Introduction to multilevel algorithms

I: Basic strategy

Martin Lüscher, Theoretical Physics Department, CERN

School on "Computing Challenges in HEP @Exascale", Benasque, 20. September 2022

Current technical limitations in lattice QCD

- Rapidly growing autocorrelation times in the continuum limit
- Exponential loss of statistical precision at large distances
- ...

The primary goal of the multilevel strategy to be discussed is to improve the precision of hadron propagator computations

But there are further applications:
Wilson loops, glueball spectrum, generation of master fields, ...

As a representative case, consider . . .

Fields of classical spins $\vec{s}(x)$ of unit length, $\vec{s}(x)^2 = 1$, on a 2d lattice

$$S(\vec{s}) = -\beta \sum_x \sum_\mu \vec{s}(x) \cdot \vec{s}(x+\hat{\mu})$$

$$\mathcal{Z} = \int \mathrm{D}[\vec{s}]\, \mathrm{e}^{-S(\vec{s})}, \qquad \langle \mathcal{O} \rangle = \frac{1}{\mathcal{Z}} \int \mathrm{D}[\vec{s}]\, \mathcal{O}(\vec{s}) \mathrm{e}^{-S(\vec{s})}$$

Will focus on computing the 2-point function

$$G(x-y) = \langle \vec{s}(x) \cdot \vec{s}(y) \rangle$$

Markov-chain MC

Generate a chain of fields

$$\vec{s}_1(x) \to \vec{s}_2(x) \to \vec{s}_3(x) \to \dots$$

according to some *transition probability density* $T(\vec{s} \to \vec{s}')$

$$\int \mathrm{D}[\vec{s}']\, T(\vec{s} \to \vec{s}') = 1$$

$$\int \mathrm{D}[\vec{s}]\, \mathrm{e}^{-S(\vec{s})} T(\vec{s} \to \vec{s}') = \mathrm{e}^{-S(\vec{s}')}$$

+ ergodicity requirements

$$\Rightarrow \quad \overline{\mathcal{O}} = \frac{1}{N} \sum_{k=1}^{N} \mathcal{O}(\vec{s}_k) = \langle \mathcal{O} \rangle \pm \text{statistical error}$$

$$\text{statistical error} = \sqrt{\frac{2\tau(\mathcal{O})}{N}}\sigma(\mathcal{O})$$

$\tau(\mathcal{O}) =$ integrated autocorrelation time of $\mathcal{O}$

$\sigma(\mathcal{O}) =$ standard deviation of $\mathcal{O}$

- $\tau(\mathcal{O})$ depends on the simulation algorithm
- $\sigma(\mathcal{O})$ only on $\mathcal{O}$

Exponential loss of precision

In the case of $\mathcal{O} = \vec{s}(x) \cdot \vec{s}(y)$ and at large $|x - y|$

$$\langle \mathcal{O} \rangle \propto \mathrm{e}^{-m|x-y|}$$

but

$$\frac{1}{2} \leq \tau(\mathcal{O}) \leq \tau_{\mathrm{exp}} \qquad \text{(a general property of Markov-chain MC)}$$

$$\sigma(\mathcal{O}) = \mathrm{O}(1) \qquad \text{(explained later)}$$

and thus

$$\text{relative statistical error} \propto \frac{\mathrm{e}^{m|x-y|}}{\sqrt{N}}$$

Ways to reduce the error

1. More powerful computers = wasteful, intellectual defeat!

2. Better simulation algorithms $\Rightarrow$ smaller $\tau(\mathcal{O})$, but no exponential improvement

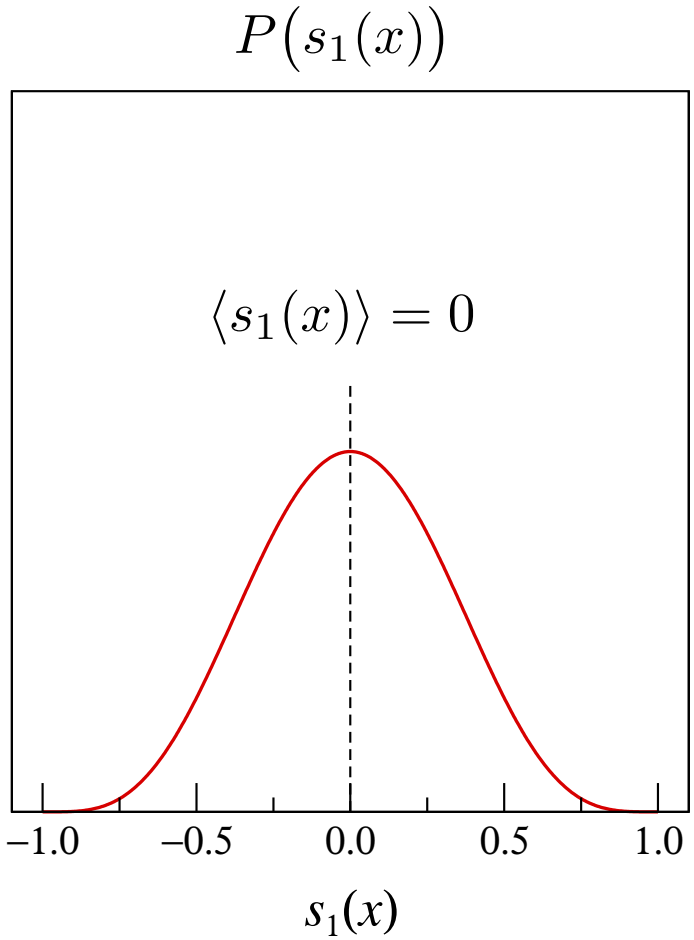
3. Construct observables $\mathcal{O}'$ such that

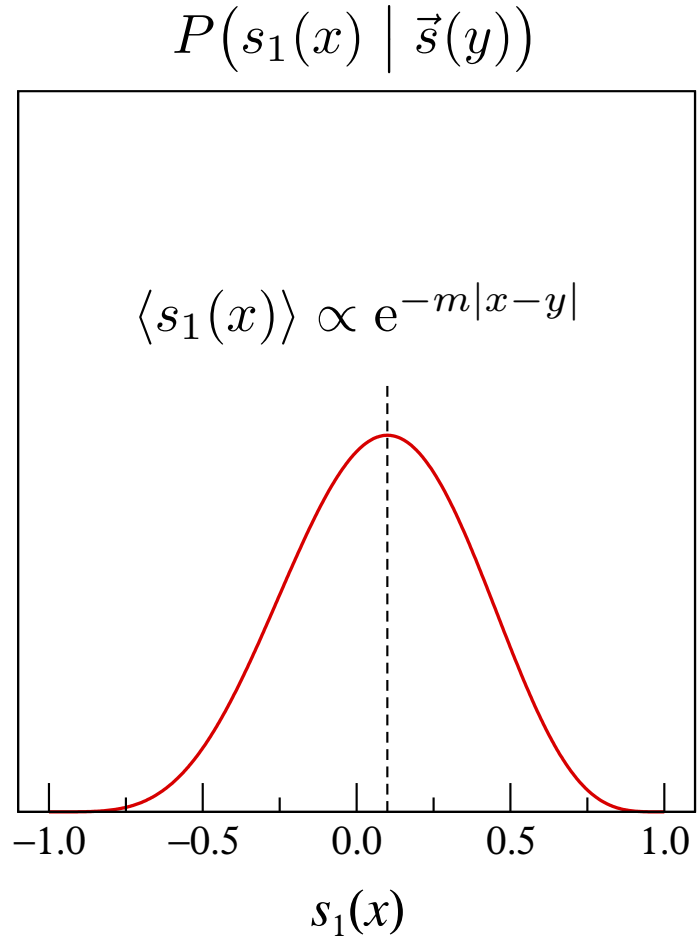
$$\langle \mathcal{O}' \rangle = \langle \mathcal{O} \rangle \quad \text{and} \quad \sigma(\mathcal{O}') \ll \sigma(\mathcal{O})$$

Improvement can be exponential!

The multilevel strategy is to construct improved observables by exploiting the locality of the theory

Why is $\sigma(\mathcal{O}) = \mathrm{O}(1)$?



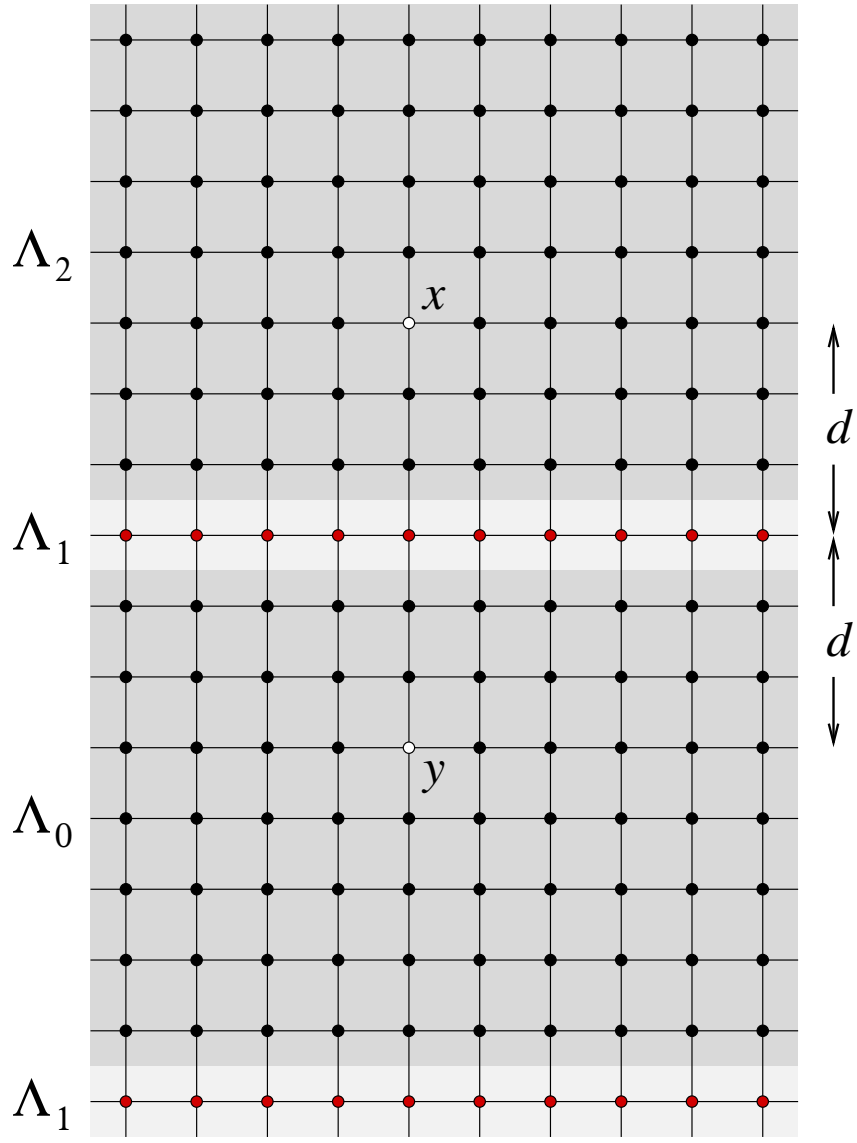


Local sampling

Consider a $T \times L$ lattice with periodic boundary conditions

At fixed spins on Λ_1:

- Λ_0 *and* Λ_2 *are decoupled*
- *Spin fields on* Λ_0, Λ_2 *may be sampled independently*
- *E.g. through a HMC algorithm*



2-level algorithm

Level 0: n_0 global updates

Level 1: n_1 local updates at fixed spins on Λ_1

$$\underbrace{\vec{s}_1 \to \vec{s}_2 \to \ldots \to \vec{s}_{n_0}}_{\text{global}} \to \underbrace{\vec{s}_{n_0+1} \to \vec{s}_{n_0+2} \to \ldots \to \vec{s}_{n_0+n_1}}_{\text{local}} \to \ldots$$

- Perform N such update cyles
- For all cycles $k = 1, \ldots, N$ compute

$$[\vec{s}(x)]_k = \frac{1}{n_1} \sum_{j=1}^{n_1} \vec{s}_{k(n_0+n_1)+j}(x) \quad \text{and} \quad [\vec{s}(y)]_k = \ldots$$

$$\Rightarrow \quad \langle \vec{s}(x) \cdot \vec{s}(y) \rangle = \frac{1}{N} \sum_{k=1}^{N} [\vec{s}(x)]_k \cdot [\vec{s}(y)]_k \pm \text{statistical error}$$

Is this a correct algorithm?

Lemma: *If* $T_1(\vec{s} \to \vec{s}')$ *and* $T_2(\vec{s} \to \vec{s}')$ *are valid transition probabilities, so is*

$$(T_1 T_2)(\vec{s} \to \vec{s}') = \int \mathrm{D}[\vec{r}]\, T_1(\vec{s} \to \vec{r}) T_2(\vec{r} \to \vec{s}')$$

In particular, $T_1 T_2$ *preserves the equilibrium distribution* $\mathrm{e}^{-S(\vec{s})}$.

$\Rightarrow$ The spins

$$\vec{s}_{k(n_0+n_1)+i}(x) \quad \text{and} \quad \vec{s}_{k(n_0+n_1)+j}(y)$$

have the correct joint distribution, for all i, j, and

$$[\vec{s}(x)]_k \cdot [\vec{s}(y)]_k = \frac{1}{n_1^2} \sum_{i,j=1}^{n_1} \vec{s}_{k(n_0+n_1)+i}(x) \cdot \vec{s}_{k(n_0+n_1)+j}(y)$$

is therefore a correct stochastic estimator of the 2-point function

Choice of n_0 and n_1

1. Choose n_1 such that

$$\sigma([s_1(x)]) \sim \langle [s_1(x)] \rangle$$

at fixed $\vec{s}(z)$ on $\partial\Lambda$

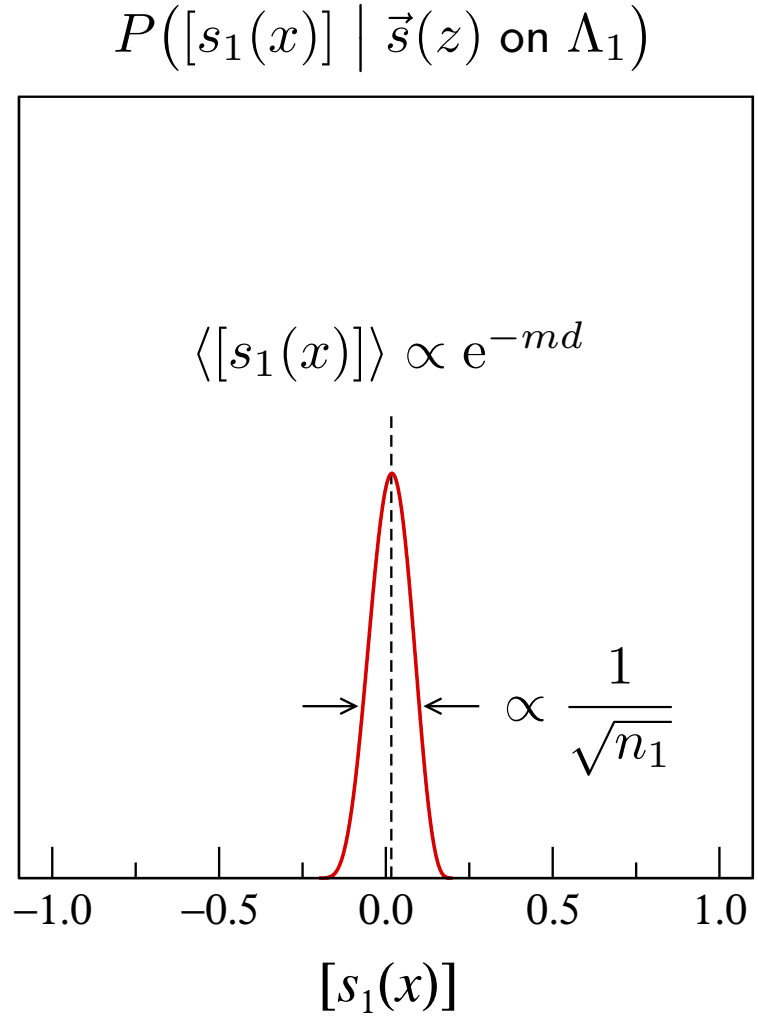
2. Since

$$[\vec{s}(x)] \cdot [\vec{s}(y)] \sim \mathrm{e}^{-2md} = \mathrm{e}^{-m|x-y|}$$

the statistical error is then of order

$$\sqrt{\frac{2\tau}{N}} \mathrm{e}^{-m|x-y|}$$

3. Increase n_0 until $\tau \simeq \frac{1}{2}$



Scaling behaviour

The number of updates is $N\,(n_0 + n_1)$, but only

$$n_1 \propto \mathrm{e}^{2md} = \mathrm{e}^{m|x-y|}$$

scales exponentially with $|x - y|$

With respect to the 1-level algorithm, where $N \propto \mathrm{e}^{2m|x-y|}$, an exponential improvement is thus achieved!

Essentially a consequence of the representation

$$\langle \vec{s}(x) \cdot \vec{s}(y) \rangle = \langle [\vec{s}(x)] \cdot [\vec{s}(y)] \rangle$$

through exponentially small factors that can be computed independently

If you have some spare time . . .

Design a 3-level algorithm with an even better scaling behaviour!