a "mass preconditioning" for lattice Dirac operators

G.M. de Divitiis[a,b], R. Petronzio[a,b], N. Tantalo[a,c]

[a] INFN sez. "Tor Vergata"
[b] Rome University "Tor Vergata"
[c] Centro Ricerche e Studi "E. Fermi"

17-06-2010

outline

- a numerical problem: heavy quark propagators decay too fast!
  - the free theory case
  - the interacting case
- a numerical trick: heavy quark propagators decay as slow as light quark propagators
  - the free theory case
  - the interacting case
  - different preconditioning
- the same numerical trick: speeding up numerical inversions for light quark

a numerical problem: heavy quark propagators decay too fast!

first principle approaches to flavour physics are of fundamental importance in the search for physics beyond the Standard Model

in order to calculate heavy flavour observables on the lattice we need to solve the linear system

$$(D + M)\,\psi = \eta^{y_0}$$

on the one hand, the numerical inversion is quite fast for heavy quarks

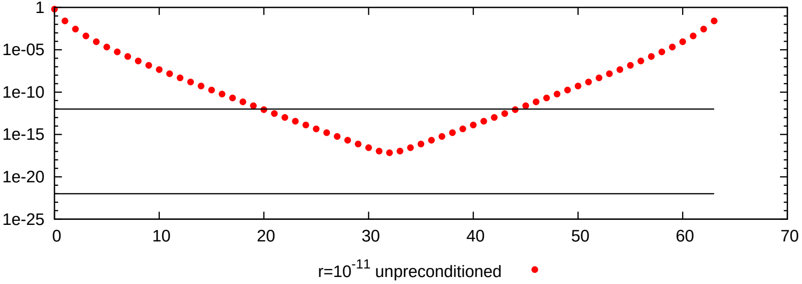
on the other hand, at large times the solution is poorly accurate because $|\psi(x_0;\vec{x})|$ may become much smaller than r for $x_0 \gg y_0$

a solver:

- $n \longrightarrow n+1$
- apply $(D+M)$ a few times ...
- a bit of linear algebra ...
- check if $|(D+M)\,\psi^n - \eta^{y_0}| < r$

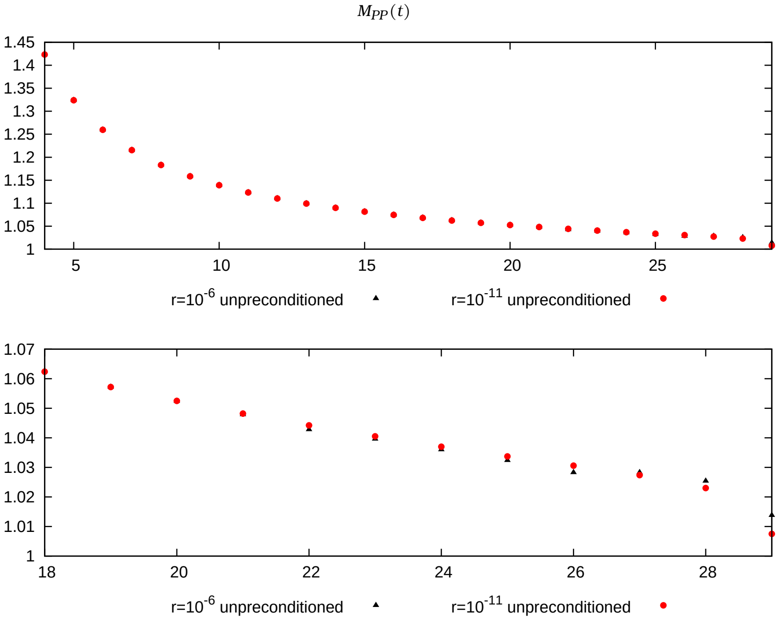
a numerical problem: free theory

$$C(t) = -\mathrm{tr}\{S(0,t)S^\dagger(t,0)\}$$

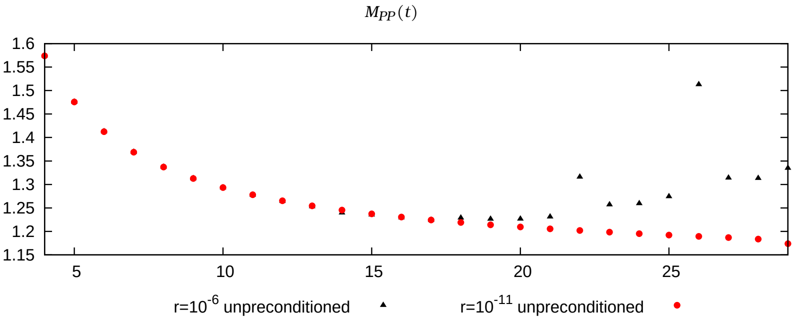


- this is a heavy-heavy "pseudoscalar-pseudoscalar" correlator in free theory
- we do expect that if we choose a "loose" residue something should go wrong at large time distances from the source ...
- in this particular case we can compare the numerical inversion performed with the loose residue $r = 10^{-6}$ with the one performed with the "small" residue $r = 10^{-11}$

a numerical problem: free theory

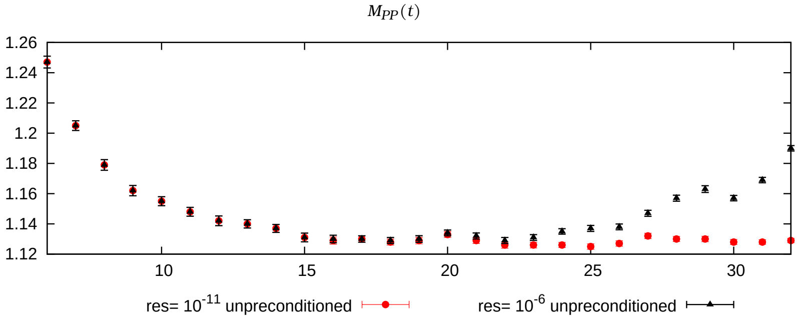


a numerical problem: free theory



- by changing a bit the quark mass the effect may become "particularly" evident
- in this work we have analyzed situations in which the problem can be easily identified and such that "exact" results can be obtained by working with double precision architectures
- we have been working with loose residues $r \sim 10^{-6}$ also because this is the best one can do with single precision architectures (presently GPU are much faster in single precision)

a numerical problem: interacting theory



- here we see the same effect in the interacting theory. . .
- simulation details: $\beta = 5.3$, $k_{sea} = 0.13625$, $am_{sea} \simeq 0.07$, $am_h \simeq 0.35$
- this correspond to the CLS gauge ensamble $E5$ (and we thank our CLS colleagues for sharing the effort of the generation of these gauge configurations)

a numerical trick: heavy quark propagators decay as slow as light quark propagators

in order to solve this numerical problem we propose to precondition the preferred lattice Dirac operator as follows

$$\psi(\vec{x}, x_0) \quad \longrightarrow \quad \alpha(x_0)\, \chi(\vec{x}, x_0)$$

$$= \begin{pmatrix} \alpha(0) & & & & & & & & \\ & \vdots & & & & & & & \\ & & \alpha(0) & & & & & & \\ & & & \alpha(1) & & & & & \\ & & & & \vdots & & & & \\ & & & & & \alpha(1) & & & \\ & & & & & & \vdots & & \\ & & & & & & & \vdots & \\ & & & & & & & & \alpha(L_0 - 1) \end{pmatrix} \begin{pmatrix} \chi(\bullet, 0) \\ \vdots \\ \chi(\bullet, 0) \\ \chi(\bullet, 1) \\ \vdots \\ \chi(\bullet, 1) \\ \vdots \\ \vdots \\ \chi(\bullet, L_0 - 1) \end{pmatrix}$$

this is a matrix that is diagonal in color, Dirac and space indexes and it must be invertible

$$(D + M)\, \psi(\vec{x}, x_0) \quad = \quad \alpha(x_0)\, (D^{prec} + M)\, \chi(\vec{x}, x_0)$$

a numerical trick: heavy quark propagators decay as slow as light quark propagators

one must be careful with boundary conditions

$$\psi(\vec{x}, T) = \psi(\vec{x}, 0) \qquad \longrightarrow \qquad \alpha(T)\,\chi(\vec{x}, T) = \alpha(0)\,\chi(\vec{x}, 0)$$

i.e. the preconditioned field satisfy the following boundary conditions

$$\chi(\vec{x}, T) = \frac{\alpha(0)}{\alpha(T)}\,\chi(\vec{x}, 0)$$

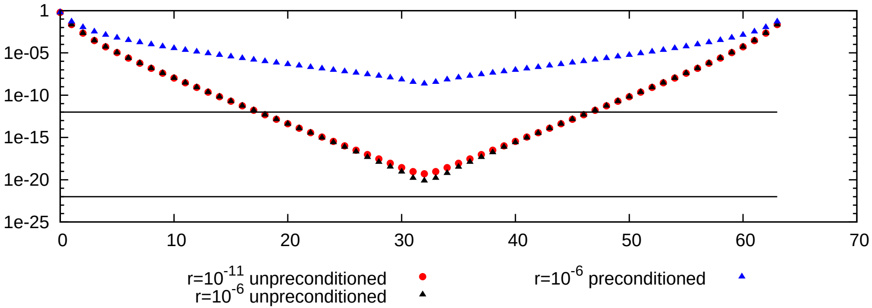
and the preconditioned operator is obtained by modifying the covariant derivatives in the time direction

$$\nabla_0 \psi(x) = U_0(x)\,\psi(x + \hat{0}) - \psi(x) \qquad \longrightarrow \qquad \frac{\alpha(x_0 + 1)}{\alpha(x_0)} U_0(x)\,\chi(x + \hat{0}) - \chi(x)$$

$$\nabla_0^\dagger \psi(x) = \psi(x) - U_0^\dagger(x - \hat{0})\,\psi(x - \hat{0}) \qquad \longrightarrow \qquad \chi(x) - \frac{\alpha(x_0 - 1)}{\alpha(x_0)} U_0^\dagger(x - \hat{0})\,\chi(x - \hat{0})$$

a numerical trick: free theory

$$C(t) = -\mathrm{tr}\{S(0,t)S^\dagger(t,0)\}$$



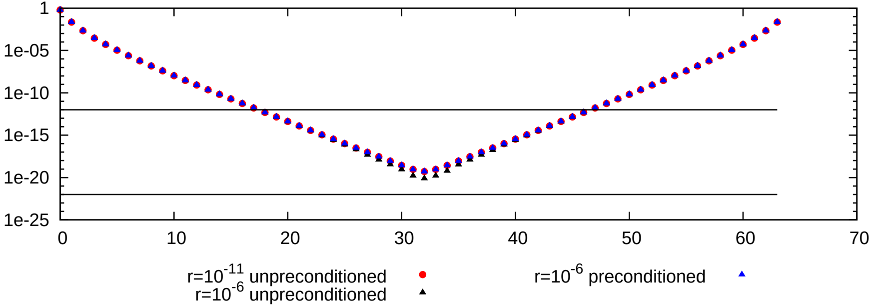
we choose:

$$\psi(\vec{x}, x_0) \longrightarrow \cosh\left[m_0(x_0 - T/2)\right] \chi(\vec{x}, x_0)$$

and, in this particular case, $m_0 = 0.4$. We calculate numerically $\chi(\vec{x}, x_0)$,

a numerical trick: free theory

$$C(t) = -\mathrm{tr}\{S(0,t)S^\dagger(t,0)\}$$

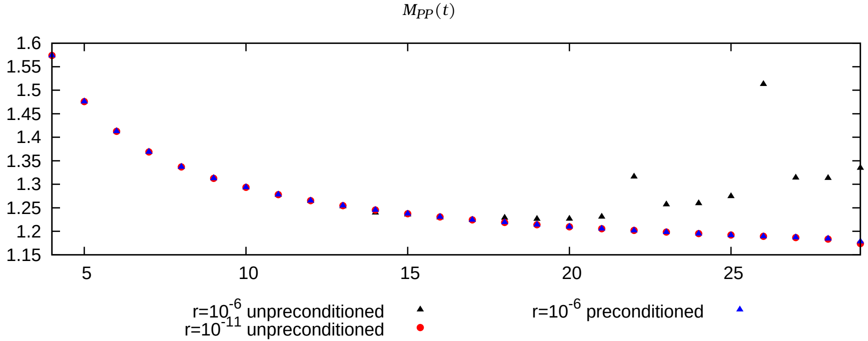


we choose:

$$\psi(\vec{x}, x_0) \longrightarrow \cosh\left[m_0(x_0 - T/2)\right] \chi(\vec{x}, x_0)$$

and, in this particular case, $m_0 = 0.4$. We calculate numerically $\chi(\vec{x}, x_0)$, and *offline* we get $\psi(\vec{x}, x_0)$

a numerical trick: free theory

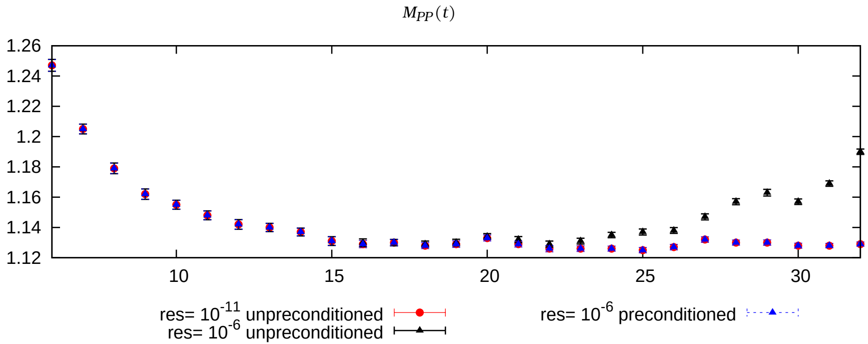


we choose:

$$\psi(\vec{x}, x_0) \longrightarrow \cosh\left[m_0(x_0 - T/2)\right] \chi(\vec{x}, x_0)$$

and, in this particular case, $m_0 = 0.4$. We calculate numerically $\chi(\vec{x}, x_0)$, and *offline* we get $\psi(\vec{x}, x_0)$

a numerical trick: interacting theory



also in the interacting case we choose:

$$\psi(\vec{x}, x_0) \longrightarrow \cosh\left[m_0(x_0 - T/2)\right]\, \chi(\vec{x}, x_0)$$

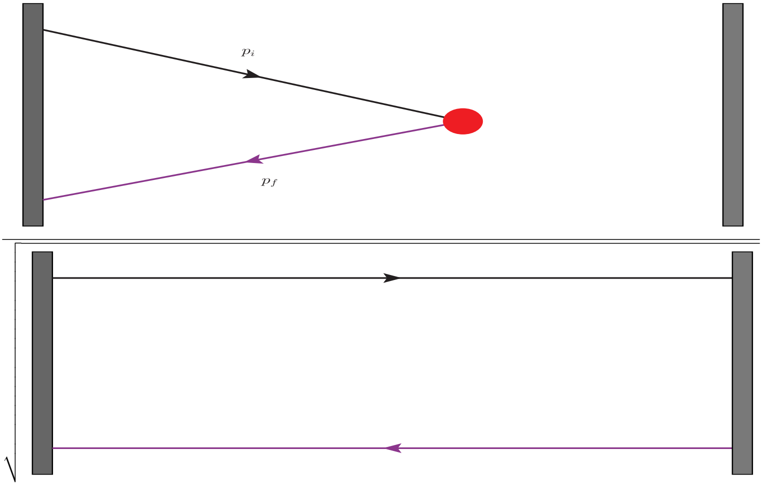
and $m_0 = 0.4$. We calculate numerically $\chi(\vec{x}, x_0)$, and *offline* we get $\psi(\vec{x}, x_0)$

a numerical trick: Schrödinger Functional setup

up to now we have been discussing the case of periodic boundary in the time direction

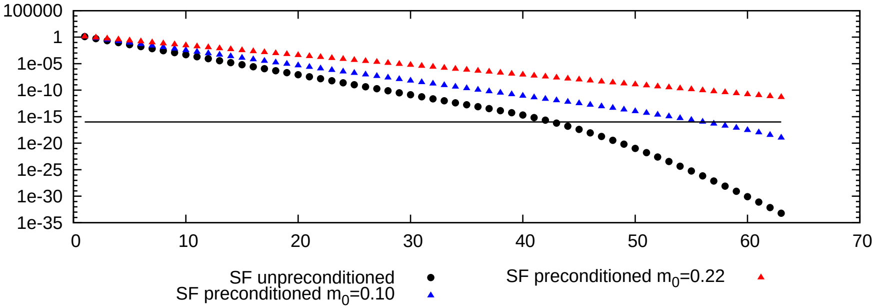
our preconditioning may be particularly relevant in the case of fixed boundary conditions in the time direction:

$F \propto$



a numerical trick: SF free theory

$$C(t) = -\mathrm{tr}\{S(0,t)S^{\dagger}(t,0)\}$$



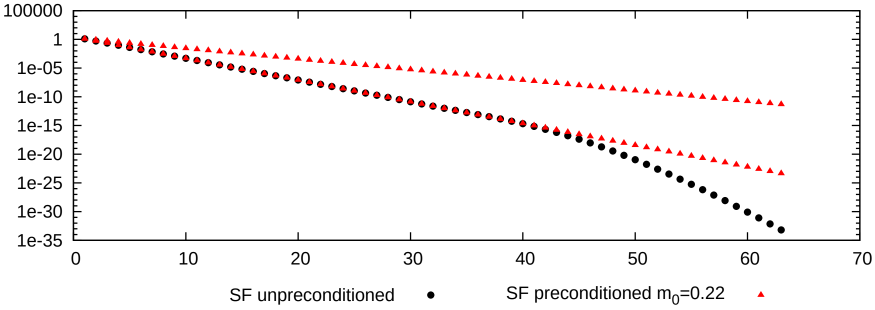
we choose:

$$\psi(\vec{x}, x_0) \longrightarrow \exp(m_0 x_0)\, \chi(\vec{x}, x_0)$$

We calculate numerically $\chi(\vec{x}, x_0)$, and *offline* we get $\psi(\vec{x}, x_0)$

a numerical trick: SF free theory

$$C(t) = -\mathrm{tr}\{S(0,t)S^{\dagger}(t,0)\}$$



we choose:

$$\psi(\vec{x}, x_0) \longrightarrow \exp(m_0 x_0)\, \chi(\vec{x}, x_0)$$

We calculate numerically $\chi(\vec{x}, x_0)$, and *offline* we get $\psi(\vec{x}, x_0)$

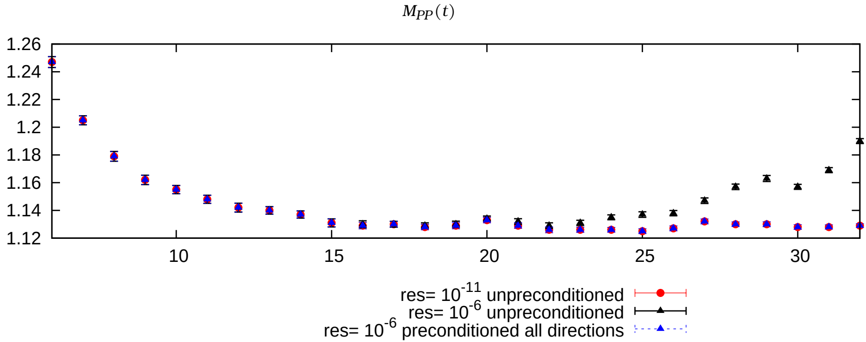
a numerical trick: different preconditioning

- the preconditioning that we have been discussing up to now
  - it is particularly simple to implement
  - solves the large time numerical precision issue for heavy quark propagators,
  - can be “removed” after having computed the correlation functions, i.e. after the contractions
- by relaxing the last property, one can as easily as before explore several other possibilities
  - extend the same trick to the other directions
  - give to the matrix α a “structure” in Dirac space
  - ...

in the following we shall briefly discuss the following preconditioning:

$$\psi(x_0, x_1, x_2, x_3) \longrightarrow \alpha(x_0)\,\alpha(x_1)\,\alpha(x_2)\,\alpha(x_3)\,\chi(x_0, x_1, x_2, x_3)$$

a numerical trick: different preconditioning



we choose ($m_0 = 0.4$):

$$\psi(x_0, x_1, x_2, x_3) \longrightarrow \left(\prod_{i=0}^{3} \cosh\left[m_0 (x_i - L_i/2)\right] \right) \chi(x_0, x_1, x_2, x_3)$$

- DD-fgcr inverter $r = 10^{-11}$ unpreconditioned: 7 iterations
- DD-fgcr inverter $r = 10^{-6}$ time preconditioned: 6 iterations
- DD-fgcr inverter $r = 10^{-6}$ all-d preconditioned: 23 iterations

the same numerical trick: speeding up numerical inversions for light quarks

we make light quark propagators decay faster by choosing:

$$\psi(x_0, x_1, x_2, x_3) \longrightarrow \left(\prod_{i=0}^{3} \frac{1}{\cosh\left[m_0(x_i - L_i/2)\right]} \right) \chi(x_0, x_1, x_2, x_3)$$

	β	$L^3 \times T$	k_{sea}	r	m_0	iterations
$D5$	5.3	$24^3 \times 48$	0.13625	10^{-11}	0.0	175
$D5$	5.3	$24^3 \times 48$	0.13625	10^{-11}	0.4	141
$E3$	5.3	$32^3 \times 64$	0.13605	10^{-10}	0.0	99
$E3$	5.3	$32^3 \times 64$	0.13605	10^{-10}	0.2	78
$E3$	5.3	$32^3 \times 64$	0.13605	10^{-10}	0.4	69
$E4$	5.3	$32^3 \times 64$	0.13610	10^{-10}	0.0	115
$E4$	5.3	$32^3 \times 64$	0.13610	10^{-10}	0.2	91
$E4$	5.3	$32^3 \times 64$	0.13610	10^{-10}	0.4	81
$E5$	5.3	$32^3 \times 64$	0.13625	10^{-10}	0.0	194
$E5$	5.3	$32^3 \times 64$	0.13625	10^{-10}	0.2	153
$E5$	5.3	$32^3 \times 64$	0.13625	10^{-10}	0.4	141

conclusions & outlooks

- we have considered a "family" of preconditionings that are easy to implement
- that can be used to perform "flavored" quark inversions on single precision architectures (e.g. GPUs, Cell, etc.) with the same numerical accuracy one would get on n-precision architectures
- on double precision machines, our preconditioning can be used to speed up the calculation of light quark propagators
- we have demonstrated that one can easily gain up to 30% in computational time without compromising the numerical accuracy
- we are exploring several other possibilities with respect to the ones discussed in this talk and, in particular, giving a Dirac "structure" to the preconditioning operator
- such kind of preconditioning may also be useful in the HMC generation of gauge field configurations