SUSY Yang-Mills on the Lattice

Alexander Ferling

November 21, 2005

the considered action

- goal: estimate the expectation values

$$\langle A \rangle = Z^{-1} \int [d\phi] \, e^{-S[\phi]} A[\phi]$$

the considered action

- goal: estimate the expectation values

$$\langle A \rangle = Z^{-1} \int [d\phi]\, e^{-S[\phi]} A\,[\phi]$$

- the lattice action

$$S_{lat} = S_g + S_f$$

the considered action

- the continuum action

$$S_{SYM} = \int d^4x \left\{ \frac{1}{4} F^a_{\mu\nu}(x) F^a_{\mu\nu}(x) + \frac{1}{2} \overline{\lambda}^a(x) \gamma_\mu D_\mu \lambda^a(x) \right\}$$

- gauge part

$$S_g[U] = \beta \sum_x \sum_{\mu\nu} \left[1 - \frac{1}{N_c} ReTrU_{\mu\nu} \right]$$

- fermionic part

$$\begin{aligned} S_f\left[U, \overline{\lambda}, \lambda\right] \quad = \quad & \frac{1}{2} \sum_x \overline{\lambda}(x)\lambda(x) \\ & + \frac{\kappa}{2} \sum_x \sum_\mu \left[\overline{\lambda}(x+\hat{\mu}) V_\mu(x)(r+\gamma_\mu)\lambda(x) \right. \\ & \left. + \overline{\lambda}(x) V^T_\mu(x) \, (r-\gamma_\mu)\lambda(x+\hat{\mu}) \right] \end{aligned}$$

the involved magnitudes

- the bare coupling

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

- gauge field link in the adjoint representation

$$[V_\mu(x)]_{ab} \equiv 2Tr\left[U_\mu^\dagger(x)T^a U_\mu(x)T^b\right] = \left[V_\mu^*(x)\right]_{ab} = \left[V_\mu^T(x)\right]_{ab}^{-1}$$

- the generators T^a in the $SU(2)$ case

$$T^a = \frac{1}{2}\tau^a$$

- the majorana fermions

$$\lambda = \lambda^{\mathcal{C}} = \mathcal{C}\overline{\lambda}^T$$

talking about κ

- gluino breaks supersymmetry

$$\mathcal{L} = \mathcal{L}_{SYM} + m_{\tilde{g}} \bar{\lambda} \lambda$$

- the hopping parameter

$$\kappa = (2m_0 + 8r)^{-1}$$

- $\rightarrow$ bare gluino mass, breaks chiral invariance

$$m_{\tilde{g},0} \propto \kappa^{-1}$$

- $\rightarrow$ tune κ to a critical κ_c, so the renormalized mass

$$m_{\tilde{g}} \rightarrow 0$$

the fermion matrix $Q_{x,y}$

- back to the fermion action

$$
\begin{aligned}
S_f\left[U,\overline{\lambda},\lambda\right] &= \frac{1}{2}\sum_x \overline{\lambda}(x)\lambda(x) \\
&+\frac{\kappa}{2}\sum_x\sum_\mu \left[\overline{\lambda}(x+\hat{\mu})V_\mu(x)(r+\gamma_\mu)\lambda(x)\right. \\
&\left.+\overline{\lambda}(x)V_\mu^T(x)\,(r-\gamma_\mu)\lambda(x+\hat{\mu})\right]
\end{aligned}
$$

the fermion matrix $Q_{x,y}$

- back to the fermion action

$$
\begin{aligned}
S_f\left[U,\overline{\lambda},\lambda\right] \;=\; & \frac{1}{2}\sum_x \overline{\lambda}(x)\lambda(x) \\
& +\frac{\kappa}{2}\sum_x\sum_\mu \left[\overline{\lambda}(x+\hat{\mu})V_\mu(x)(r+\gamma_\mu)\lambda(x)\right. \\
& \left.+\overline{\lambda}(x)V_\mu^T(x)\,(r-\gamma_\mu)\lambda(x+\hat{\mu})\right]
\end{aligned}
$$

- $\rightarrow$ the fermion matrix

$$
\begin{aligned}
Q_{x,y}\left[U\right] \equiv \delta_{x,y} - \kappa\sum_\mu & \left[\delta_{y,x+\hat{\mu}}(1+\gamma_\mu)V_\mu(x)\right. \\
& \left.+\delta_{y+\hat{\mu}}(1-\gamma_{y+\hat{\mu}})V_\mu^T(y)\right]
\end{aligned}
$$

the fermion matrix $Q_{x,y}$

- from this, we can write compactly

$$S_f = \frac{1}{2}\sum_{xy} \overline{\lambda}(x) Q_{x,y} \lambda(y)$$

- $\rightarrow$ the fermion matrix

$$\int [d\lambda]\, e^{-S_f} = \int [d\lambda]\, e^{-\frac{1}{2}\overline{\lambda} Q \lambda} = \pm\sqrt{\det Q}$$

- because of the pfaffian

$$\begin{aligned} pf(\mathcal{M}) &\equiv \frac{1}{N!2^N} \epsilon_{\alpha_1\beta_1 \ldots \alpha_N\beta_N} \mathcal{M}_{\alpha_1\beta_1} \ldots \mathcal{M}_{\alpha_N\beta_N} \\ &= \int [d\lambda_i]\, e^{-\frac{1}{2}\lambda_\alpha \mathcal{M}_{\alpha\beta}\lambda_{beta}} \end{aligned}$$

single step approximation

- the polynomial approximation relies on

$$|\det Q|^{N_f} = \left[\det (Q^\dagger Q)\right]^{\frac{N_f}{2}} \approx \lim_{n\to\infty} \left[\det P(\tilde{Q}^2)\right]^{-1}$$

with $\tilde{Q}^2 = Q^\dagger Q$

single step approximation

- the polynomial approximation relies on

$$|\det Q|^{N_f} = \left[\det(Q^\dagger Q)\right]^{\frac{N_f}{2}} \approx \lim_{n\to\infty} \left[\det P(\tilde{Q}^2)\right]^{-1}$$

  with $\tilde{Q}^2 = Q^\dagger Q$

- where the polynomial $P_n(x)$ satisfies

$$\lim_{n\to\infty} P_n(x) = x^{-\frac{N_f}{2}} \quad \text{for} \quad x \in [\epsilon, \lambda]$$

single step approximation

- the polynomial approximation relies on

$$|\det Q|^{N_f} = \left[\det(Q^\dagger Q)\right]^{\frac{N_f}{2}} \approx \lim_{n\to\infty} \left[\det P(\tilde{Q}^2)\right]^{-1}$$

  with $\tilde{Q}^2 = Q^\dagger Q$

- where the polynomial $P_n(x)$ satisfies

$$\lim_{n\to\infty} P_n(x) = x^{-\frac{N_f}{2}} \quad \text{for} \quad x \in [\epsilon, \lambda]$$

- and

$$\epsilon \leq \min \mathrm{spec}(Q^\dagger Q)$$
$$\lambda \geq \max \mathrm{spec}(Q^\dagger Q)$$

single step approximation

- using roots of the polynomial r_j

$$P_n(Q^\dagger Q) = P_n(\tilde{Q}) = r_0 \prod_{j=1}^{n} (\tilde{Q}^2 - r_j)$$

whith $r_j \equiv \rho^* \rho \equiv (\mu_j + i\nu_j)^2$, it follows

$$P_n(\tilde{Q}) = r_0 \prod_{j=1}^{n} ((\tilde{Q} - \rho_j^*)(\tilde{Q} - \rho_j))$$

single step approximation

- using roots of the polynomial r_j

$$P_n(Q^\dagger Q) = P_n(\tilde{Q}) = r_0 \prod_{j=1}^{n} (\tilde{Q}^2 - r_j)$$

whith $r_j \equiv \rho^* \rho \equiv (\mu_j + i\nu_j)^2$, it follows

$$P_n(\tilde{Q}) = r_0 \prod_{j=1}^{n} ((\tilde{Q} - \rho_j^*)(\tilde{Q} - \rho_j))$$

- the multi-boson representation of the fermion determinant

$$r_0 \prod_{j=1}^{n} (\mathrm{det}(\tilde{Q} - \rho_j^*)(\tilde{Q} - \rho_j))^{-1}$$

$$\propto \int \mathcal{D}[\Phi] e^{-\sum_{j=1}^{n} \sum_{xy} \Phi_j^\dagger(y) \left[(\tilde{Q}-\rho_j^*)(\tilde{Q}-\rho_j)\right]_{xy} \Phi_j(x)}$$

two-step multi-bosonic scheme

- problem: small fermion masses
  $\rightarrow$ hugh condition-number $\frac{\lambda}{\epsilon} \sim \mathcal{O}(10^4 - 10^6)$

two-step multi-bosonic scheme

- problem: small fermion masses
  $\rightarrow$ hugh condition-number $\frac{\lambda}{\epsilon} \sim \mathcal{O}(10^4 - 10^6)$
- the key:

$$\lim_{n_2 \to \infty} P_{n_1}^{(1)}(x) P_{n_2}^{(2)}(x) = x^{-\frac{N_f}{2}}$$

two-step multi-bosonic scheme

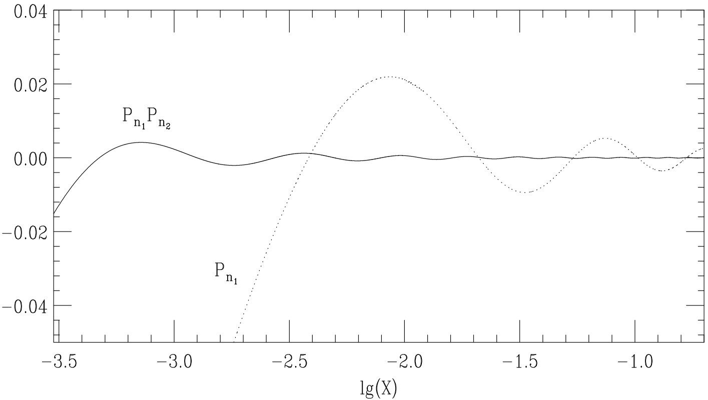
- problem: small fermion masses
  $\rightarrow$ hugh condition-number $\frac{\lambda}{\epsilon} \sim \mathcal{O}(10^4 - 10^6)$
- the key:

$$\lim_{n_2 \to \infty} P^{(1)}_{n_1}(x) P^{(2)}_{n_2}(x) = x^{-\frac{N_f}{2}}$$

- we get

$$|\det(Q)|^{N_f} \simeq \frac{1}{\det P^{(1)}_{n_1}(\tilde{Q}^2) \det P^{(2)}_{n_2}(\tilde{Q}^2)}$$

two-step multi-bosonic scheme



- relative deviation of the successive polynomial approximation

the noisy correction

- test this approximation $\rightarrow$ fulfill the detailed balance

$$\frac{P(U \rightarrow U')}{P(U' \rightarrow U)} = \frac{\exp -(S_g\,[U'] + \log\left[\det \tilde{Q}\right]^{\frac{N_f}{2}})}{\exp -(S_g\,[U] + \log\left[\det \tilde{Q}\right]^{\frac{N_f}{2}})}$$

$$= \frac{\det\left(\tilde{Q}^{N_f}[U']P^{(1)}_{n_1}\tilde{Q}^2[U']\right)}{\det\left(\tilde{Q}^{N_f}[U]P^{(1)}_{n_1}\tilde{Q}^2[U]\right)}\,\frac{e^{S^{(n_1)}[U',\phi^\dagger,\phi]}}{e^{S^{(n_1)}[U,\phi^\dagger,\phi]}}$$

the noisy correction

- test this approximation $\rightarrow$ fulfill the detailed balance

$$\frac{P(U \rightarrow U')}{P(U' \rightarrow U)} = \frac{\exp -\left(S_g\left[U'\right] + \log \left[\det \tilde{Q}\right]^{\frac{N_f}{2}}\right)}{\exp -\left(S_g\left[U\right] + \log \left[\det \tilde{Q}\right]^{\frac{N_f}{2}}\right)}$$

$$= \frac{\det\left(\tilde{Q}^{N_f}[U'] P^{(1)}_{n_1} \tilde{Q}^2[U']\right)}{\det\left(\tilde{Q}^{N_f}[U] P^{(1)}_{n_1} \tilde{Q}^2[U]\right)} \frac{e^{S^{(n_1)}[U',\phi^\dagger,\phi]}}{e^{S^{(n_1)}[U,\phi^\dagger,\phi]}}$$

- since the update polynomial $P^{(1)}_{n_1}$ fulfills

$$\frac{P_\phi(U \rightarrow U')}{P_\phi(U' \rightarrow U)} = \frac{e^{S^{(n_1)}[U',\phi^\dagger,\phi]}}{e^{S^{(n_1)}[U',\phi^\dagger,\phi]}}$$

- we can use as an acceptance probability

$$P_{NC}\left(U \rightarrow U'\right) = \min\left(1, \frac{\det\left(\tilde{Q}^{N_f}[U'] P^{(1)}_{n_1} \tilde{Q}^2[U']\right)}{\det\left(\tilde{Q}^{N_f}[U] P^{(1)}_{n_1} \tilde{Q}^2[U]\right)}\right)$$

the noisy correction

- following the idea of the multi-boson algorithm we will approximate this new determinant by another polynomial $P_{n_2}^{(2)}$

$$\left(\det \tilde{Q}^2\right)^{\frac{N_f}{2}} \det P_{n_1}^{(1)}(\tilde{Q}^2) \simeq \frac{1}{\det P_{n_1}^{(1)}(\tilde{Q}^2)} = \int \mathcal{D}[\eta^\dagger]\mathcal{D}[\eta] e^{\eta^\dagger P_{n_2}^{2}(\tilde{Q}^2)\eta}$$

the noisy correction

- following the idea of the multi-boson algorithm we will approximate this new determinant by another polynomial $P^{(2)}_{n_2}$

$$\left(\det \tilde{Q}^2\right)^{\frac{N_f}{2}} \det P^{(1)}_{n_1}(\tilde{Q}^2) \simeq \frac{1}{\det P^{(1)}_{n_1}(\tilde{Q}^2)} = \int \mathcal{D}[\eta^\dagger]\mathcal{D}[\eta] e^{\eta^\dagger P^2_{n_2}(\tilde{Q}^2)\eta}$$

- and this second polynomial $P^{(2)}_{n_2}$ fulfills

$$\lim_{n_2 \to \infty} P^{(2)}_{n_2}(x) = x^{-\frac{N_f}{2}} P^{(1)}_{n_1}(x)^{-1} \quad \forall x \in [\epsilon, \lambda]$$

the noisy correction

- using the correction, first one has to generate a complex gaussian random vector η according to the normalized gaussian distribution

$$d\rho(\eta) = \frac{\mathrm{e}^{-\eta^\dagger P^{(2)}_{n_2}(\tilde{Q}^2)\eta}}{\int \mathcal{D}[\eta]\mathrm{e}^{-\eta^\dagger P^{(2)}_{n_2}(\tilde{Q}^2)\eta}}$$

the noisy correction

- using the correction, first one has to generate a complex gaussian random vector η according to the normalized gaussian distribution

$$d\rho(\eta) = \frac{\mathrm{e}^{-\eta^\dagger P^{(2)}_{n_2}(\tilde{Q}^2)\eta}}{\int \mathcal{D}[\eta]\mathrm{e}^{-\eta^\dagger P^{(2)}_{n_2}(\tilde{Q}^2)\eta}}$$

- and then accept the change of the gauge fiels $[U] \to [U']$ with the probability measure

$$P_{NC} = \min\left(1, e^{-\eta^\dagger\left(P^{(2)}_{n_2}(\tilde{Q}[U']^2) - P^{(2)}_{n_2}(\tilde{Q}[U]^2)\right)\eta}\right)$$

the noisy correction

- using the correction, first one has to generate a complex gaussian random vector η according to the normalized gaussian distribution

$$d\rho(\eta) = \frac{\mathrm{e}^{-\eta^\dagger P^{(2)}_{n_2}(\tilde{Q}^2)\eta}}{\int \mathcal{D}[\eta]\mathrm{e}^{-\eta^\dagger P^{(2)}_{n_2}(\tilde{Q}^2)\eta}}$$

- and then accept the change of the gauge fiels $[U] \to [U']$ with the probability measure

$$P_{NC} = \min\left(1, e^{-\eta^\dagger\left(P^{(2)}_{n_2}(\tilde{Q}[U']^2) - P^{(2)}_{n_2}(\tilde{Q}[U]^2)\right)\eta}\right)$$

- the needed noisy estimator η is easily obtained from a simpel gaussian distributed vector η'

$$d\rho(\eta') = \frac{\mathrm{e}^{-\eta^\dagger\eta'}}{\int \mathcal{D}[\eta']\mathrm{e}^{-\eta'^\dagger\eta'}} \text{ and } \eta = P^{(2)}_{n_2}(\tilde{Q}^\dagger)^{-\frac{1}{2}}\eta'$$

the update

- gauge field:

  metropolis sweeps

  overrelaxation

the update

- gauge field:

  metropolis sweeps

  overrelaxation

- multibosonic updating:

  heatbath

  overrelaxation

the update

- preconditioning
- determinant breakup
- gauge field:

  metropolis sweeps

  overrelaxation
- multibosonic updating:

  heatbath

  overrelaxation
- reweighting

the metropolis algorithm

- the probability for going from one configuration $[\phi]$ to $[\phi\prime]$ is given by

$$P([\phi\prime] \leftarrow [\phi]) \propto F\left(\frac{e^{-S[\phi\prime]}}{e^{-S[\phi]}}\right) \tag{1}$$

the metropolis algorithm

- the probability for going from one configuration $[\phi]$ to $[\phi\prime]$ is given by

$$P([\phi\prime] \leftarrow [\phi]) \propto F\left(\frac{e^{-S[\phi\prime]}}{e^{-S[\phi]}}\right) \tag{1}$$

- with any function, that maps $[0, \infty]$ to $[0, 1]$ and fullfills

$$\frac{F(x)}{F\left(\frac{1}{x}\right)} = x$$

the metropolis algorithm

- the probability for going from one configuration $[\phi]$ to $[\phi\prime]$ is given by

$$P([\phi\prime] \leftarrow [\phi]) \propto F\left(\frac{e^{-S[\phi\prime]}}{e^{-S[\phi]}}\right) \tag{1}$$

- with any function, that maps $[0, \infty]$ to $[0, 1]$ and fullfills

$$\frac{F(x)}{F\left(\frac{1}{x}\right)} = x$$

- usually for F one chooses

$$F(x) = \min(1, x)$$

the metropolis algorithm

- so, first a randomly chosen configuration is generated

the metropolis algorithm

- so, first a randomly chosen configuration is generated
- $\longrightarrow$ every configuration with a lower action (higher Boltzman factor) is accepted, while otherwise just with

$$e^{-(S[\phi']-S[\phi])}$$

  or even rejected

generalization of the metropolis algorithm

- a generalization to (1) is to generate P by a proposed change and an accept reject step $P = P_A P_C$ with

$$P_C\left([\phi\prime] \leftarrow [\phi]\right)$$

is an arbitrary probability distribution for the proposed change of the configuration $[\phi] \rightarrow [\phi']$ and

$$P_A\left([\phi\prime] \leftarrow [\phi]\right)$$

the acceptance probability is defined in such way, that it compensates for P_C, namely

$$P_A\left([\phi\prime]\right) \propto \min\left\{1, \frac{P_C([\phi] \leftarrow [\phi'])W_c[\phi']}{P_C([\phi] \leftarrow [\phi'])W_c[\phi]}\right\}$$

heatbath algorithm and overrelaxation

- the heatbath algorithm

  in every step just a part of the field variables, e.g. the gauge link at one particular lattice site, ist changed by combining many such steps, ergodicity can be archieved

heatbath algorithm and overrelaxation

- the heatbath algorithm

  in every step just a part of the field variables, e.g. the gauge link at one particular lattice site, ist changed by combining many such steps, ergodicity can be archieved

- the overrelaxation algorithm

  the configurations are changed in a way, which leaves the action invariant and ensures

$$[\phi] \overset{\text{update}}{\longrightarrow} [\phi'] \overset{\text{update}}{\longrightarrow} [\phi]$$

in each single update step, always with $P_A = 1$

heatbath algorithm and overrelaxation

- the heatbath algorithm

  in every step just a part of the field variables, e.g. the gauge link at one particular lattice site, ist changed by combining many such steps, ergodicity can be archieved

- the overrelaxation algorithm

  the configurations are changed in a way, which leaves the action invariant and ensures

  $$[\phi] \overset{\text{update}}{\longrightarrow} [\phi'] \overset{\text{update}}{\longrightarrow} [\phi]$$

  in each single update step, always with $P_A = 1$

- $\rightarrow$ the action is left unchanged, so this algorithm is not ergodic.

speed up the code: preconditioning

- preconditioning decreasing the condition number $\frac{\lambda}{\epsilon}$ by even-odd preconditioning

speed up the code: preconditioning

- preconditioning decreasing the condition number $\frac{\lambda}{\epsilon}$ by even-odd preconditioning
- decompose the fermion matrix $\tilde{Q}$ in subspaces, containing the odd, respectively the even points of the lattice

$$\tilde{Q} = \gamma_5 Q = \begin{pmatrix} \gamma_5 & -\kappa\gamma_5 M_{oe} \\ -\kappa\gamma_5 M_{eo} & \gamma_5 \end{pmatrix}$$

speed up the code: preconditioning

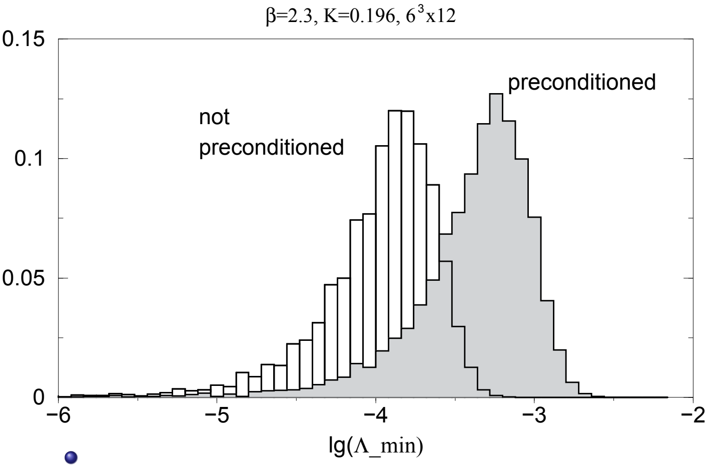
- preconditioning decreasing the condition number $\frac{\lambda}{\epsilon}$ by even-odd preconditioning
- decompose the fermion matrix $\tilde{Q}$ in subspaces, containing the odd, respectively the even points of the lattice

$$\tilde{Q} = \gamma_5 Q = \begin{pmatrix} \gamma_5 & -\kappa\gamma_5 M_{oe} \\ -\kappa\gamma_5 M_{eo} & \gamma_5 \end{pmatrix}$$

- for the fermion determinant we have

$$\det \tilde{Q} = \det \hat{Q}, \text{ with } \hat{Q} \equiv \gamma_5 - K^2\gamma_5 M_{oe} M_{eo}$$

two-step multi-bosonic scheme



- distribution of the smallest eigenvalues of the squared preconditioned fermion matrix $\tilde{Q}^2$ versus the non-preconditioned one

speed up the code: determinant breakup

- use the factorization of the fermionic determinant in several factors, also allowing for some "fractional" number of flavours

$$|\,\mathrm{det}\,(\tilde{Q})|^{N_f} = \left[|\,\mathrm{det}\,(\tilde{Q}^2)|^{\frac{N_f}{2n_B}}\right]^{n_B}$$

speed up the code: determinant breakup

- use the factorization of the fermionic determinant in several factors, also allowing for some "fractional" number of flavours

$$|\det(\tilde{Q})|^{N_f} = \left[|\det(\tilde{Q}^2)|^{\frac{N_f}{2n_B}} \right]^{n_B}$$

- measurement correction: reweighting

$$\lim_{n_4 \to \infty} P^{(1)}_{n_1}(x) P^{(2)}_{n_2}(x) P^{(4)}_{n_4}(x) \text{ with } P^{(4)}_{n_4}(x) = \frac{1}{\sqrt{P^{(2)}_{n_2}}}$$

speed up the code: determinant breakup

- use the factorization of the fermionic determinant in several factors, also allowing for some "fractional" number of flavours

$$|\det(\tilde{Q})|^{N_f} = \left[|\det(\tilde{Q}^2)|^{\frac{N_f}{2n_B}} \right]^{n_B}$$

- measurement correction: reweighting

$$\lim_{n_4 \to \infty} P^{(1)}_{n_1}(x) P^{(2)}_{n_2}(x) P^{(4)}_{n_4}(x) \text{ with } P^{(4)}_{n_4}(x) = \frac{1}{\sqrt{P^{(2)}_{n_2}}}$$

- after reweighting, the expectation value of a quantity A is given by

$$\langle A \rangle = \frac{\left\langle A \exp\left\{ \eta^\dagger \left[1 - P^{(4)}_{n_4}(Q^\dagger Q) \right] \eta \right\} \right\rangle_{U,\eta}}{\left\langle \exp\left\{ \eta^\dagger \left[1 - P^{(4)}_{n_4}(Q^\dagger Q) \right] \eta \right\} \right\rangle_{U,\eta}}$$

the polynomials

- $$P_{n_1}^{(1)} \simeq \frac{1}{x^{\alpha}}$$

the polynomials

- $$P^{(1)}_{n_1} \simeq \frac{1}{x^\alpha}$$
- $$P^{(2)}_{n_2} \simeq \frac{1}{P^{(1)}_{n_1}(x)x^\alpha}$$

the polynomials

- $$P^{(1)}_{n_1} \simeq \frac{1}{x^\alpha}$$
- $$P^{(2)}_{n_2} \simeq \frac{1}{P^{(1)}_{n_1}(x)x^\alpha}$$
- $$P^{(4)}_{n_4}(x) = \frac{1}{\sqrt{P^{(2)}_{n_2}(x)}}$$

hybrid monte carlo

- basic idea: move the configuration through configuration space $\rightarrow$ in each step all field variables are updated by computing their trajectory through a coupled set of equations of motions

hybrid monte carlo

- basic idea: move the configuration through configuration space $\rightarrow$ in each step all field variables are updated by computing their trajectory through a coupled set of equations of motions
- why not simply random walks? $\rightarrow$ a molecular dynamics trajectory is assumed to move more rapidly away from the original configuration

hybrid monte carlo

- basic idea: move the configuration through configuration space $\rightarrow$ in each step all field variables are updated by computing their trajectory through a coupled set of equations of motions
- why not simply random walks? $\rightarrow$ a molecular dynamics trajectory is assumed to move more rapidly away from the original configuration
- for the derivation of the equations of motions, we need to look at

$$H\left[P, U, \phi\right] \equiv \frac{1}{2}\sum_{x\mu j} P^2 + S_g[U] + \sum_{xy} \phi(x)\tilde{Q}^2_{x,y}\phi^*(y)$$

hybrid monte carlo

- $$H[P,U,\phi] \equiv \frac{1}{2}\sum_{x\mu j} P^2 + S_g[U] + \sum_{xy} \phi(x)\tilde{Q}^2_{x,y}\phi^*(y)$$

  during a (lepfrog...) trajectory the pseudofermion field ϕ is constant and generated from a simple gaussian

- $$d\eta d\eta^\dagger e^{-(\eta\eta^\dagger)}$$

  $$\phi = \eta\tilde{Q} \rightarrow \eta = \phi\tilde{Q}^{-1} \rightarrow d\phi d\phi^\dagger e^{-\phi\tilde{Q}^{-2}\phi^\dagger}$$

  at the beginning of the trajectory the conjugate momenta P are generated according to the distribution

- $$dPe^{-\frac{1}{2}P^2}$$

the polynomials

- with the corresponding canonical differential equations

$$\begin{aligned}\dot{P}(x,\mu,j) &= -D_{x,\mu,j}S[U] \\ \dot{U}(x,\mu) &= iP(x,\mu)U(x,\mu)\end{aligned}$$

this derivative is defined as

$$D_{x\mu j}f(U(x,\mu)) = \left.\frac{d}{d\alpha}\right|_{\alpha=0} f\left(e^{i\alpha T_j}U(x,\mu)\right)$$

the leapfrog trajectory

- a step of length $\Delta\tau$ in P

$$P'_{x\mu j} = P_{x\mu j} - D_{x\mu j}\Delta\tau S[U]$$

- a step in $U_{x\mu}$

$$U'_{x\mu} = Ux\mu e^{\sum_{j=1}^{3} \Delta\tau i T_j P_{x\mu j}}$$

the leapfrog trajectory

- a step of length $\Delta\tau$ in P

$$P'_{x\mu j} = P_{x\mu j} - D_{x\mu j}\Delta\tau S[U]$$

- a step in $U_{x\mu}$

$$U'_{x\mu} = Ux\mu e^{\sum_{j=1}^{3}\Delta\tau i T_j P_{x\mu j}}$$

- the trajectory is a succesive approximation of

$$T(\Delta\tau) = T_P\left(\frac{\Delta\tau}{2}\right) T_U\left(\Delta\tau\right) T_P\left(\frac{\Delta\tau}{2}\right)$$

multiple time scales

- in case of the hamiltonian, we have

$$H[P,U] = \frac{1}{2}P^2 + \sum_{i=1}^{k} S_i[U] \quad (k \geq 1)$$

for a trajectory with length τ, we define decreasing time steps

$$\Delta\tau_i = \frac{\Delta\tau_{i+1}}{N_i} = \frac{\tau}{N_k N_{k-1} \cdots N_i}$$

with N_i = step number, $(0 \leq i \leq k)$, $(\Delta\tau_{k+1} \equiv \tau)$

multiple time scales

- we can change the gauge field as before by

$$T_U(\Delta\tau): \qquad U'_{x\mu} = U_{x\mu} e^{i\Delta\tau \sum_{j=1}^{3} T_j P_{x\mu j}}$$

- and define a step in P by

$$T_{S_i}(\Delta\tau): \qquad P'_{x\mu j} = Px\mu j - D_{x\mu j} \Delta\tau S_i[U]$$

sexton-weingarten higher order integrator

- let us define

$$T_0(\Delta\tau_0) = T_{S_0}\left(\frac{\Delta\tau_0}{2}\right) T_U(\Delta\tau_0) T_{S_0}\left(\frac{\Delta\tau_0}{2}\right)$$

sexton-weingarten higher order integrator

- let us define

$$T_0(\Delta\tau_0) = T_{S_0}\left(\frac{\Delta\tau_0}{2}\right) T_U(\Delta\tau_0) T_{S_0}\left(\frac{\Delta\tau_0}{2}\right)$$

- and for $i = 1, 2, \dots, k$

$$T_i(\Delta\tau_i) = T_{S_i}\left(\frac{\Delta\tau_i}{2}\right) \{T_{i-1}(\Delta\tau_{i-1})\}^{N_{i-1}} T_{S_i}\left(\frac{\Delta\tau_i}{2}\right)$$

- sexton-weingarten

$$T_0(\Delta\tau_0) =$$

$$T_{S_0}\left(\frac{\Delta\tau_0}{6}\right) T_U\left(\frac{\Delta\tau_0}{2}\right) T_{S_0}\left(\frac{2\Delta\tau_0}{3}\right) T_U\left(\frac{\Delta\tau_0}{2}\right) T_{S_0}\left(\frac{\Delta\tau_0}{6}\right)$$

polynomial hybrid monte carlo

- in polynomial hmc one approximates the Q-matrices with polynomials

$$(\bar{\lambda}\tilde{Q}^{-2}\lambda) \simeq (\lambda P(\tilde{Q}^2)\lambda)$$

- instead of $N_f = 2$ one can although work with determinant breakup

$$(\det \tilde{Q}^2)^{\frac{N_f}{2}} = \left[\left(\det \tilde{Q}\right)^{\alpha}\right]^{N_b} \rightarrow \sum_{n_b=1}^{N_b} (\bar{\lambda}_{n_b} P(\tilde{Q})^2 \lambda)$$

- unsing the product rule in the derivative

$$D_{x\mu j}\tilde{Q}^2 = \tilde{Q}(D_{x\mu j}\tilde{Q}) + (D_{x\mu j}\tilde{Q})\tilde{Q}$$

- Montvay, Münster, *Quantum Fields on a Lattice*
- I.Campos et.al. , *Monte Carlo simulation of $SU(2)$ Yang-Mills theory with light gluinos*, [arXiv:hep-lat/9903014]
- E.E. Scholz, *Light Quark Fields in QCD: Numerical Simulations and Chiral Perturbation Theory*, PhD, 2005
- R. Peetz, *Spectrum of $N = 1$ Super Yang Mills Theory on the Lattice with a light gluino*, PhD, 2003
- C. Gebert, *Light quark fields in lattice gauge theories*, PhD, 2002
- B. Gehrmann, *The step scaling function of QCD at negative flavor number*, PhD, 2002
- I.Montvay, E.E. Scholz , *Updating algorithms with multi-step stochastic correction*, [arXiv:hep-lat/0506006]

M. Lüscher , *A new approach to the problem of dynamical quarks in numerical simulations of lattice QCD*, Nuc.Phys. B418, S.637-648, 1994