Equation of Motion

$$\dot{\pi}=-\omega^2\phi,\qquad \dot{\phi}=\pi$$

- Leapfrog integrator

$$\exp\left[\Delta\tau\times iL_T\right]\begin{pmatrix}\pi\\ \phi\end{pmatrix}=\exp\left[\Delta\tau\times\pi\frac{\partial}{\partial\phi}\right]\begin{pmatrix}\pi\\ \phi\end{pmatrix}=\begin{pmatrix}\pi\\ \phi+\Delta\tau\times\pi\end{pmatrix}=\begin{pmatrix}1 & 0\\ \Delta\tau & 1\end{pmatrix}\begin{pmatrix}\pi\\ \phi\end{pmatrix}$$

$$\exp\left[\Delta\tau\times iL_V\right]\begin{pmatrix}\pi\\ \phi\end{pmatrix}=\exp\left[-\Delta\tau\times\frac{\partial S}{\partial\phi}\frac{\partial}{\partial\pi}\right]\begin{pmatrix}\pi\\ \phi\end{pmatrix}=\begin{pmatrix}\pi-\Delta\tau\times\omega^2\phi\\ \phi\end{pmatrix}=\begin{pmatrix}1 & -\omega^2\Delta\tau\\ 0 & 1\end{pmatrix}\begin{pmatrix}\pi\\ \phi\end{pmatrix}$$

$$\begin{pmatrix}\pi'\\ \phi'\end{pmatrix}=U_{\mathrm{LF}}(\tau,N_{MD})\begin{pmatrix}\pi\\ \phi\end{pmatrix}=\left[\begin{pmatrix}1 & 0\\ \frac{\Delta\tau}{2} & 1\end{pmatrix}\begin{pmatrix}1 & -\omega^2\Delta\tau\\ 0 & 1\end{pmatrix}\begin{pmatrix}1 & 0\\ \frac{\Delta\tau}{2} & 1\end{pmatrix}\right]^{N_{MD}}\begin{pmatrix}\pi\\ \phi\end{pmatrix}$$

The Jacobian of this LF transformation is 1.

$$\Delta\tau\equiv\tau/N_{\mathrm{MD}}$$

When $|\Delta\tau|>2/\omega$, the MD evolution becomes unstable.

Problem 4

(1) Gaussian distribution with HMC

- Algorithm

(Step 0) Generate initial state ϕ

(Step 1) Generate initial momentum π from Gaussian dist.

(Step 2) MD evolution to get (π', ϕ') from (π, ϕ)

(Step 3) (flip momentum $\pi' = -\pi'$)

(Step 4) Do Metropolis test with prob. $P = \min(1, e^{-H'+H})$

If accepted $(\phi = \phi')$ and add (ϕ) to the ensemble.

If rejected add (ϕ) to the ensemble.

(Step 5) Goto Step 1

- For a Harmonic oscillator HMC algorithm is rather trivial. The Gaussian distribution of π is transmitted to ϕ distribution.

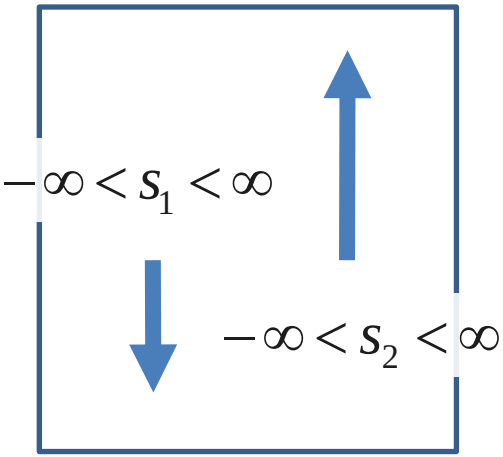
5. examples

(2) 2-site scalar model

- Partition function

$$\beta < 1$$

$$Z(\vec{\eta}) = \int_{-\infty}^{\infty} d\vec{s} \exp\left[-\left(\beta s_1 s_2 + \frac{{s_1}^2 + {s_2}^2}{2} \right) + \vec{\eta} \cdot \vec{s} \right]$$



- Observables

$$\left\langle \frac{s_1 + s_2}{2} \right\rangle = \frac{1}{Z(\vec{0})} \int_{-\infty}^{\infty} d\vec{s} \, \frac{s_1 + s_2}{2} \exp\left[-\left(\beta s_1 s_2 + \frac{{s_1}^2 + {s_2}^2}{2} \right) \right] = 0$$

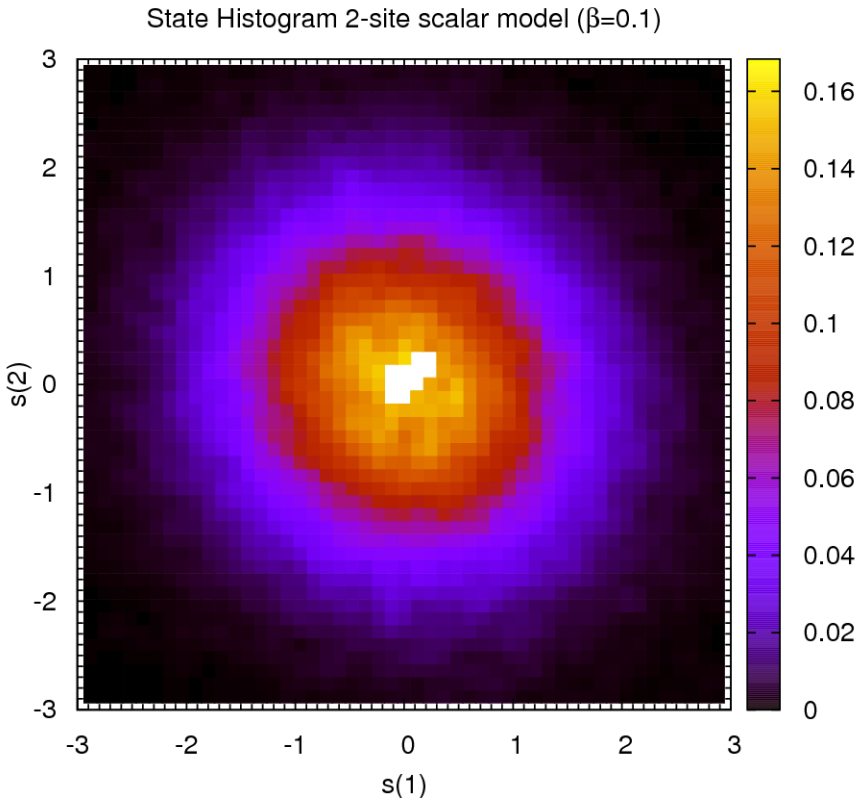
$$\langle s_1 s_2 \rangle = \frac{1}{Z(\vec{0})} \int_{-\infty}^{\infty} d\vec{s} \, (s_1 s_2) \exp\left[-\left(\beta s_1 s_2 + \frac{{s_1}^2 + {s_2}^2}{2} \right) \right] = -\frac{\beta}{1-\beta^2}$$

(2) 2-site scalar model

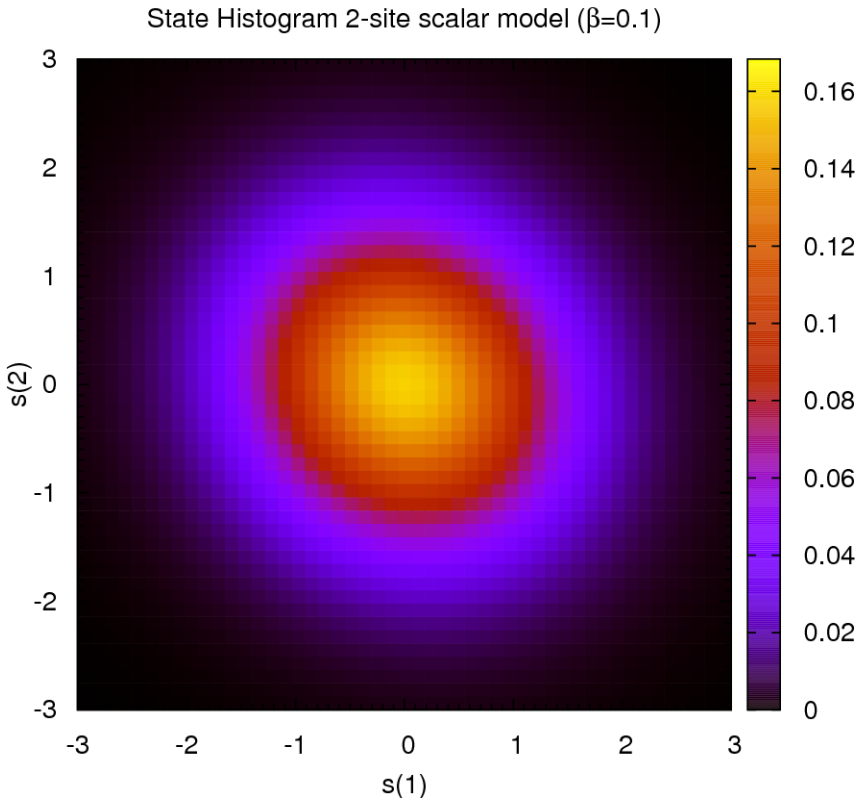
- In HMC, this corresponds to the coupled two Harmonic oscillators.
- Fortran Program [http://theo.phys.sci.hiroshima-u.ac.jp/~ishikawa/ASLFT2010/2SiteScalarHMC.tar.gz]
- 10,000,000 samples are generated. But we save 10,000 samples with interval 100. We use τ=2, N_{MD}=4 for candidate generation.

- State weight/histogram generated via Metropolis algorithm

HMC measured

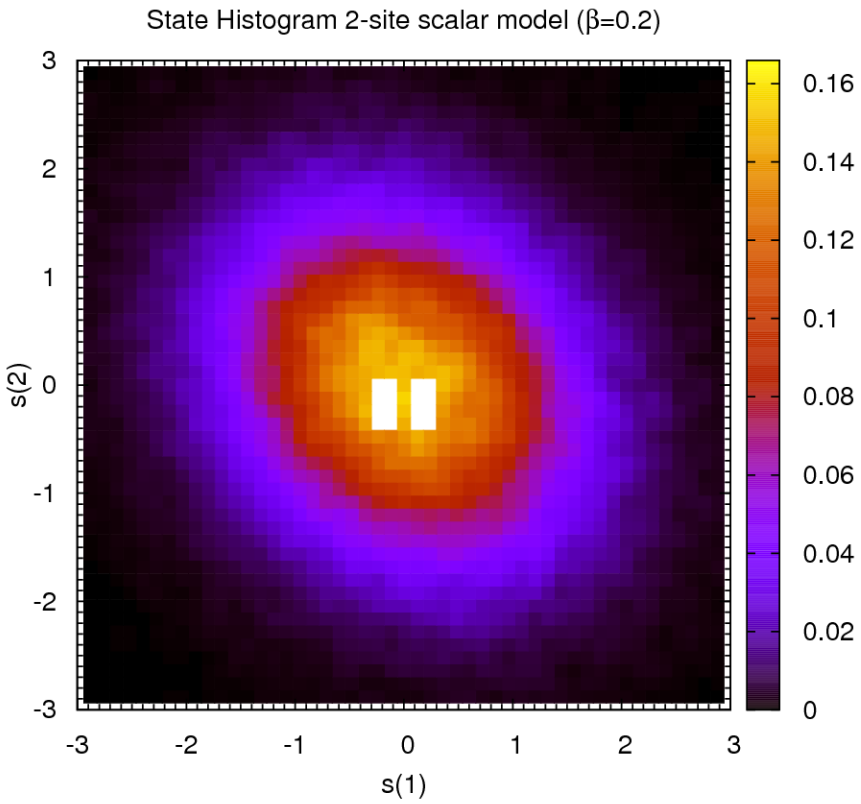


Theoretical

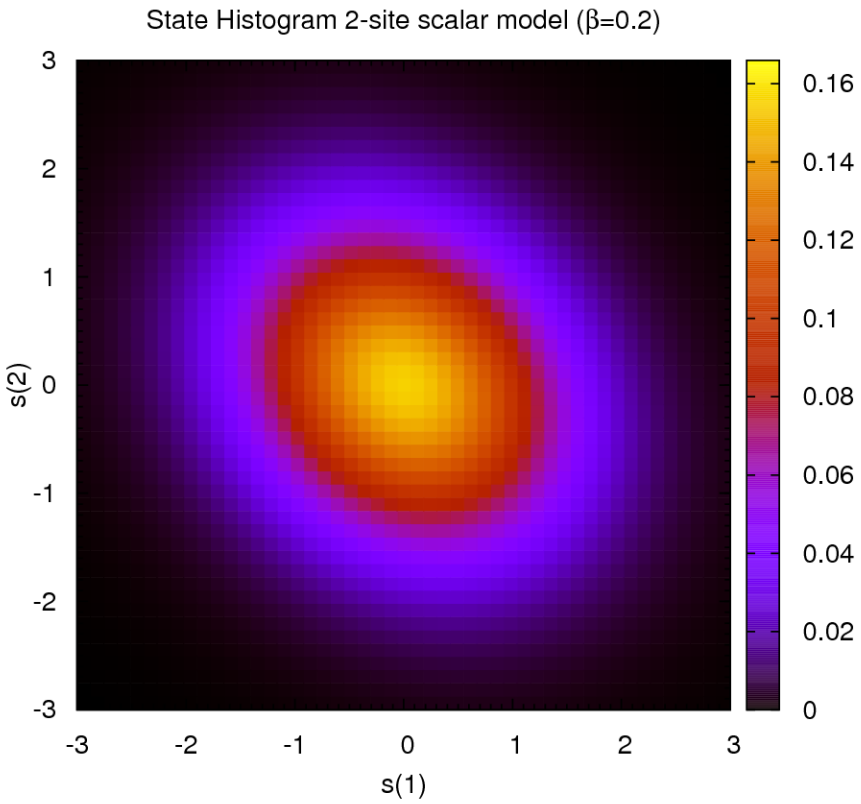


- State weight/histogram generated via Metropolis algorithm

HMC measured

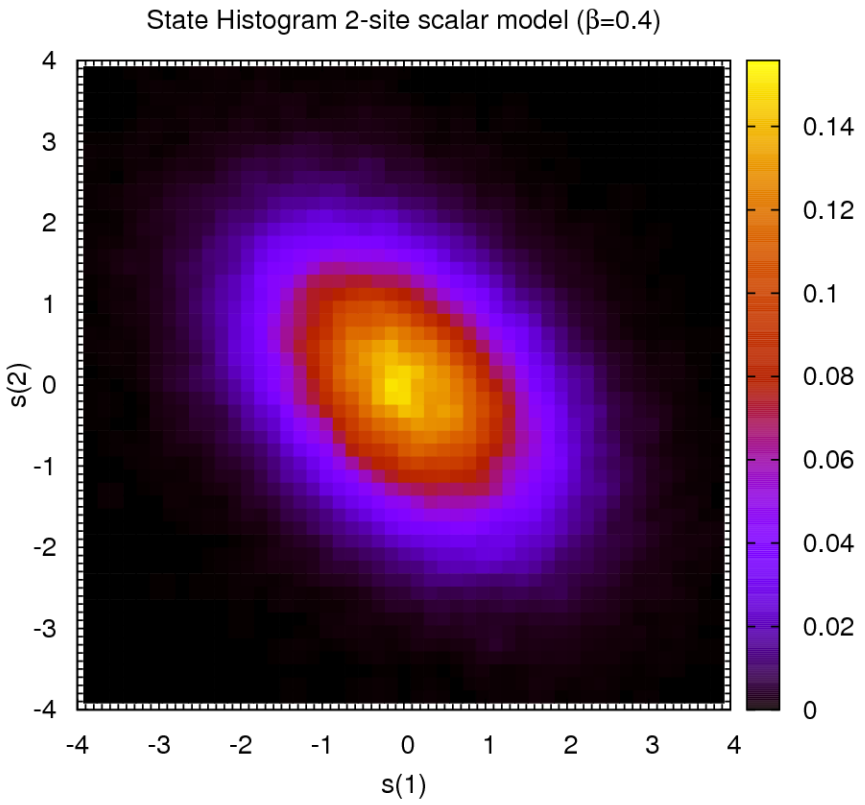


Theoretical

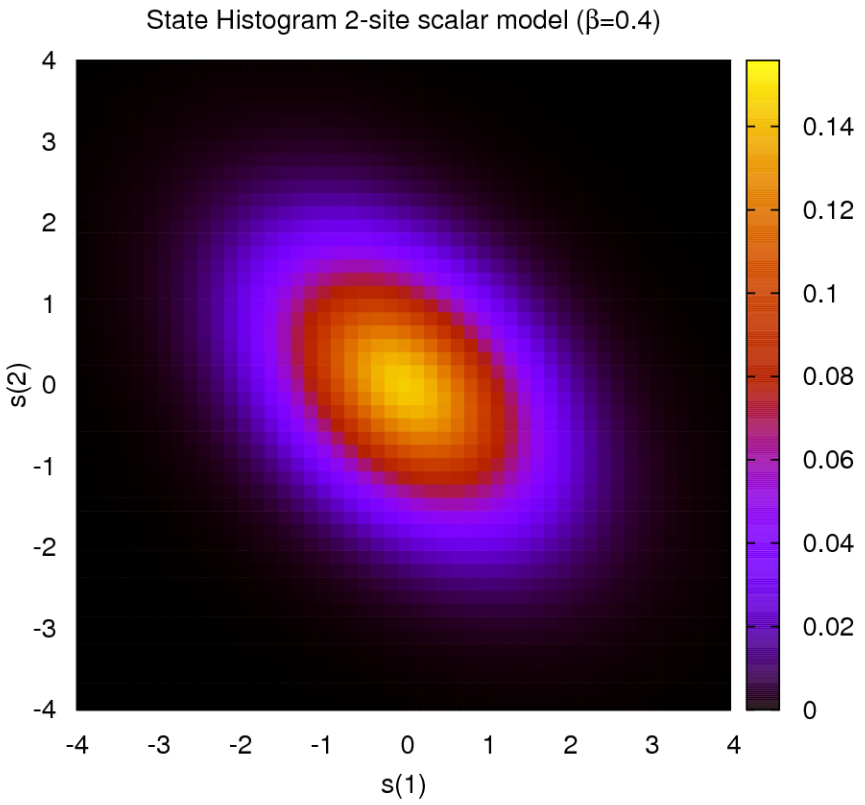


- State weight/histogram generated via Metropolis algorithm

HMC measured

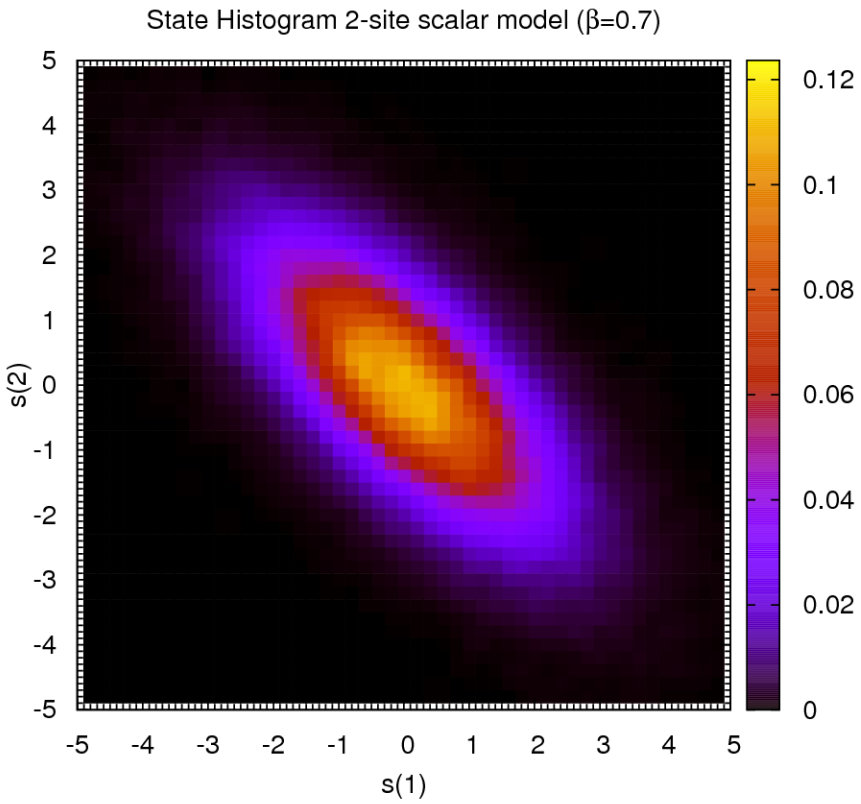


Theoretical

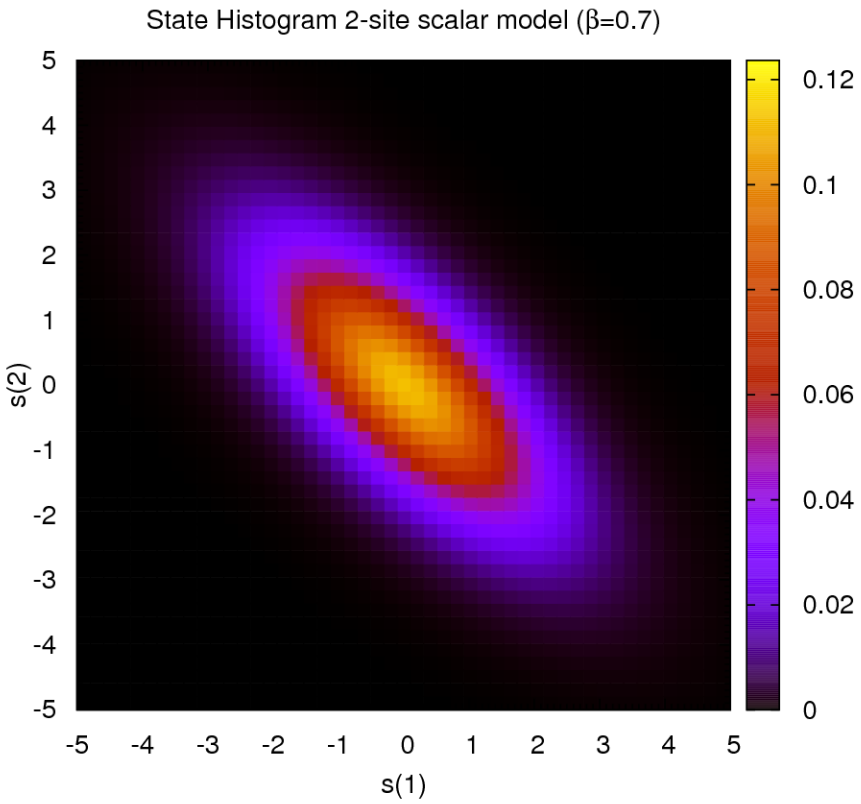


- State weight/histogram generated via Metropolis algorithm

HMC measured

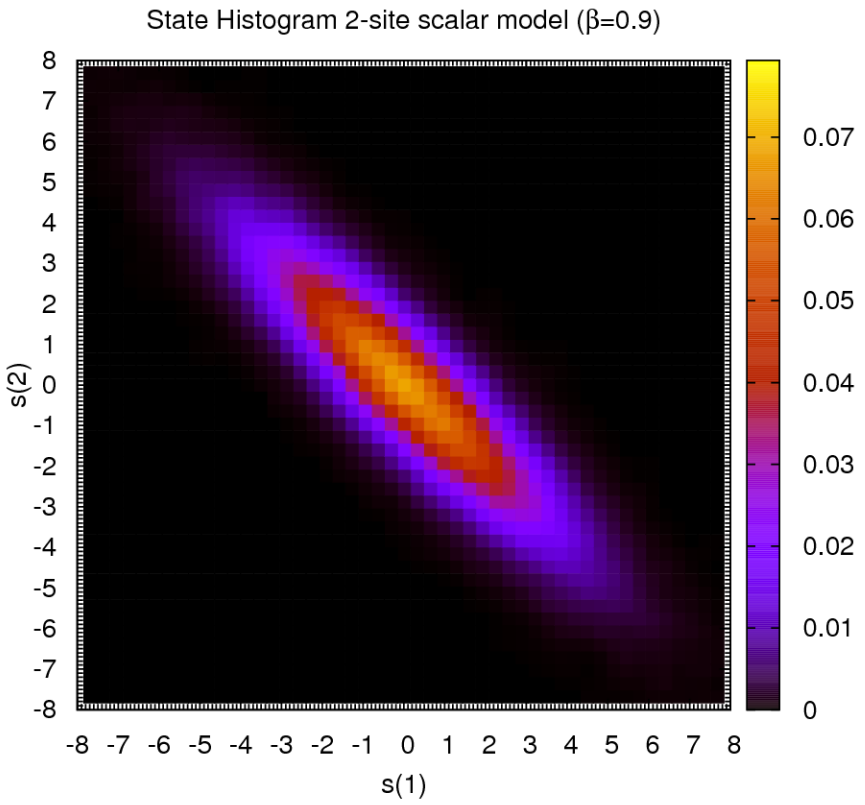


Theoretical

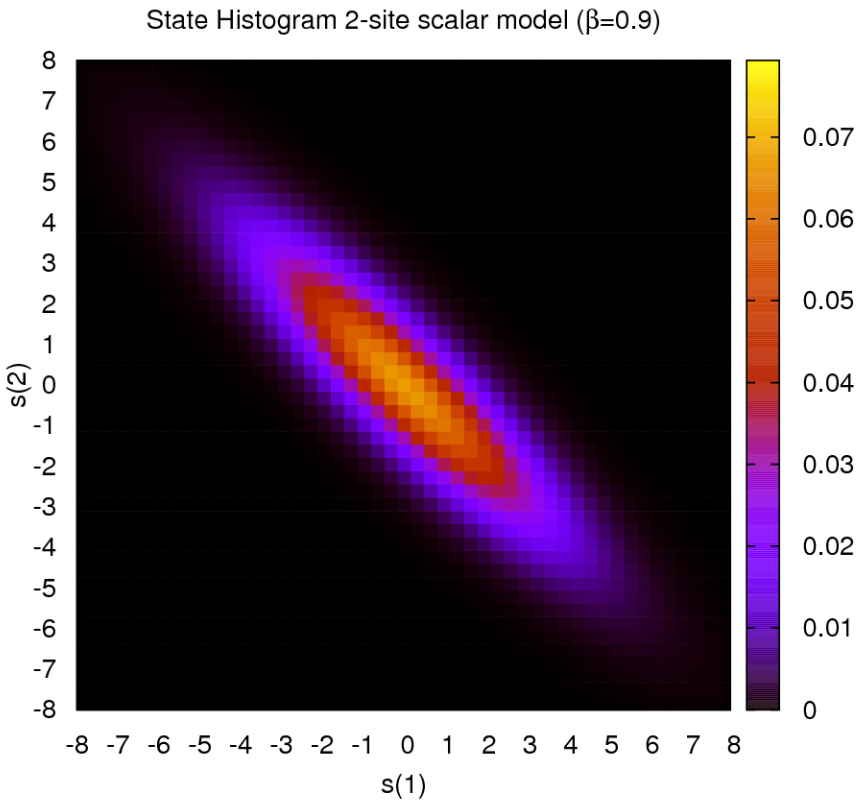


- State weight/histogram generated via Metropolis algorithm

HMC measured

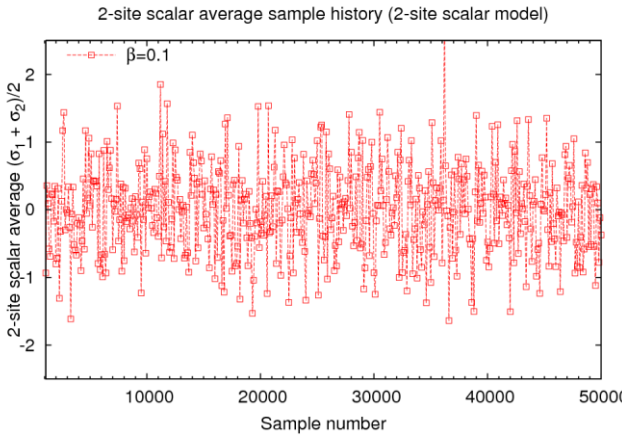


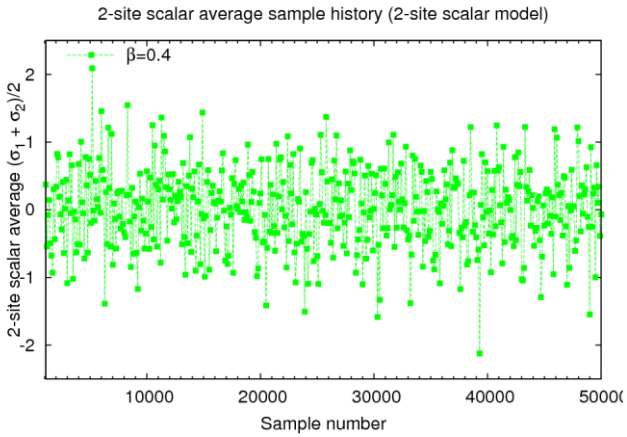
Theoretical

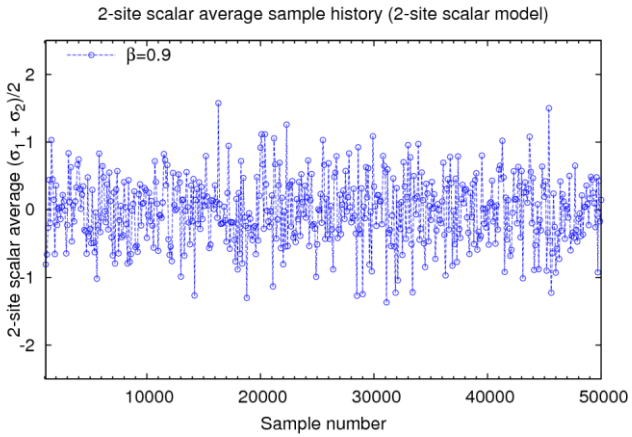


- Spin average/Spin correlation history

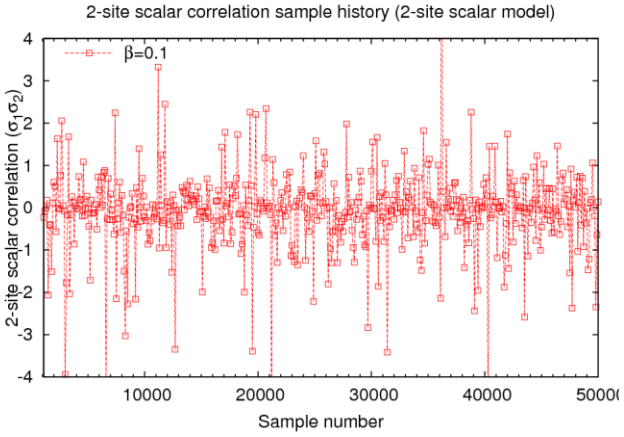
Spin average

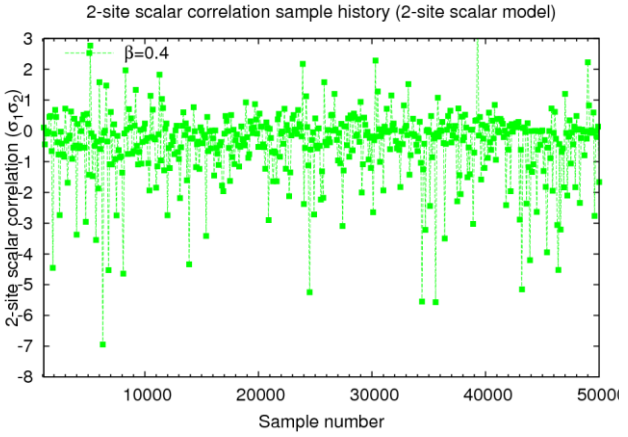


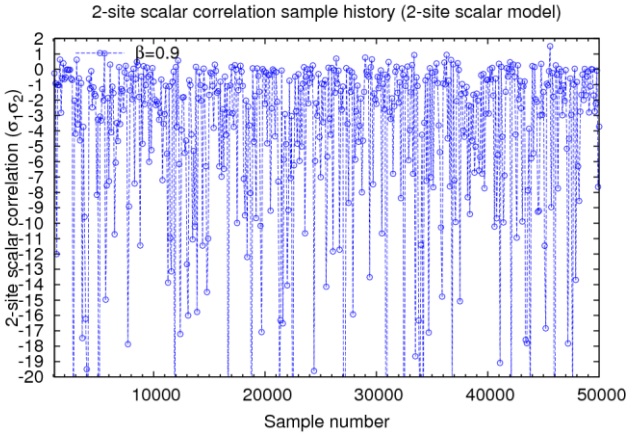




Spin correlation

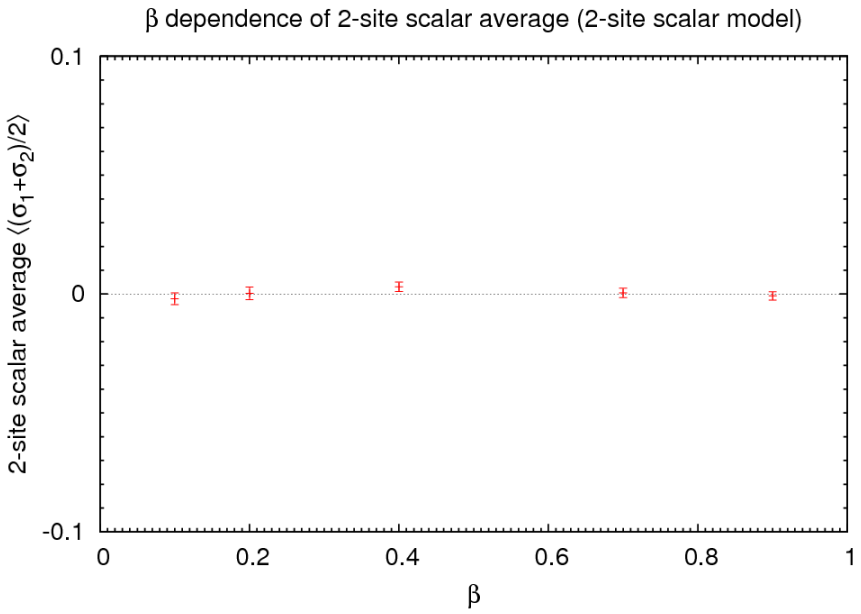




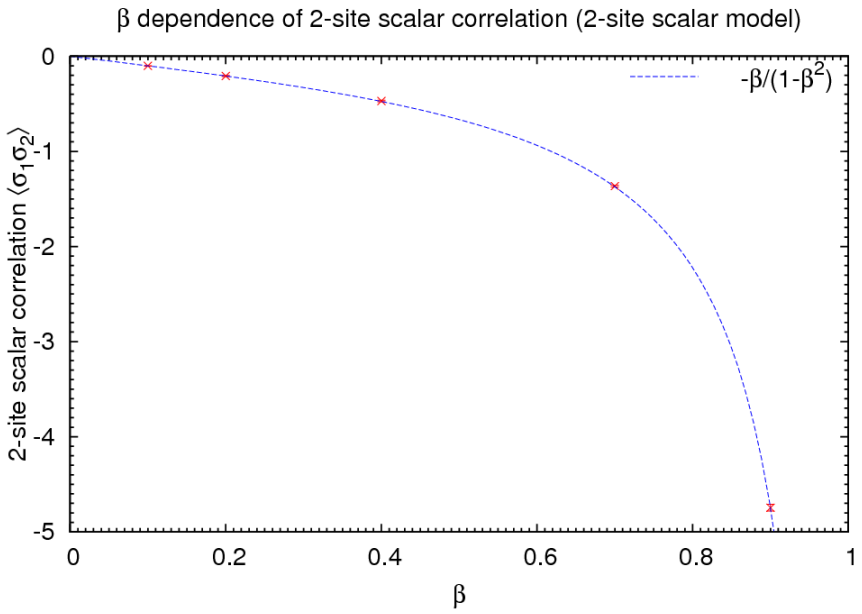


- β dependence of expectation values

Spin average



Spin correlation



- Statistical averaging reproduces theoretical results.
- Acceptance rate is about 90% for this simple model.

6. HMC for LQCD

- For LQCD we can employ HMC algorithm to generate link variable ensemble. However several non-trivial issues still remain.
  - (1) We have to introduce fictitious momentum for link variables for MD evolution. This is not trivial as scalar variables because link variables U take SU(3) values. How can we introduce the fictitious momentum?

$$Z(0) = \int_{-\infty}^{\infty} DU \prod_{f=u,d,s} \det[D_f[U]] \exp\left[-S_G[U]\right] \qquad \Pi \quad \text{for} \quad U \ ?$$

  - The Gauge action is expressed in local form. However the quark determinant term has non-local form/non-trivial dependence on link variables. How can we incorporate the determinant in the MD evolution?
  - I simply give the answers for these questions.

6-1 MD momentum for *U* (SU(3) matrix)

- Consider single *U* for example.
- In order to move in the SU(3) compact phase space, we have to introduce group manifold.
- *U* can be expressed as $U = \exp[iA] \quad A \in su(3),\ \mathrm{Tr}\, A = 0,\ A^\dagger = A$
- We define the time derivative of *U* as $\frac{\partial}{\partial \tau} U = i\Pi U$
- This corresponds to $U(\tau) = T \exp\left[i \int_0^\tau ds \Pi(s) \right] U(0)$
- Where Π is the momentum for A. $\Pi \in su(3),\ \mathrm{Tr}\, \Pi = 0,\ \Pi^\dagger = \Pi$
- We use (Π, U) for the MD evolution instead of (Π, A) although we have introduced Π for A.

- How to derive the equation of motion?
  - The kinetic term is introduced as

$$H(\Pi,U)=\frac{1}{2}\mathrm{Tr}\left[\Pi\Pi\right]+S(U)=\frac{1}{4}\sum_{a=1}^{8}\left(\Pi^{a}\right)^{2}+S(U)$$

  - Where we used the following generator.

$$\Pi=\sum_{a=1}^{8}\Pi^{a}T^{a}\qquad \mathrm{Tr}\left[T^{a}T^{b}\right]=\frac{1}{2}$$

Kinetic term normalization is irrelevant for the MD evolution. This simply changes the scale of fictitious time.

  - The force expression is non-trivial since we use (Π,U) as variables to be evolved. (we need derivatives w.r.t. SU(3) matrix *U.*)
  - To avoid derivatives w.r.t. *U,* we may force Energy conservation low to the Hamiltonian with the definition of momentum.

$$\frac{\partial}{\partial\tau}H\left(\Pi,U\right)=0\quad\text{and}\quad\frac{\partial}{\partial\tau}U=i\Pi U$$

  - From this we can extract the equation of motion for Π .

- HMC for Single SU(3) matrix model

Analytic integration and expressions are known.

  - Partition function:

$$Z(\beta) = \int dU \exp\left[-S(U,\beta)\right]$$

$$S(U,\beta) = -\beta \,\mathrm{Tr}\left[U + U^\dagger\right] = -2\beta \,\mathrm{ReTr}\left[U\right]$$

  - Observables:

$$\left\langle \left(2\,\mathrm{Re\,Tr}[U]\right)^k \right\rangle = \frac{1}{Z(\beta)} \int dU \left(2\,\mathrm{ReTr}[U]\right)^k \exp(-S(U)) = \frac{1}{Z(\beta)} \frac{\partial^k Z(\beta)}{\partial \beta^k}$$

  - HMC

$$Z_{HMC}(\beta) = \int d\Pi dU \exp\left[-H(\Pi,U,\beta)\right]$$

$$H(\Pi,U) = \frac{1}{2}\mathrm{Tr}\left[\Pi\Pi\right] + S(U) = \frac{1}{4}\sum_{a=1}^{8}\left(\Pi^a\right)^2 + S(U)$$

    - Hamiltonian eq. of. M.

$$\frac{\partial}{\partial\tau} H(\Pi,U) = 0 \quad \text{and} \quad \frac{\partial}{\partial\tau} U = i\Pi U$$

$$0 = \frac{\partial}{\partial\tau} H(\Pi,U) = \mathrm{Tr}\left[\Pi\dot{\Pi} - \beta\left(i\Pi U - U^\dagger \Pi^\dagger i\right)\right] = \mathrm{Tr}\left[\Pi\left(\dot{\Pi} - i\beta\left(U - U^\dagger\right)\right)\right]$$

$$\dot{\Pi} = i\beta\left[V - \frac{\mathrm{Tr}[V]}{3}\right] \quad \text{and} \quad \dot{U} = i\Pi U$$

$$V \equiv U - U^\dagger$$

Traceless and Hermitican property of su(3) is imposed to the force computation.

- HMC for Single SU(3) matrix model
  - HMC Molecular dynamics

$$\dot{\Pi} = i\beta\left[V - \frac{\mathrm{Tr}[V]}{3}\right] \quad \text{and} \quad \dot{U} = i\Pi U$$
$$V \equiv U - U^{\dagger}$$

  - Coordinate (U) update:
    - Solve $\dot{U} = i\Pi U$ for $\Delta\tau$ with constant Π approx.

$$Q(\Delta\tau)\begin{pmatrix}\Pi\\ U\end{pmatrix} \equiv \begin{pmatrix}\Pi\\ \exp\left[i\Delta\tau\times\Pi\right]U\end{pmatrix},$$

Numerical evaluation of Matrix exponential is required.
- Taylor expansion method.
- Diagonalization method.

  - Momentum (Π) update:

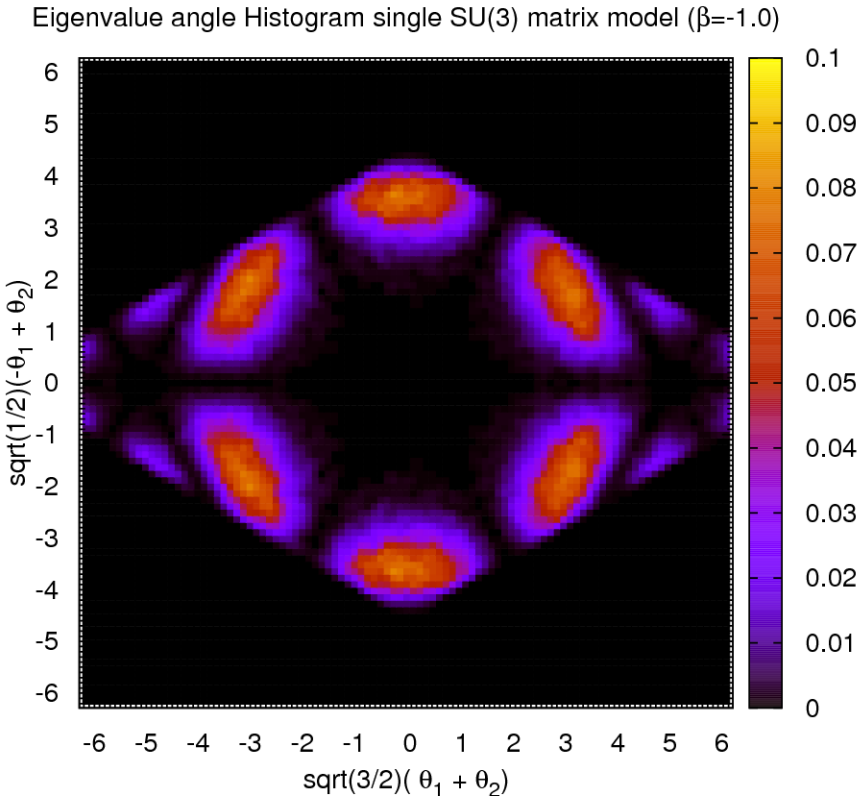
$$P(\Delta\tau)\begin{pmatrix}\Pi\\ U\end{pmatrix} \equiv \begin{pmatrix}\Pi + \Delta\tau\times F\\ U\end{pmatrix}, F = i\beta\left[V - \frac{\mathrm{Tr}[V]}{3}\right], V \equiv U - U^{\dagger}$$

Simply add the force to the momentum as usual.

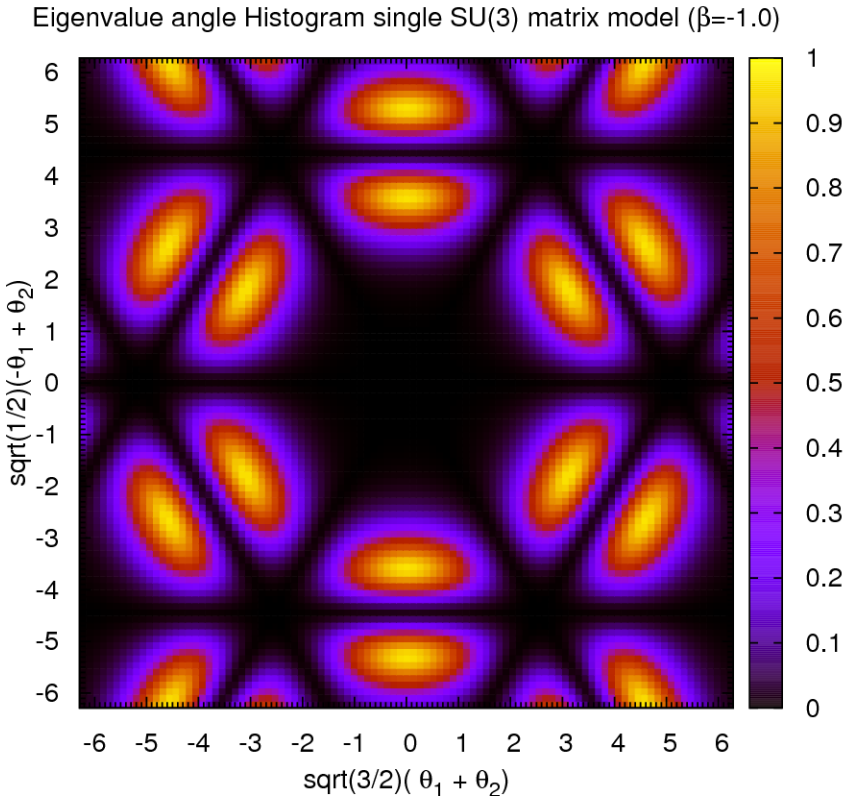
Note: In order to be *U* in SU(3) matrix.
Re-orthonormalization and det[U]=1 condition is forced in the MD evolution.

- HMC for Single SU(3) matrix model
  - Fortran program: [http://theo.phys.sci.hiroshima-u.ac.jp/~ishikawa/ASLFT2010/SingleSU3MatrixHMC.tar.gz]
  - SU(3) matrix eigenvalues phase distribution: (100000 trajs, 10000 samples,10-intervals, N_{MD}=4, tau=1)

HMC measured



Theoretical

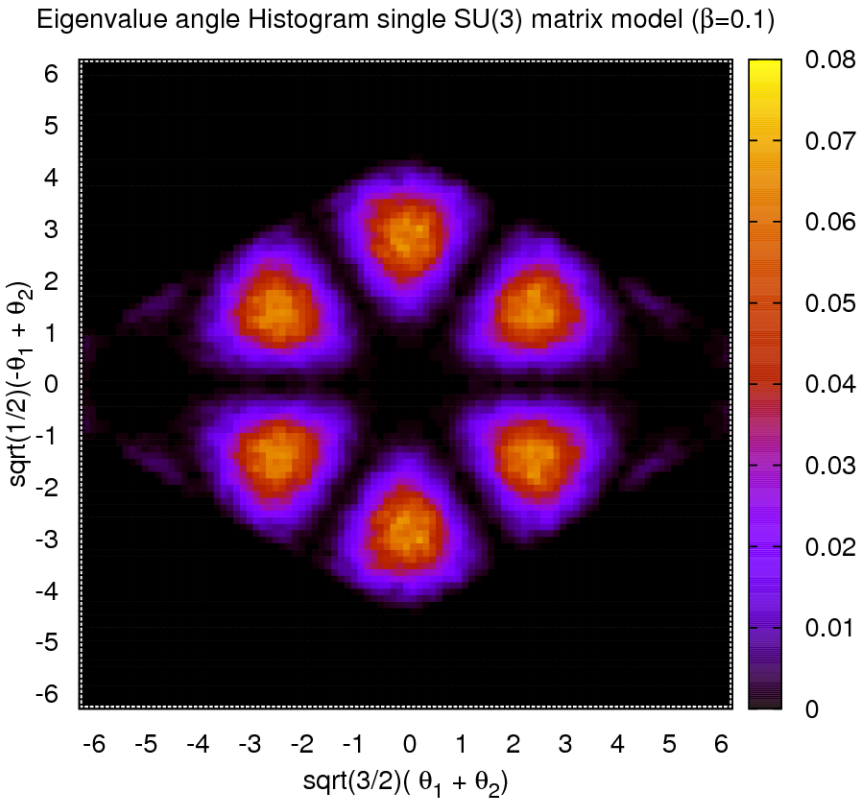


$$(\theta_1, \theta_2, \theta_3), \theta_1 + \theta_2 + \theta_3 = 0$$

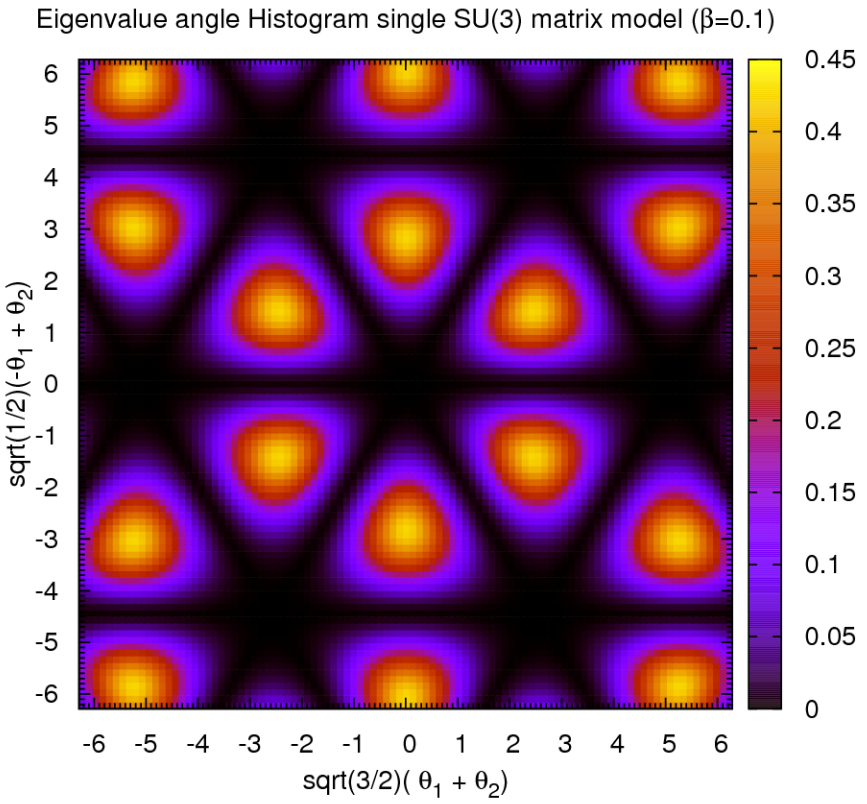
- HMC for Single SU(3) matrix model
  - SU(3) matrix eigenvalues phase distribution:

$$(\theta_1,\theta_2,\theta_3),\theta_1+\theta_2+\theta_3=0$$

HMC measured



Theoretical



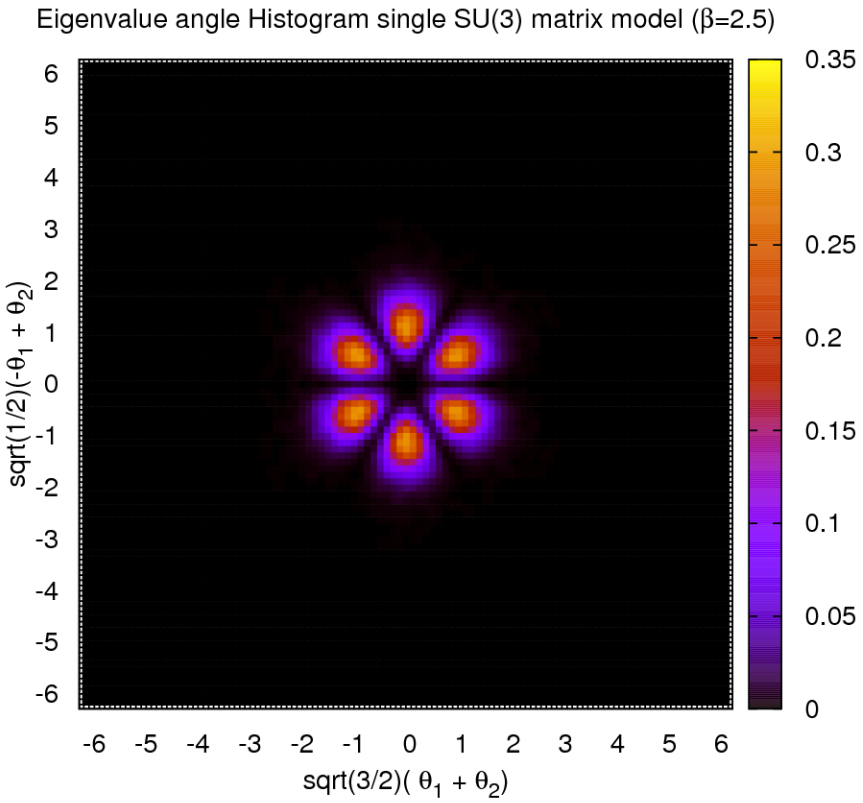
Theoretical distribution via Weyl parameterization.

$$p(\theta_1,\theta_2)\propto \sin^2\left(\frac{\theta_1-\theta_2}{2}\right)\sin^2\left(\frac{2\theta_1+\theta_2}{2}\right)\sin^2\left(\frac{\theta_1+2\theta_2}{2}\right)\exp\left[2\beta\left(\cos\theta_1+\cos\theta_2+\cos(\theta_1+\theta_2)\right)\right]$$

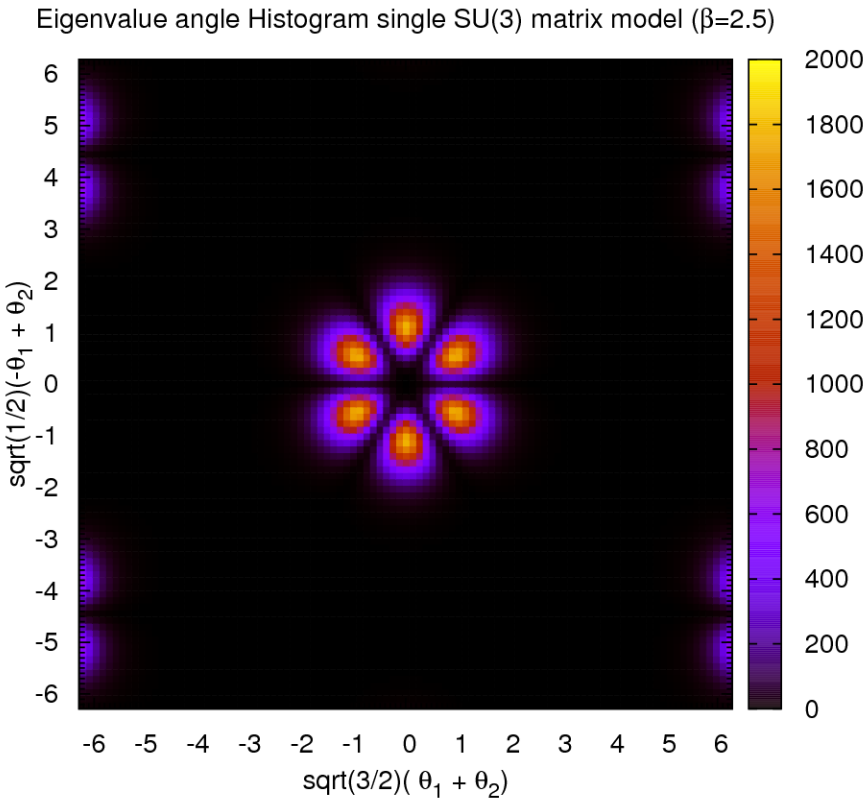
- HMC for Single SU(3) matrix model
  - SU(3) matrix eigenvalues phase distribution:

$$(\theta_1,\theta_2,\theta_3),\theta_1+\theta_2+\theta_3=0$$

HMC measured



Theoretical

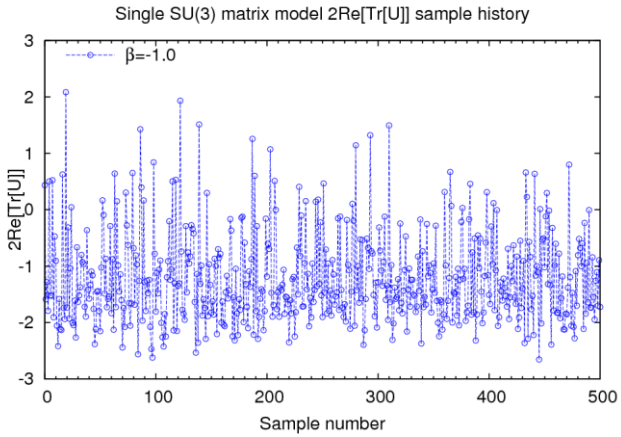


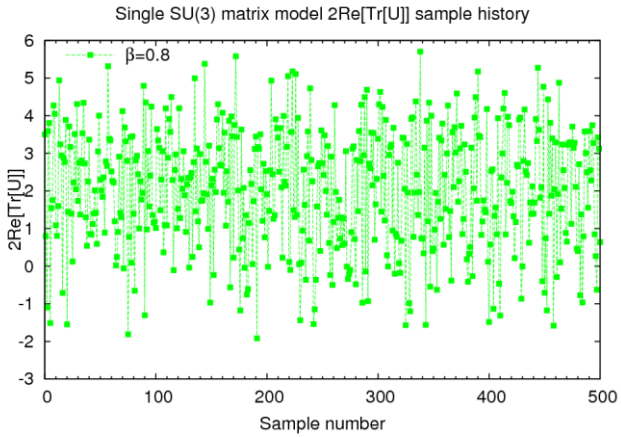
Theoretical distribution via Weyl parameterization.

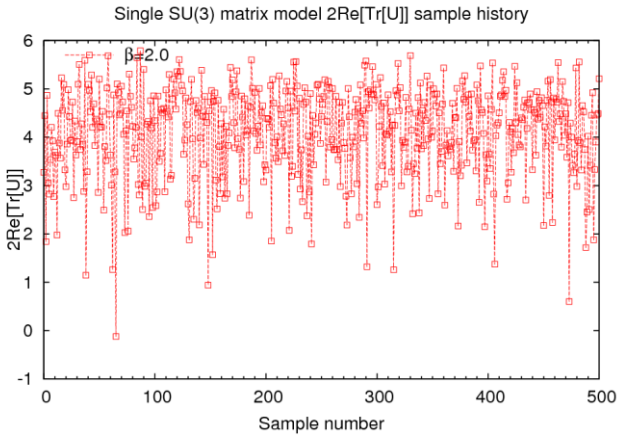
$$p(\theta_1,\theta_2)\propto \sin^2\left(\frac{\theta_1-\theta_2}{2}\right)\sin^2\left(\frac{2\theta_1+\theta_2}{2}\right)\sin^2\left(\frac{\theta_1+2\theta_2}{2}\right)\exp\left[2\beta\left(\cos\theta_1+\cos\theta_2+\cos(\theta_1+\theta_2)\right)\right]$$

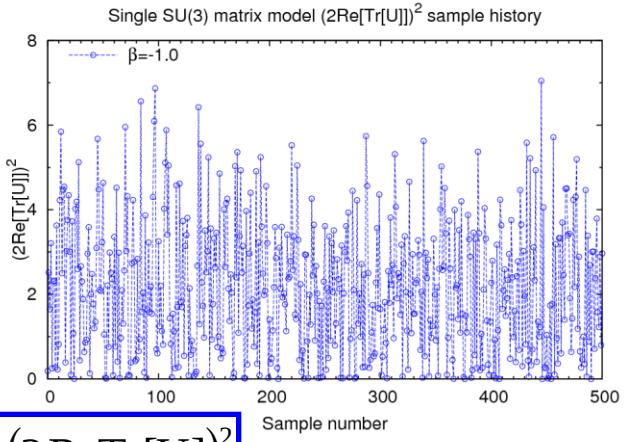
- HMC for Single SU(3) matrix model
  - Oberevables: $2\,\mathrm{ReTr}[U]$, and $(2\,\mathrm{ReTr}[U])^2$
  - HMC history

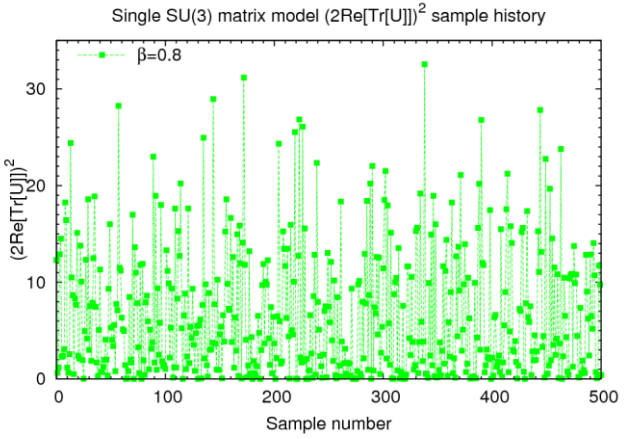
$2\,\mathrm{ReTr}[U]$

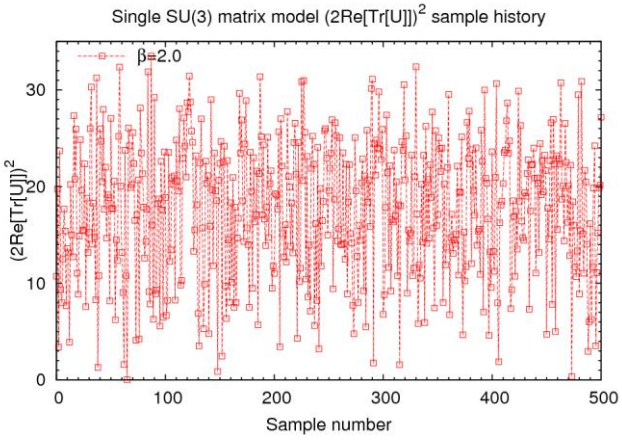








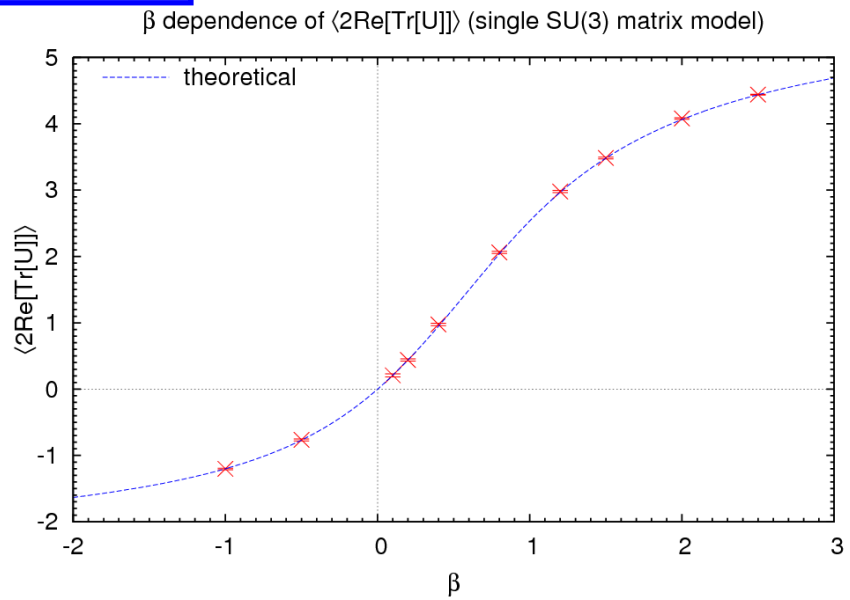




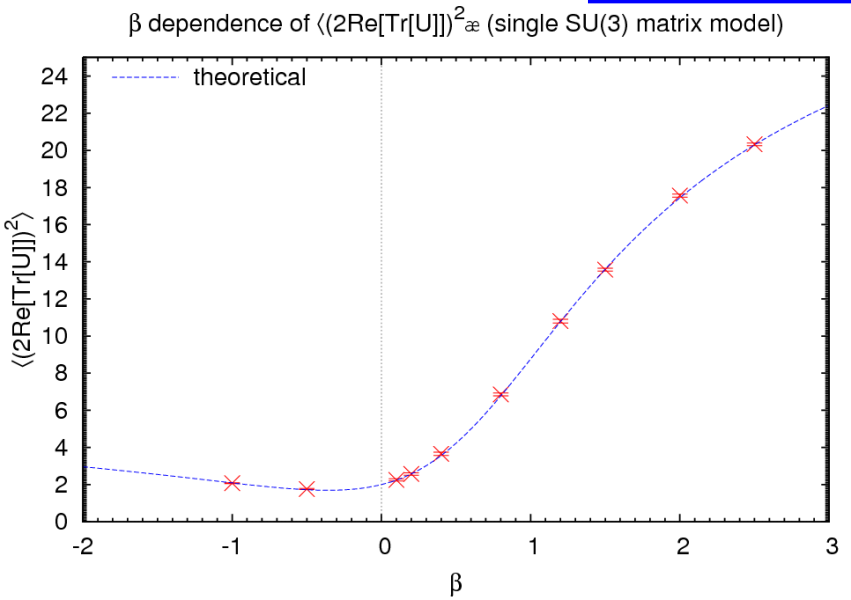
$(2\,\mathrm{ReTr}[U])^2$

- HMC for Single SU(3) matrix model
  - Oberevables: $\left\langle 2\,\mathrm{ReTr}[U]\right\rangle$, and $\left\langle \left(2\,\mathrm{ReTr}[U]\right)^2\right\rangle$
  - Beta dependence of the expectation values.

$\left\langle 2\,\mathrm{ReTr}[U]\right\rangle$



$\left\langle \left(2\,\mathrm{ReTr}[U]\right)^2\right\rangle$



Check the SU(3) matrix related computations in the program

- Projection to Traceless-Hermitan,
- MatrixExp,
- Orthonormalization&det[U]=1 condition, ….

Note that the HMC algorithm flow is identical to that of 2-site scalar model.

How about for LQCD?

- How to derive the equation of motion for LQCD?
  - For Wilson gauge action:

$$H[\Pi,U]=\frac{1}{2}\sum_{n,\mu}\mathrm{Tr}[\Pi_\mu(n)\Pi_\mu(n)]+S_G[U]$$

$$S_G[U]=\beta\sum_n\sum_{\mu>\nu}\left(1-\frac{1}{2N_c}\mathrm{Tr}\left[P_{\mu\nu}(n)+P_{\mu\nu}(n)^\dagger\right]\right)$$

$$P_{\mu\nu}(n)\equiv U_\mu(n)U_\nu(n+\hat{\mu})U_\mu(n+\hat{\nu})^\dagger U_\nu{}^\dagger(n)$$

$$0=\frac{\partial}{\partial\tau}H(\Pi,U)=\sum_{n,\mu}\mathrm{Tr}\left[\Pi_\mu(n)\dot{\Pi}_\mu(n)\right]+\frac{\partial}{\partial\tau}S_G(U)$$

$$=\sum_{n,\mu}\mathrm{Tr}\left[\Pi_\mu(n)\dot{\Pi}_\mu(n)-\Pi_\mu(n)F_\mu(n)\right]=\sum_{n,\mu}\mathrm{Tr}\left[\Pi_\mu(n)\left(\dot{\Pi}_\mu(n)-F_\mu(n)\right)\right]$$

$$\dot{U}_\mu(n)=i\Pi_\mu(n)U_\mu(n)$$

  - We obtain $\dot{\Pi}_\mu(n)=F_\mu(n)$

$$F_\mu(n)=\frac{i\beta}{6}\left(\left[V_\mu(n)-V_\mu(n)^\dagger\right]-\frac{1}{3}\mathrm{Tr}\left[V_\mu(n)-V_\mu(n)^\dagger\right]\right)$$

Traceless condition is forced explicitly for numerical stability

$$V_\mu(n)=\sum_{\nu\neq\mu}\left[U_\mu(n)U_\nu(n+\hat{\mu})U_\mu(n+\hat{\nu})^\dagger U_\nu(n)^\dagger+U_\mu(n)U_\nu(n+\hat{\mu}-\hat{\nu})^\dagger U_\mu(n-\hat{\nu})^\dagger U_\nu(n-\hat{\nu})\right]$$

Problem 7

6-2 Quark Determinant incorporation in the HMC

- The matrix size of quark operator is extremely large. Thus the exact/analytic computation of the Determinant is impossible.

$D[U]$: A Lattice Dirac operator is a complex matrix with dimension : $(3\times 4\times N_T\times N_Z\times N_Y\times N_X)$

Typical DIM = $3x2^{18}$ ~ 800000 = for 16^4 lattice

- Stochastic estimate of the determinant have been used in the HMC for LQCD.
- The most common stochastic method is the "Pseudo-Fermion method".
- Here I Briefly explain the pseudo-fermion method.

- Pseudo-Fermion method.
  - The determinant of a matrix can be expressed by a integral of scalar field.

$$\frac{1}{\det[A]} = C\int d\vec{\phi}^{\dagger} d\vec{\phi}\, \exp\left[-\vec{\phi}^{\dagger} A \vec{\phi}\right]$$

$\vec{\phi}$ Complex scalar field(=Complex vecotr)

A Complex Matrix (Non-singular)

  - If *A* is Hermitian and positive $\exp\left[-\vec{\phi}^{\dagger} A \vec{\phi}\right]$ can be treated as a probability distribution for $(\vec{\phi}^{\dagger}, \vec{\phi})$.
  - For Quark determinant with 2-flavor (up,down) with degenerate masses. We can transform the determinant to the integration of complex scalar variables.

- Pseudo-Fermion method.

$$\det[D_u]\det[D_d] = \det[D]\det[D] = \det[D]\left(\det[D]^*\right)$$
$$= \left|\det[D]\right|^2 = \det\left[\left(DD^\dagger\right)\right]$$
$$= C\int d\vec{\phi}^\dagger d\vec{\phi}\exp\left[-\vec{\phi}^\dagger\left(DD^\dagger\right)^{-1}\vec{\phi}\right]$$
$$= C\int d\vec{\phi}^\dagger d\vec{\phi}\exp\left[-\left|\left(D\right)^{-1}\vec{\phi}\right|^2\right]$$

  - Where $D^\dagger = \gamma_5 D\gamma_5,\ \left(\gamma_5\right)^2 = 1,\ \det[AB] = \det[BA]$ are used.
  - $(\vec{\phi}^\dagger, \vec{\phi})$ are called as the Pseudo-fermion field.
  - The pseudo-fermion field is incorporated in the random variables. Thus we have to generate the ensemble for $(\Pi, U, \vec{\phi}^\dagger, \vec{\phi})$ ..
  - In order for $\exp\left[-\left|D^{-1}\vec{\phi}\right|^2\right]$ to be a probability, we have used the positivity and realnelss of (det[D]det[D]) using two quarks with the identical masses. For odd-number flavor simulations tricks are required to introduce the pseudo-fermion. (I will skip this issue. See Polynomial-HMC, Rational-HMC)

- HMC algorithm for Two-flavor LQCD

$$\left\langle O[U, D[U]^{-1}] \right\rangle = \frac{1}{Z[0]} \int \prod_n d\phi^\dagger(n) d\phi(n) \prod_{n,\mu} dU_\mu(n)$$
$$O[U, D[U]^{-1}] \ \exp\left[-S_G[U] - S_{PF}[U, \phi^\dagger, \phi,] \right]$$

$$S_{PF}[U, \phi^\dagger, \phi] \equiv \sum_n \left| \sum_m \left(D[U] \right)^{-1}{}_{(n,m)} \phi(m) \right|^2 = \left| \left(D[U] \right)^{-1} \vec{\phi} \right|^2$$

  - HMC algorithm is applied to $S_G[U] + S_{PF}[U, \phi^\dagger, \phi]$.
    - The momentum Π is introduced only for U .
    - $(\vec{\phi}^\dagger, \vec{\phi})$ are treated as a auxiliary field and not evolved during the MD. $(\vec{\phi}^\dagger, \vec{\phi})$ are generated directly by the disitribution

$$\exp\left[-\left| D^{-1} \vec{\phi} \right|^2 \right]$$

$$\phi = D[U]\eta, \quad \text{with Gaussian random number } \eta.$$

$$\text{Prob}(\eta^\dagger, \eta) \propto \exp\left[-|\eta|^2 \right]$$

• HMC algorithm for Two-flavor LQCD

$$S_{PF}[U,\phi^\dagger,\phi] \equiv \left|\left(D[U]\right)^{-1}\vec{\phi}\right|^2$$

$$H[\Pi,U,\phi^\dagger,\phi] = \frac{1}{2}\sum_{n,\mu}\mathrm{Tr}\left[\Pi_\mu(n)\Pi_\mu(n)\right] + S_G[U] + S_{PF}[U,\phi^\dagger,\phi]$$

(Step 0) Generate initial state U

(Step 1) Generate initial momentum Π from Gaussian dist.

Generate pseudo-fermion field ϕ from $\phi = D[U]\eta$ with Gaussian dist'd η.

(Step 2) MD evolution to get (Π', U') from (Π, U)

(Step 3) (flip momentum $\Pi' = -\Pi'$)

(Step 4) Do Metropolis test with prob. $P = \min(1, e^{-H'+H})$

If accepted $(U = U')$ and add (U) to the ensemble.

If rejected add (U) to the ensemble.

(Step 5) Goto Step 1

- Note 1: We have new contribution from S_{PF} term in the MD evolution. The force computation is required.
- Note 2: We need inversion computation $\left(D[U]\right)^{-1}\vec{\phi}$ to evaluate S_{PF} value.

- # Pseudo-Fermion Contribution to the MD.
  - We can derive the MD force expression via $0 = \frac{\partial}{\partial \tau} H[\Pi, U, \phi^\dagger, \phi]$
  - as before.
  - Evaluating $$\frac{\partial}{\partial \tau} S_{PF}[U, \phi^\dagger, \phi] \equiv \frac{\partial}{\partial \tau} \left| \left(D[U]\right)^{-1} \vec{\phi} \right|^2$$
  - We have $$\frac{\partial}{\partial \tau} \left| \left(D[U]\right)^{-1} \vec{\phi} \right|^2 = -\left[X^\dagger \left(D[U]\right)^{-1} \frac{\partial D[U]}{\partial \tau} X \right] - [h.c.] = -\left[\mathrm{Tr} \left[\frac{\partial D[U]}{\partial \tau} X X^\dagger \left(D[U]\right)^{-1} \right] \right] - [h.c.]$$
    $$= -\left[\mathrm{Tr} \left[i\Pi_\mu U_\mu \left(\frac{\partial D[U]}{\partial U_\mu} \right) X Y^\dagger \right] \right] - [h.c.] = \mathrm{Tr}\left[-i\Pi_\mu F_{PF\mu} \right]$$
  - where $$X \equiv \left(D[U]\right)^{-1} \vec{\phi}, \quad Y \equiv \left(D[U]^\dagger\right)^{-1} \left(D[U]\right)^{-1} \vec{\phi} = \gamma_5 \left(D[U]\right)^{-1} \gamma_5 \left(D[U]\right)^{-1} \vec{\phi}$$
  - Thus we need two-inversion computation to evaluate the MD force from the pseudo-fermion at every time step.

$$F_{PF\mu} = F_{PF\mu}[U, X, Y]$$

Problem 8

This is the most time consuming part of LQCD simulations

- In the last lecture I will explain the methods to invert the lattice Dirac operator *D[U].*
- However the explanation using the explicit form of D[U] is rather tedious. I will explain the common part of the algorithm to invert more simple lattice discretised differential operators.
- I will employ the Poisson equation on discretized space(=3D/2D/1D lattices).

$$D[U]\phi = \eta \Leftrightarrow -\Delta\phi = \rho$$

Lattice Dirac equation with source term

Lattice Poisson equation with source term

$$\phi = D[U]^{-1}\eta \Leftrightarrow \phi = -\Delta^{-1}\rho$$

Problems

(1) Check that the Leapfrog scheme has the error term explained in [page 19].

(2) Check that the Leapfrog scheme satisfies the area conservation low [page 20].

(3) Check the detailed balance condition for the HMC transition probability described in [page 23].

(4) Evaluate the eigenvalues of the MD transition matrix for a Harmonic Oscillator [page 26]. When does the evolution become unstable?

(5) Get and compile the 2-Site Scalar model. Check the result numerically. [page 28-35]

(6) Get and compile the Single SU(3) matrix model. Check the result numerically. [page 39-45]

(7) Derive the MD force expression for the Wilson gauge action (quenched LQCD) [page 46].

(8) [Advanced] Derive the MD force from the pseudo-fermion part with the Wilson-Dirac action [page 52][page 4 for the explicit form of $D[U]$].

Backups

- SU(3) matrix from su(3) matrix.
  - We need to compute SU(3) matrix from su(3) Hermitian traceless matrix via matrix exponential form.

$$V = \exp\left[iaH\right]$$

$$H: \quad H^{\dagger} = H \text{ and } \mathrm{Tr}[H] = 0$$

$$a: \quad \text{real parameter}$$

  - There are Two categories to compute this
    - (1) method based on Eigen decomposition.
    - (2) method based on Taylor expansion.

Backups

- (1) method based on Eigen decomposition.
  - *H* can be diagonalized and has real eigen values.
  - The characteristic polynomial is

$$\det[H-\lambda] = -\lambda^3 + x\lambda + y \qquad x = \frac{Tr[H^2]}{2}, \quad y = \frac{Tr[H^3]}{3} \in R$$

  - We can diagonalize as $H = Q^\dagger \Lambda Q$
  - We can compute the matrix exponential as

$$V = Q^\dagger \exp[ia\Lambda] Q = Q^\dagger \begin{pmatrix} e^{ia\lambda_1} & & \\ & e^{ia\lambda_2} & \\ & & e^{-ia(\lambda_1+\lambda_2)} \end{pmatrix} Q^\dagger$$

  - This method needs some care when a pair of the eigenvalues nearly degenerates. To avoid loss of significant digit in the computetion some special formula should be used.

Backups

- (2) method based on Taylor expansion.
  - Any matrix exponential are defined as

$$V = \exp\left[iaH\right] = \sum_{j=0}^{\infty} \frac{\left(iaH\right)^j}{j}$$

  - We can truncate the series at

$$V = \exp\left[iaH\right] \approx \sum_{j=0}^{N_{\exp}} \frac{\left(iaH\right)^j}{j} \qquad \frac{\left|aH\right|^{N_{\exp}}}{N_{\exp}!} < \varepsilon$$

  - However |aH|>1 case the loss of significant digit occurs when computing the series even if we use Horner's method.
  - We make use of the ideintity:

$$\exp\left[iaH\right] = \left(\exp\left[\frac{iaH}{2^m}\right]\right)^{2^m}$$

$V = 1$

do $j = N_{\exp}$,1,-1

$V = 1 + \frac{ia}{j} HV$

enddo

Backups

- (2) method based on Taylor expansion.
  - We make use of the ideintity:

$$\exp\left[iaH\right]=\left(\exp\left[\frac{iaH}{2^m}\right]\right)^{2^m}$$

  - Divide and Squaerd method.

$$\text{determin}\ \ m\ \ \text{s.t.}\ \ \left|\frac{iaH}{2^m}\right|<\frac{1}{2}$$

  - 12^{th}-14^{th} order expansion

is sufficient for the approx.
However this part can be
further economized as follows.

$$V=\text{Approx}\left(\exp\left[\frac{iaH}{2^m}\right],N_{\exp}=12\sim14\right)$$

$$\text{do}\ \ j=1,m$$

$$V=V^2$$

$$\text{enddo}$$

Backups

- (2) method based on Taylor expansion.
  - From the characteristic polynomial, we have

$$\det[H-\lambda] = -\lambda^3 + x\lambda + y \qquad x = \frac{Tr[H^2]}{2}, \quad y = \frac{Tr[H^3]}{3} \in R$$

  - Cayley-Hamilton theorem leads

$$-H^3 + xH + y = 0$$

  - Thus any analytic function of H (f(H)) should have the following form.

$$f(H) = c_2 H^2 + c_1 H + c_0$$

  - These coefficients {c_i} are computed via a kind of modulo computation for polynomial.

$$f(H) = \sum_{j=-\infty}^{\infty} f_j H^j = c_2 H^2 + c_1 H + c_0 \mod H^3 - xH - y$$

Backups

- (2) method based on Taylor expansion.
  - Instead of doing modulo computation for infinite series we compute

$$H^j = d_2^{(j)} H^2 + d_1^{(j)} H + d_0^{(j)} \mod H^3 - xH - y$$

  - {d's} are computed as follows,
  - Define the 2nd order polynomial of H as a vector expressed by the coefficients.

$$\{d_0^{(j)}, d_1^{(j)}, d_2^{(j)}\} = \begin{pmatrix} d_0^{(j)} \\ d_1^{(j)} \\ d_2^{(j)} \end{pmatrix} \Leftrightarrow d_0^{(j)} + d_1^{(j)} H + d_2^{(j)} H^2 = H^j$$

This is a coefficient domain representation of a su(3) matrix function.

  - We can evaluate next H^(j+1) from using the Cayley-Hamiton theorem.

Backups

- (2) method based on Taylor expansion.
  - Instead of doing modulo computation for infinite series we compute

$$H^j = d_2^{(j)} H^2 + d_1^{(j)} H + d_0^{(j)} \mod H^3 - xH - y$$

$$H^{j+1} = d_0^{(j+1)} + d_1^{(j+1)} H + d_2^{(j+1)} H^2 = H \times \left(d_0^{(j)} + d_1^{(j)} H + d_2^{(j)} H^2 \right)$$
$$= d_0^{(j)} H + d_1^{(j)} H^2 + d_2^{(j)} H^3 \underset{\lambda_1}{=} y d_2^{(j)} + (d_0^{(j)} + x d_2^{(j)}) H + d_1^{(j)} H^2$$

$$\vec{d}^{(j+1)} = \begin{pmatrix} d_0^{(j+1)} \\ d_1^{(j+1)} \\ d_2^{(j+1)} \end{pmatrix} = \begin{pmatrix} y d_2^{(j)} \\ d_0^{(j)} + x d_2^{(j)} \\ d_1^{(j)} \end{pmatrix} = \begin{pmatrix} & & y \\ 1 & & x \\ & 1 & \end{pmatrix} \begin{pmatrix} d_0^{(j)} \\ d_1^{(j)} \\ d_2^{(j)} \end{pmatrix} = \breve{H} \vec{d}^{(j)}$$

  - Thus we have

$$\vec{d}^{(j)} = \breve{H}^j \vec{d}^{(0)}, \qquad \vec{d}^{(0)} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

Backups

- (2) method based on Taylor expansion.
  - The matrix exponential is now expressed as

$$\exp[iaH] = \sum_{j=0}^{\infty} \frac{(iaH)^j}{j!} = \sum_{j=0}^{\infty} \frac{(ia)^j}{j!} \left(d_2{}^{(j)} H^2 + d_1{}^{(j)} H + d_0{}^{(j)} \right)$$

$$= \left(\sum_{j=0}^{\infty} \frac{(ia)^j d_2{}^{(j)}}{j!} \right) H^2 + \left(\sum_{j=0}^{\infty} \frac{(ia)^j d_1{}^{(j)}}{j!} \right) H^2 + \left(\sum_{j=0}^{\infty} \frac{(ia)^j d_0{}^{(j)}}{j!} \right)$$

$$= \left(1, H, H^2\right) \begin{pmatrix} \sum_{j=0}^{\infty} \frac{(ia)^j d_0{}^{(j)}}{j!} \\ \sum_{j=0}^{\infty} \frac{(ia)^j d_1{}^{(j)}}{j!} \\ \sum_{j=0}^{\infty} \frac{(ia)^j d_2{}^{(j)}}{j!} \end{pmatrix} = \left(1, H, H^2\right) \sum_{j=0}^{\infty} \frac{(ia)^j}{j!} \vec{d}^{(j)}$$

$$= \left(1, H, H^2\right) \sum_{j=0}^{\infty} \frac{(ia)^j}{j!} \breve{H}^j \vec{d}^{(0)} = \left(1, H, H^2\right) \exp\left[ia\breve{H}\right] \vec{d}^{(0)}$$

Backups

- (2) method based on Taylor expansion.
  - The computation of $\exp\left[ia\breve{H}^j\right]\vec{d}^{(0)}$ is much easier and economical than $V=\exp[iaH]$ using Taylor series,

because $\breve{H}$ is real and sparse. And we need a vector multiplied form of $\breve{H}\vec{v}$ in this representation. Finally we have

$$V=\exp[iaH]=e_2H^2+e_1H+e_0=\left(1,H,H^2\right)\vec{e}$$

$$\vec{e}=\exp\left[ia\breve{H}^j\right]\vec{d}^{(0)}\approx\sum_{j=0}^{N_{\exp}}\frac{\left(ia\breve{H}\right)^j\vec{d}^{(0)}}{j!}$$

$$\breve{H}\equiv\begin{pmatrix}&&y\\1&&x\\&1&\end{pmatrix}\qquad x=\frac{Tr[H^2]}{2},\quad y=\frac{Tr[H^3]}{3}\in R$$

$\vec{d}^{(0)}=(1,0,0)^T$

$\vec{e}=\vec{d}^{(0)}$

do $j=N_{\exp},1,-1$

$\vec{e}=\vec{d}^{(0)}+\frac{ia}{j}\breve{H}\vec{e}$

enddo