Landau Gauge Fixing on GPUs

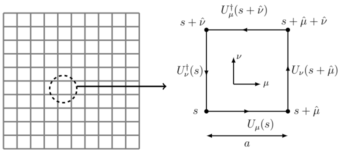
Nuno Cardoso*, **Pedro Bicudo***,
Paulo J. Silva† and **Orlando Oliveira**†

*CFTP, Departamento de Física, Instituto Superior Técnico,
Universidade Técnica de Lisboa, Av. Rovisco Pais, 1049-001 Lisbon,
Portugal
†CFC, Departamento de Física, Faculdade de Ciências e Tecnologia,
Universidade de Coimbra, 3004-516 Coimbra, Portugal

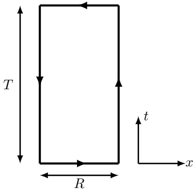
Excited QCD 2012

10/05/2012

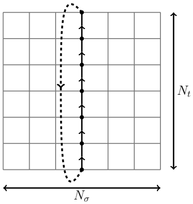
Gauge invariant operators in LQCD



- Plaquette → E_i^2 and B_i^2



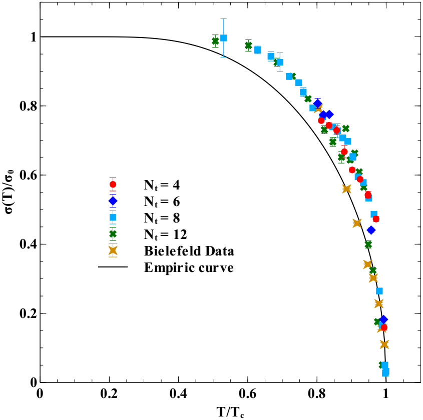
- Wilson Loop → quark-antiquark potential



- Polyakov Loop →
  $\langle L(x)\rangle = \prod_{t=0}^{N_\tau-1} U_{\mu=0}(x,t) \propto \exp\left(-\frac{F_q}{T}\right)$
  $\langle L\rangle = 0$ → **confinement**
  $\langle L\rangle \neq 0$ → **deconfinement**
- Two Polyakov Loops → color averaged free energy:
  $e^{-F_{\mathbf{avg}}(r,T)/T+C} = \frac{1}{N^2}\left\langle \mathrm{Tr}\, L(y)\mathrm{Tr}\, L^\dagger(x)\right\rangle$

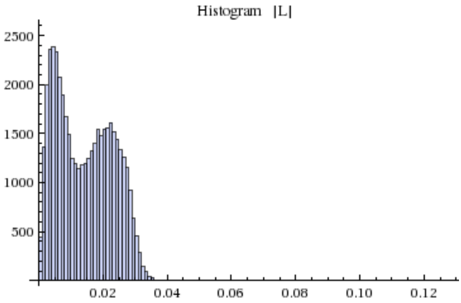
String Tension at Finite Temperature

- String tension extracted from **Colour Average Free Energy**
  - Gauge Invariant
- 4 Lattice volumes: $48^3 \times 4$, $48^3 \times 6$,$48^3 \times 8$ and $48^3 \times 12$
- $\sim$ 13200 independent configurations
- To improve the signal in the Polyakov loop correlation functions and reduce the error, we employ the multi-hit procedure.



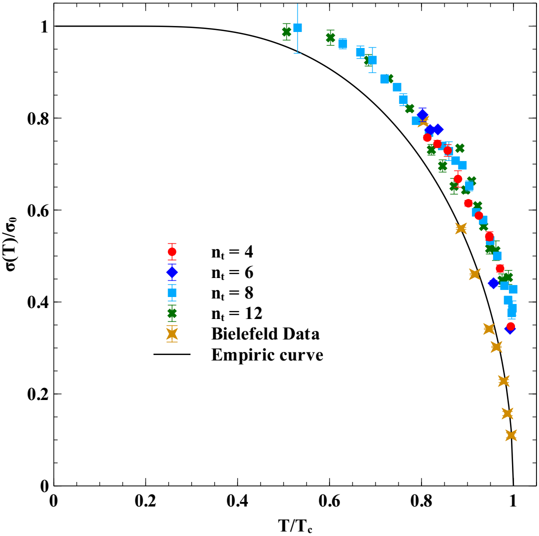
String Tension at Finite Temperature

- Near the phase transition, there are configurations in the wrong phase.
- It is necessary to analyse the Polyakov loop history.
- For example: for $48^3 \times 8$, $T = 0.998\,T_c$



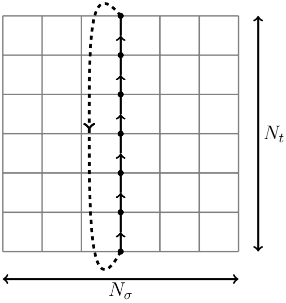
- Double peak structure → second peak in the deconfinement region.

String Tension at Finite Temperature



- N. Cardoso and P. Bicudo, PhysRevD.85.077501, 2012

More Operators in LQCD



- Polyakov Loop →
  $\langle L(x)\rangle = \prod_{t=0}^{N_\tau-1} U_{\mu=0}(x,t) \propto \exp\left(-\frac{F_q}{T}\right)$
  $\langle L\rangle = 0 \rightarrow$ **confinement**
  $\langle L\rangle \neq 0 \rightarrow$ **deconfinement**
- Two Polyakov Loops → color averaged free energy:
  $e^{-F_{\mathbf{avg}}(r,T)/T+C} = \frac{1}{N^2}\left\langle \mathrm{Tr}\, L(y) \mathrm{Tr}\, L^\dagger(x)\right\rangle$

- Two Polyakov Loops → Singlet free energy:
  $e^{-F_{\mathbf{singlet}}(r,T)/T+C} = \frac{1}{N}\left\langle \mathrm{Tr}\, \left(L(y)\, L^\dagger(x)\right)\right\rangle \rightarrow$ **NOT GAUGE INVARIANT**

- we will use **Landau Gauge Fixing**

Landau Gauge Fixing

- On the lattice, the Landau gauge is defined by maximising the functional

$$F_U[g] = \frac{1}{4N_c V} \sum_x \sum_\mu \mathrm{Re}\left[\mathrm{Tr}\left(g(x)U_\mu(x)g^\dagger(x+\mu)\right)\right]$$

  with N_c the dimension of the gauge group and V the lattice volume.

- On the gauge fixing process, the quality of the gauge fixing is measured by

$$\theta = \frac{1}{N_c V}\sum_x \mathrm{Tr}\left[\Delta(x)\Delta^\dagger(x)\right]$$

  where

$$\Delta(x) = \sum_\nu \left[U_\nu(x - a\hat{\nu}) - U_\nu(x) - \text{h.c.} - \text{trace}\right]$$

  is the lattice version of $\partial_\mu A_\mu = 0$.

- For Landau gauge fixing, a popular local optimization method is the **steepest descent method**.

- However, when it is applied to larger lattices, this method faces the problem of critical slowing down.

Landau Gauge Fixing

- This problem can be attenuated by **Fourier acceleration**, Davies et al., 1987.
- In the Fourier accelerated method, at each iteration one chooses

$$g(x) = \exp\left[\hat{F}^{-1}\frac{\alpha}{2}\frac{p^2_{\max}a^2}{p^2a^2}\hat{F}\left(\sum_\nu \Delta_{-\nu}\left[U_\nu(x) - U^\dagger_\nu(x)\right] - \text{trace}\right)\right]$$

  with

$$\Delta_{-\nu}\left(U_\mu(x)\right) = U_\mu(x - a\hat{\nu}) - U_\mu(x)$$

  p^2 are the eigenvalues of $\left(-\partial^2\right)$, a is the lattice spacing and $\hat{F}$ represents a fast Fourier transform (FFT).
- For the parameter α, we use the value 0.08.
- For numerical purposes, it is enough to expand to first order the exponential, followed by a reunitarization.

Landau Gauge Fixing

```
calculate $\Delta(x)$, $F_g[U]$ and $\theta$
while $\theta \geq \epsilon$ do
    for all element of $\Delta(x)$ matrix do
        apply FFT
        apply $p^2_{\mathrm{max}}/p^2$
        apply IFFT
        normalize
    end for
    for all $x$ do
        obtain $g(x)$ and reunitarize
    end for
    for all $x$ do
        for all $\mu$ do
            $U_\mu(x) \rightarrow g(x)U_\mu(x)g^\dagger(x+\hat{\mu})$
        end for
    end for
    calculate $\Delta(x)$, $F_g[U]$ and $\theta$
end while
```

12 real number parametrization

In order to reduce memory traffic we use the unitarity of SU(3) matrices and store only the 1st 2 rows (12 real numbers) and reconstruct the 3rd row on the fly when needed instead of storing it,

$$\begin{pmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{pmatrix} \in \mathrm{SU}(3), \qquad \mathbf{c} = (\mathbf{a} \times \mathbf{b})^* \tag{1}$$

Although, in SU(3), we can use 8 real numbers, this approach proves to be numerically unstable, M. Clark et. al, Comput.Phys.Commun.181, 2010 and R. Babich, arXiv:hel-lat/1011.0024.

FFT on GPUs: CUFFT library by NVIDIA

- support for 1D, 2D and 3D FFTs;
- there is no direct support for 4D FFTs
- solution: 4 1D FFT's, 2 2D FFTs or 3D+1D FFTs
    - 4 1D FFTs: necessary to change de order of the data 7 times
        - The drawback with this approach is that the time it takes to change the order of the data can be longer than to actually perform the 1D FFT.
    - 3D + 1D FFTs: necessary to change the order of the data 2 times
    - 2D + 2D FFTs: necessary to change the order of the data 2 times
- cufftPlanMany() allows the execution of multiple independent FFts in parallel, which supports launching a batch of 1D, 2D and 3D FFTs.
- best choice for our 4D problem: 2D + 2D FFTs.

CUDA Kernels and FLOPs

kernel	18real		12real	
per thread	load/store	flop	load/store	flop
k1	0/1	20	0/1	20
k2	144/20	505	96/14	841
k3	2/2	0	2/2	0
k4	3/1	2	3/1	2
k5	18/18	153	12/12	153
k6	162/72	1584	108/48	1962

Kernels:

- k1: kernel to obtain an array with $p^2_{\max}/p^2$.
- k2: kernel to calculate $\Delta(x)$, $F_g[U]$ and θ. The sum of $F_g[U]$ and θ over all the lattice sites are done with the parallel reduction code in the NVIDIA GPU Computing SDK package.
- k3: kernel to perform a data ordering.
- k4: apply $p^2_{\max}/p^2$ and normalize.
- k5: obtain $g(x)$ and reunitarize.
- k6: perform $U_\mu(x) \to g(x)U_\mu(x)g^\dagger(x+\hat{\mu})$.

The number of floating-point operations using 2D+2D FFTs is given by $\mathit{volume} \times 5(\log_2(n_x \times n_y) + \log_2(n_z \times n_t))$.

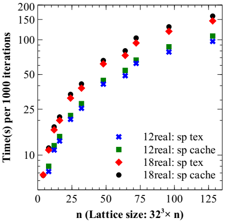
GPU

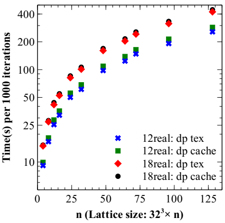
NVIDIA Tesla C2070	
Number of GPUs	1
CUDA Capability	2.0
Multiprocessors (MP)	14
Cores per MP	32
Total number of cores	448
Core clock	1.15 GHz
Global memory	6144 MB
Memory Bandwidth	144 GB/s
Shared memory (per SM)	48KB or 16KB
L1 cache (per SM)	16KB or 48KB
L2 cache (chip wide)	768KB
Clock rate	1.15 GHz
ECC support	yes

Server example with 8 Tesla GPUs

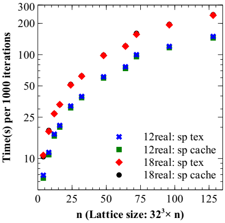
GPU Performance results

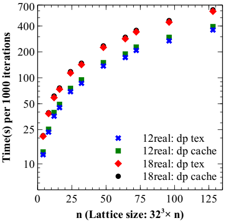
- With ECC OFF



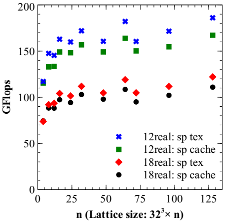


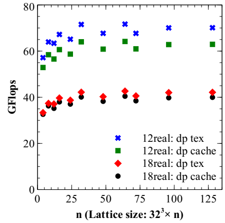
- With ECC ON



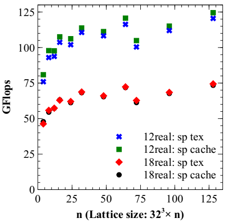


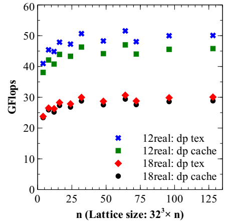
- With ECC OFF





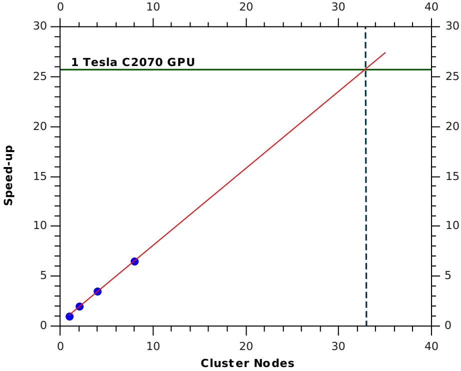
- With ECC ON





CPU versus GPU

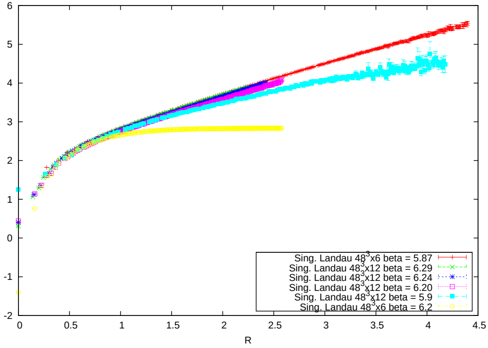
- CPU code: With **Chroma Library** for Lattice Field Theory + PFFT of Michael Pippig
  - Cluster Centaurus at Coimbra
  - 16 nodes
  - per node: 8 cores and 24GB per node with 2 processors Intel Xeon E5620@2.4 GHz (quad core)
  - Infiniband DDR (20GB/s)
  - 2700€+taxes per node.
- GPU code: 12 real number and using Texture Memory and ECC OFF
- Lattice volume: 32^4 in **Double Precision**
- Landau Gauge Fixing Parameters: $\alpha = 0.08$ and $\theta < 10^{-15}$



- **1 Tesla C2070 GPU $\sim$ 32 nodes $\sim$ 256 cores**

String Tension at Finite Temperature

- String tension extracted from **Singlet Energy**
  - Landau Gauge Fixing



β	N_t	T/T_c
6.20	6	1.639
5.87	6	0.957
6.29	12	0.935
6.24	12	0.871
6.20	12	0.822
5.90	12	0.507

$\rightarrow$ PRELIMINARY Results

- For T $>$ Tc:
  - $\sigma/\sigma_0 = 0$
- For T $<$ Tc:
  - $\sigma \sim 0.7\sigma_0$ and temperature independent.
- String tension from Singlet in Landau gauge is constant and temperature independent in the confined phase.

Thanks