Automated Lattice Perturbation Theory in the Schrödinger Functional

Dirk Hesse

with Georg von Hippel Rainer Sommer

John von Neumann Institute for Computing (NIC),
DESY Zeuthen

WWU Münster 2010

Outline

Why Doing Lattice Perturbation Theory, why SF?

Lattice perturbation theory is needed/used for:

- Matching with perturbative results.
- Obtaining improvement coefficients.
- Renormalization.

The SF is used ...

- for *scale dependent* problems, e.g. the running coupling, or
- renormalization,
- for obtaining improvement coefficients (e.g. c_{SW}).

Automatic for the People!

Actions in LQCD

- The choice of action is not unique.
- Each possible action has advantages & disadvantages.
- There is no "best choice"!

What does it mean for LPT?

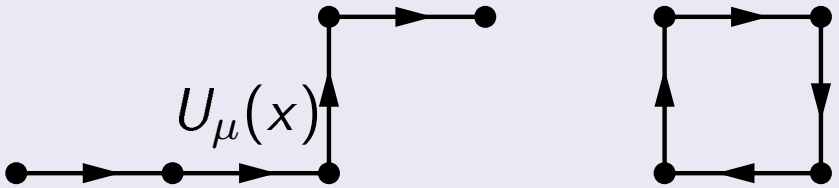
- Lots of different actions are in use.
- Each action leads to different Feynman rules!
- Calculations have to be repeated.
- Automation is suggested.

Outline

Basic Building Blocks

Curves

- Curve $\mathcal{C} = (s_1\mu_1, \ldots, s_l\mu_l)$, $\mu_i \in \{x, y, z, t\}$, $s_i = \pm 1$
- Path $U_{\mathcal{C}}$ = "Product of links along $\mathcal{C}$"



Actions

$$S_G = \sum_{\mathcal{C}_i(x,x)\in\Lambda} w_i \mathrm{ReTr}(1 - U_{\mathcal{C}_i(x,x)})$$

$$S_F = \sum_{\mathcal{C}_i(x,y)\in\Lambda} w_i \overline{\psi}(x)\Gamma U_{\mathcal{C}_i}(x,y)\psi(y)$$

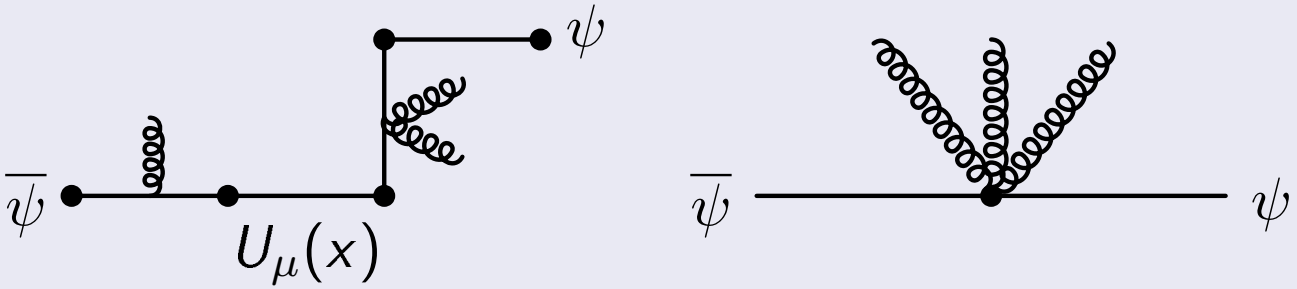
Perturbation Theory.

Expanding the action in powers of g_0,

$$S = \sum_r g_0^r S^{(r)},$$

determines the Feynman rules, i.e. propagators & vertices.

In LPT, vertices occur that have no continuum counterpart!



Paths.

A path $U_{\mathcal{C}}$ reads

$$U_{\mathcal{C}(x,y)} = U^{s_1}_{\mu_1}(\tilde{x}_1)U^{s_2}_{\mu_2}(\tilde{x}_2)\cdots U^{s_l}_{\mu_l}(\tilde{x}_l),$$

with

$$x_i + s_i\hat{\mu}_i = x_{i+1}, \quad x_0 := x, y = x_{l+1} \quad \tilde{x}_i = (x_i + x_{i+1})/2,$$

such that U lives on the mid-point between two lattice sites.

Expanding Paths

Expanding U_C in powers of g_0 yields

$$U_{C(x,y)} = \prod_{i=1}^{I} \sum_{r_i} s_i^{r_i} \frac{g_0^{r_i}}{r_i!} A_{\mu_i}^{r_i}(\tilde{x}_i),$$

or

$$U_{C(x,y)} = \prod_{i=1}^{I} \sum_{r_i} \frac{g_0^{r_i}}{r_i!} (I^a X_i^a)^{r_i},$$

with

$$X_i = s_i A_{\mu_i}(\tilde{x}_i).$$

Expanding Paths (2)

Picking *one* power r of g_0 ...

$$U^{(r)}_{C(x,y)} = \frac{g_0^r}{\alpha_1! \cdots \alpha_l!} I^{a_1} \cdots I^{a_r} X^{a_1}_{i_1} \cdots X^{a_r}_{i_l},$$

where $\alpha_j = \sum_k \delta(j, i_k)$ counts how many of the X belong to the jth link.

We have a *path independent* color factor

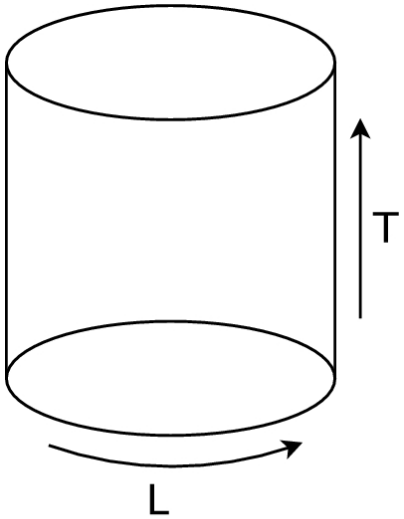
$$C^{a_1,\ldots,a_r} = I^{a_1} \cdots I^{a_r}$$

and a *path dependent* kinetic factor

$$K(C, x) = X^{a_1}_{i_1} \cdots X^{a_r}_{i_l}.$$

Outline

Meet the Schrödinger Functional.



- *M. Lüscher, R. Narayanan, P. Weisz et al.*,
'The Schrödinger functional: A Renormalizable probe for nonAbelian gauge theories',
Nucl. Phys. **B384**, 168-228 (1992).
`hep-lat/9207009`
- *S. Sint*,
'The Schrödinger functional in QCD',
Nucl. Phys. Proc. Suppl. **42**, 835-837 (1995).
`hep-lat/9411063`

- QCD on a $L^3 \times T$ cylinder.
- *Fixed temporal* boundary conditions at $t = 0, t = T$.
- *Periodic spatial* boundary conditions.

The Schrödinger Functional Boundary.

Fermions.

- Temporal: Dirichlet.
- Spatial: Periodic with phase angle θ,

$$\psi(x + L\hat{k}) = e^{i\theta_k}\psi(x).$$

Gluons.

- $U_k(x)|_{x_0=0} = W_k(\mathbf{x}) \quad U_k(x)|_{x_0=T} = W'_k(\mathbf{x})$,
- Commonly $W_k = e^C$ with C diagonal (W' analog).
- Spatial: Periodic.

The Background Field.

Gluon Boundary Conditions Induce a *Background Field*.

$$V_\mu(x) = (V[C, C'])_\mu(x)$$

- V_μ minimizes the action for given C, C'.
- Therefore, V_μ is the new expansion point.
- Decompose $U_\mu(x) = e^{g_0 q_\mu(x)} V_\mu(x_0)$.
- $q_\mu(x)$ are the d.o.f.

Expansion with a Background Field.

Expanding paths with background field will yield terms like

$$U^{(r)}_{\mathcal{C}(x,y)} = \underbrace{I^{a_1} I^{a_2} V_{\mu_1}(x_1) I^{a_3} \cdots I^{a_r} V_{\mu_l}(x_l)}_{\text{color factor C}} \times \underbrace{q^{a_1}_{\mu_1}(\tilde{x}_1) q^{a_2}_{\mu_1}(\tilde{x}_1) \cdots q^{a_r}_{\mu_l}(\tilde{x}_l)}_{\text{kinetic}} + \ldots$$

The color factor becomes path dependent as well!

Outline

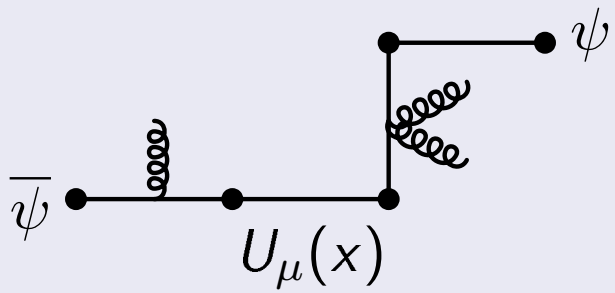
How to Automate?

Lüscher and Weisz: Expansion 'by induction'.

- Start with a single link $U_{\mathcal{C}} = U_\mu(x)$
- Find a multiplication algorithm

$$U^{(r)}_{\mathcal{C}}, U^{(s)}_{\mathcal{C}'} \mapsto U^{(r)}_{\mathcal{C}} \cdot U^{(s)}_{\mathcal{C}'} \in U^{(r+s)}_{\mathcal{C},\mathcal{C}'}$$

- Build up the desired path by multiplying links.



M. Lüscher, P. Weisz,
Nucl. Phys. **B266**, 309 (1986).

Expansion of a Link.

A single link reads

$$U_\mu(x) = \left(\sum_r \frac{g_0}{r!} I^a q_\mu^a(x) \right) V_\mu(x),$$

with a simple color factor in each order

$$C^{a_1,\ldots,a_r}(\mathcal{C}) = I^{a_1} \cdots I^{a_r} V_\mu(x).$$

What about the multiplication?

The Background Field

Useful Properties of *Abelian* Background Fields

- $V_\mu(x)^\dagger = V_\mu(x)^{-1}, \quad V_0(x) = 1, \quad V_k(x) = V(x_0)$
- $V(x_0) I^a = I^a V(x_0) e^{i\phi_a(x_0)}$
- $V(x_0 + \Delta t) = V(x_0) e^{i\mathcal{E}\Delta t}$
- $e^{i\mathcal{E}} I^a = I^a e^{i\mathcal{E}} e^{i\phi'_a}$

Color Factors Can be Cast Into a *Standard Form*.

$$C^{a_1,\ldots a_r}(x_0, A, B, \mathbf{C}, \mathbf{D}) = I^{a_1} \ldots I^{a_r} V(x_0)^A e^{i\mathcal{E}B} e^{i/2(\phi'\cdot\mathbf{C} + \phi(x_0)\cdot\mathbf{D})}$$

S. Takeda, U. Wolff,
PoS **LAT2007**, 257 (2007).

Color Multiplication.

The color factor decomposes into a path independent and path dependent part.

$$C^{a_1,\dots a_r}(x_0, A, B, \mathbf{C}, \mathbf{D}) = I^{a_1} \cdots I^{a_r} C(x_0, A, B, \mathbf{C}, \mathbf{D}).$$

One can formulate a multiplication for the path dependent bit.

$$C, C' \mapsto C'' = C \cdot C'$$
$$(x_0, A, B, \mathbf{C}, \mathbf{D}), (x_0', A', B', \mathbf{C}', \mathbf{D}') \mapsto (x_0, A'', B'', \mathbf{C}'', \mathbf{D}'').$$

S. Takeda, U. Wolff,
PoS **LAT2007**, 257 (2007).

Kinetic Multiplication.

Formulating a multiplication for the kinetic part is much simpler!
Let's remember: The kinetic part consists of lists

- of the directions $(\mu_1, \ldots, \mu_l)$,
- of the signs $(s_1, \ldots, s_l)$,
- and the sites $(\tilde{x}_1, \ldots \tilde{x}_l)$.

Multiplication of these boils down to concatenating the lists!

$$(\mu_1, \ldots, \mu_l), (\mu'_1, \ldots, \mu'_s) \mapsto (\mu_1, \ldots, \mu_l, \mu'_1, \ldots, \mu'_s)$$

etc.

Outline

Some History.

What has been done so far?

- *Lüscher & Weisz* proposed the algorithm.

  M. Lüscher, P. Weisz, Nucl. Phys. **B266**, 309 (1986).

- *Hart et al.* implemented a recursive version in `python`.

  A. Hart, G. M. von Hippel, R. R. Horgan et al., Comput. Phys. Commun. **180**, 2698-2716 (2009).

- *Takeda & Wolff* started extending Hart's code to the SF.

  S. Takeda, U. Wolff, PoS **LAT2007**, 257 (2007).

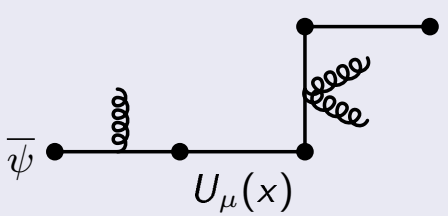
The issues.

- Code is OK for gluonic and simple fermionic Paths.
- It is not OK for general fermionic paths.
- Interplay between `python` and `FORTRAN` code.

Expansion Algorithm

Wanted: Expansion $U_{\mathcal{C}}^{(r)}$ (length$(\mathcal{C}) = l$, Λ: lattice, $\mathcal{B}$: boundary)

for $i = 1$ to l **do**
  if $\tilde{x}_i \notin \Lambda$ **then**
    return
  else
    expand links $X_i^{(j)}$, $j = 0, \ldots, r$
  end if
end for
for all $(n_1, \ldots, n_l)$, $\sum_k n_k = r$, $n_i = 0$ if $\tilde{x}_i \in \mathcal{B}$ **do**
  $U_{\mathcal{C}}^{(r)} \to U_{\mathcal{C}}^{(r)} \cup \left\{ X_1^{(n_1)} X_2^{(n_2)} \ldots X_l^{(n_l)} \right\}$
end for



$$\to (1,0,2,0) \to X_1^{(1)} V_{\mu_2}(\tilde{x}_2) X_3^{(2)} V_{\mu_4}(\tilde{x}_4)$$

Outline

Some Remarks.

Lattice Perturbation Theory in the SF.

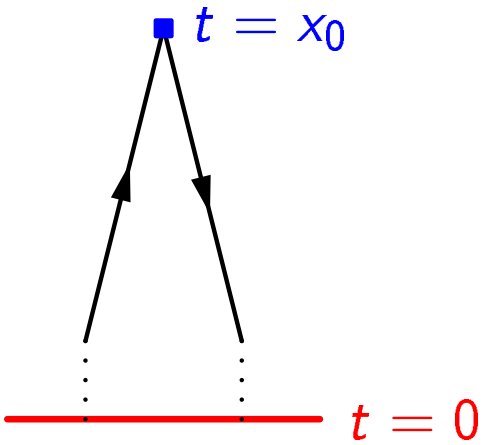
- Usually coordinate or time-momentum space.
- No integral over Brillouin zone, but sum (finite volume).

How to use the Code.

- Propagators are calculated by expanding S and inverting $(CG)^a$.
- Quark-anti quark-n-gluon vertices and n-gluon vertices also calculated from the actions.
- Diagrams are coded explicitly, like one would do it on paper.

[a] The gluon propagator needs to be *gauge-fixed* before inversion.

Correlation Functions.



$$f_A(x_0) = -\sum_{\mathbf{y},\mathbf{z}} \frac{1}{3} \langle A_0^a(x) \zeta(\mathbf{y}) \gamma_5 \tau^a / 2 \zeta(\mathbf{z}) \rangle$$

$$f_P(x_0) = -\sum_{\mathbf{y},\mathbf{z}} \frac{1}{3} \langle P^a(x) \zeta(\mathbf{y}) \gamma_5 \tau^a / 2 \zeta(\mathbf{z}) \rangle$$

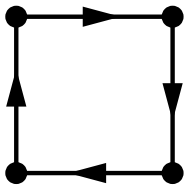
M. Lüscher, S. Sint, R. Sommer et al.,
Nucl. Phys. **B478**, 365-400 (1996).`hep-lat/9605038`

$$f_X^{(0)}(x_0) = -\frac{1}{2} \sum_{\mathbf{y},\mathbf{z}} \operatorname{tr} \{ P_+ \gamma_5 S_0(y,x) \gamma_5 \Gamma_X S_0(x,z) \} \Big|_{y_0 = z_0 = a}$$

$$\Gamma_A = -\gamma_0,\ \Gamma_P = 1,\ P_+ = (1+\gamma_0)/2$$

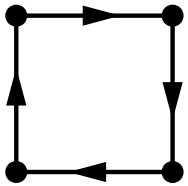
Some Action(s)!

Wilson

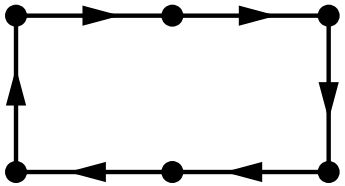


Iwasaki

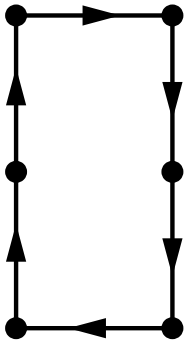
c_0



$+ c_1$

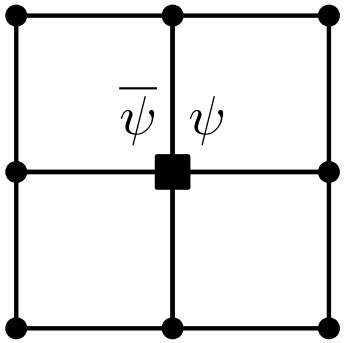


$+ c_1$



Clover term

$\sigma_{\mu\nu}$



Outline

Expanding made Easy!

The Wilson Gauge Action

```
Curve C;
C += mu, nu, -mu, -nu; // mu, nu are keywords!
Action Wilson;
Wilson.PushBack(C);    // actions can contain
                       // many paths
```

- Any vertex of order r: `Wilson.Expand(r)`!
- Summation over $\mu \neq \nu$ implied!

Really Easy!

The Iwasaki Gauge Action

```
Curve P(c0), R1(c1), R2(c1);
P += mu, nu, -mu, -nu;                    // plaquette
R1 += mu, mu, -nu,-mu, -mu, -nu;          // 2x1 rect.
R2 += mu, nu, nu, -mu, -nu, -nu;          // 1x2 rect.
Action Iwasaki;
Iwasaki.PushBack(P);
Iwasaki.PushBack(R1);
Iwasaki.PushBack(R2);
```

Templates so far:

- Gauge: *Wilson, Iwasaki*
- Fermion: *Wilson, Clover – (**Dirac**, Pauli, Scalar)*

Tree Level.

f_A at Tree Level.

```
// get the propagator
FermionPropagator P(WilsonFermion(0));
Storage S = P.Get(0,FermionMomentum());
// get fA
fA = -0.5*(Pp * g5 * S(1, x0) *
           g5 * g0 * S(x0, 1)  ).tr();
// output
cout << "fA(param().T()/2) = "
     << fA.real() << endl;
```

Output

fA(8) = -0.5113075898406796 (-0.5113075898406484)

Outline

Z_P to one loop.

First Test: Calculate the one loop contribution to

$$Z_P(g_0, L/a) = c\frac{\sqrt{f_1}}{f_P(T/2)}.$$

S. Capitani, M. Guagnelli, M. Luescher et al.,
Nucl. Phys. Proc. Suppl. **63**, 153-158 (1998). `hep-lat/9709125`

Expectation:

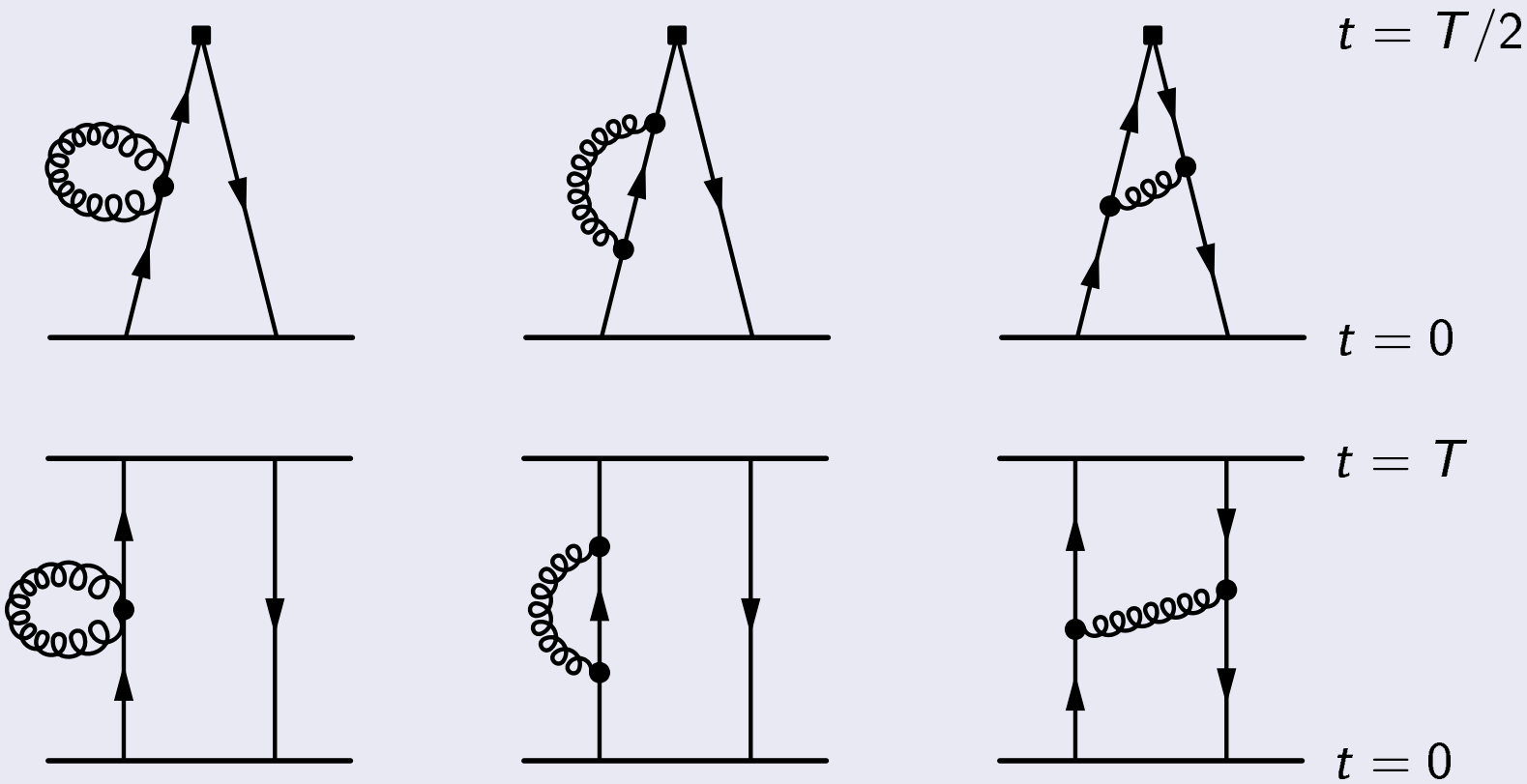
$$Z_P(g_0, L/a) = 1 + \sum_{r=0}^{\infty} g_0^{2k} Z_P^{(r)}(L/a),$$

where $Z_P^{(r)}(L/a)$ is a polynomial of degree r in $ln(L/a)$. We are interested in the finite part of $Z_P^{(1)}$,

$$Z_P^{(1)} = C_F z_p(\theta, \rho) - d_0 \ln(L/a) + O(a/L).$$

Diagrams.

Main Contributions to $f_P^{(1)}$, $f_1^{(1)}$.



More Details.

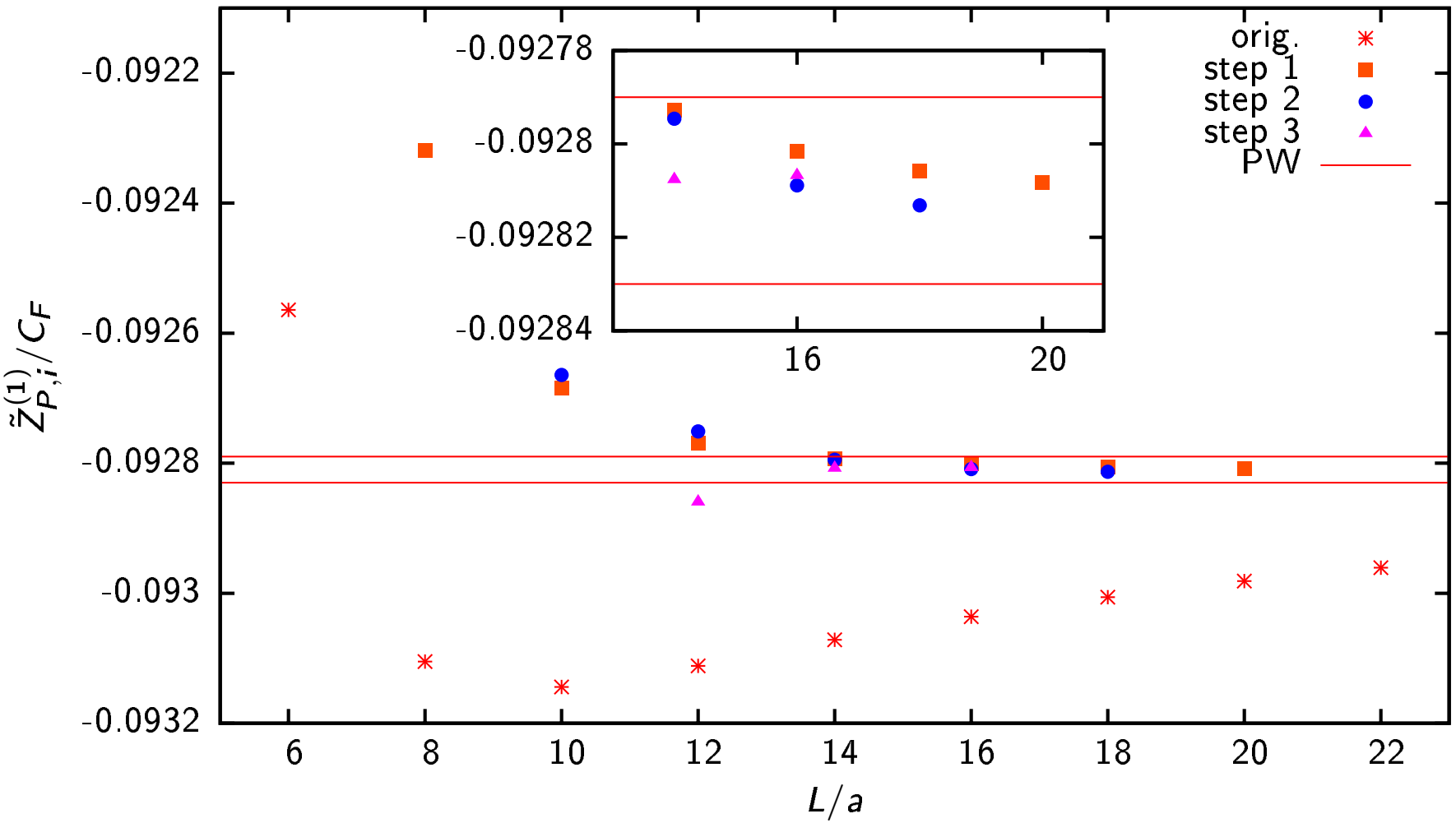
Diagrams were evaluated ...

- for $L = 4, 6, \ldots, 22,\ T = L,\ \theta = 0.5$
- with clover improved Wilson fermions,
- and Wilson gauge action,
- in O(hours).

S. Sint & P. Weisz in `hep-lat/9808013`

$$z_P(\theta = 0.5, T/L = 1) = -0.09281(2)$$

Z_P, a Picture.


-0.0922
-0.0924
-0.0926
-0.0928
-0.093
-0.0932
$\tilde{Z}^{(1)}_{P,i}/C_F$
6
8
10
12
14
16
18
20
22
L/a
-0.09278
-0.0928
-0.09282
-0.09284
16
20
orig.
step 1
step 2
step 3
PW

Outline

The PCAC Mass.

We define an *unrenormalized* quark mass,

$$m(m_0) = \frac{1}{2}\frac{\tilde{\partial} f_A^I(T/2)}{f_P(T/2)}.$$

K. Jansen, C. Liu, M. Luscher et al.,
Phys. Lett. **B372**, 275-282 (1996). `hep-lat/9512009`

The critical *bare* m_c mass is defined by

$$m(m_c) \stackrel{!}{=} 0,$$
$$m_c = m_c^{(0)} + g_0^2 m_c^{(1)} + \dots .$$

$\tilde{\partial} = 1/2(\partial_0^* + \partial_0)$

The Critical mass.

We expand the quark mass,

$$m(m_0) = m^{(0)}(m_0) + g_0^2 m^{(1)}(m_0) + \dots .$$

The tree and one loop parts of m_c can be obtained:

Require

$$m^{(0)}(m_c^{(0)}) + g_0^2 \left\{ m^{(1)}(m_c^{(0)}) + m_c^{(1)} \partial_{m_0} m^{(0)}(m_c^{(0)}) \right\} \stackrel{!}{=} 0.$$

At both orders!

One Loop Part.

The one loop contribution to m reads

$$m^{(1)} = \frac{1}{2}\frac{\tilde{\partial} f_A^{(1)}(T/2)}{f_P^{(0)}(T/2)} - f_P^{(1)}(T/2)\frac{\tilde{\partial} f_A^{(0)}(T/2)}{2[f_P^{(0)}(T/2)]^2} + c_A^{(1)}\frac{\partial_0^* \partial_0 f_P^{(0)}(T/2)}{2 f_P^{(0)}(T/2)}$$

Need to calculate only $\tilde{\partial} f_A^{(1)}$ at one loop!
Same diagrams (in principle) as for f_P!

Some Numbers.

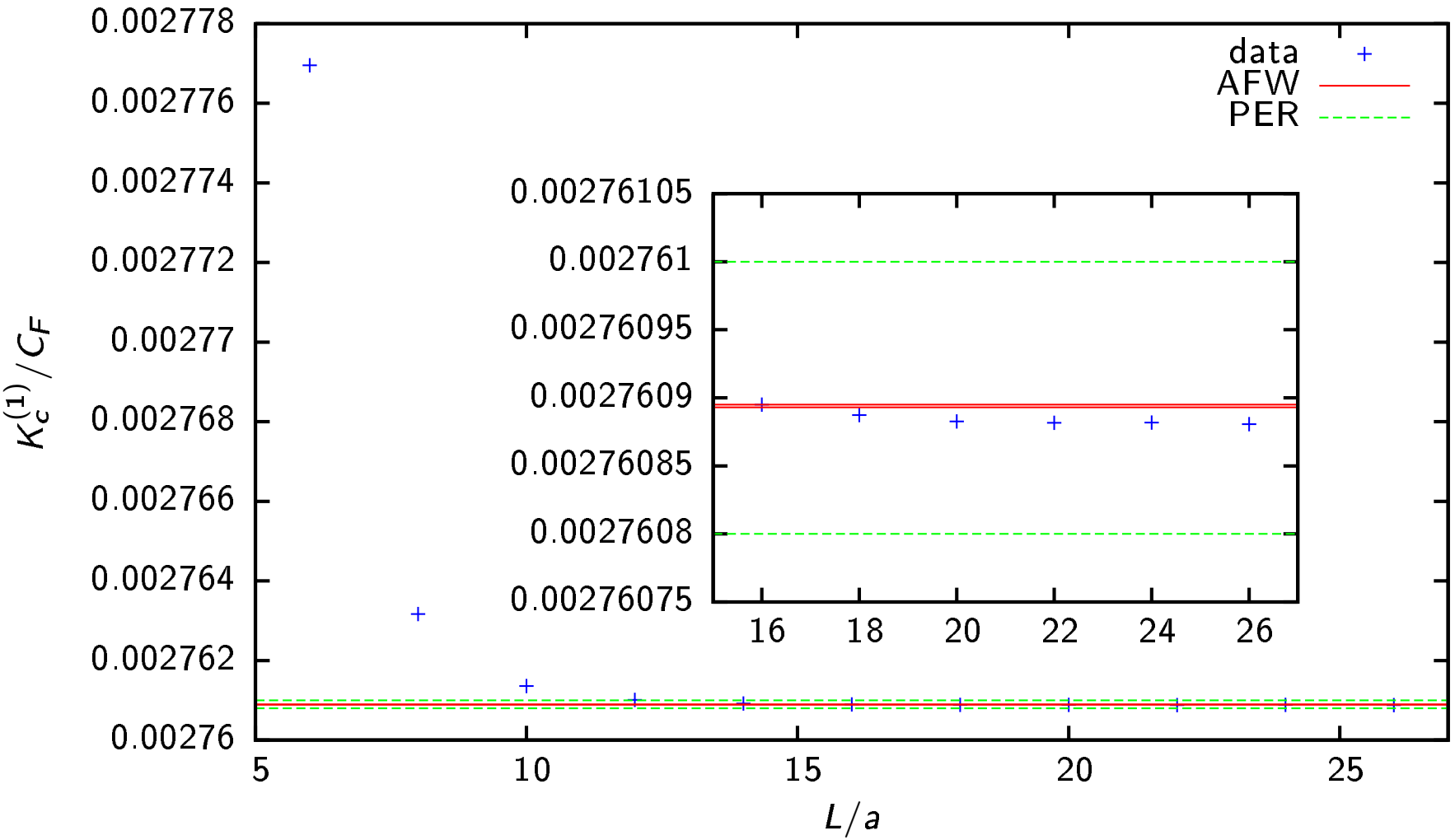
Calculations were performed...

- with Iwasaki gauge action and
- clover improved Wilson fermions ($\theta = 0$).
- For $T = L = 6, 8, \ldots$.

Aoki, Frezzotti, Weisz in `hep-lat/9808007`

$$K_C^{(1)} = 2.760894(1) \times 10^{-3} C_F$$

m_C, a Picture.


0.002778
0.002776
0.002774
0.002772
0.00277
0.002768
0.002766
0.002764
0.002762
0.00276
5
10
15
20
25
L/a
$K_c^{(1)}/C_F$
data
AFW
PER
0.00276105
0.002761
0.00276095
0.0027609
0.00276085
0.0027608
0.00276075
16
18
20
22
24
26

Summary

- Perturbative calculation must be done.
- You should and can automate them, even in the Schrödinger functional!

- Outlook
  - The code is still beta!
  - More applications to come (e.g. HQET)!

Thank you!

For Further Reading


M. Lüscher and P. Weisz,
Efficient Numerical Techniques For Perturbative Lattice Gauge Theory Computations.
Nucl. Phys. B **266**, *309 (1986).*


A. Hart, G. M. von Hippel, R. R. Horgan and E. H. Müller,
Automated generation of lattice QCD Feynman rules.
Comput. Phys. Commun. **180**, *2698 (2009) [arXiv:0904.0375 [hep-lat]].*


S. Takeda and U. Wolff,
Automatic generation of vertices for the Schrödinger functional.
PoS **LAT2007**, *257 (2007) [arXiv:0709.4167 [hep-lat]].*