Monte Carlo algorithms for lattice QCD

Speaker: Lorenzo Barca (DESY Zeuthen)

Project is part of DFG-Research unit *FOR5269*
“Future methods for studying confined gluons in QCD“

FH SciComp Workshop

DESY. | Hamburg | 02/07/2024

FOR 5269

DESY.

Extracting physics from Monte Carlo simulations

Exact result

$$\langle \mathrm{O} \rangle = \frac{1}{\mathcal{Z}} \int [dU][d\psi][d\bar{\psi}]\; e^{-S[U,\psi,\bar{\psi}]}\; \mathrm{O}[U,\psi,\bar{\psi}]$$

generate U_i with probability $dP \propto e^{-S}$

Monte Carlo estimate

$$\langle \hat{\mathrm{O}} \rangle = \frac{1}{N} \sum_i^N \mathrm{O}_i \pm \mathcal{O}\left(\frac{\sqrt{var(\mathrm{O})}}{\sqrt{N}} \right)$$

Proton two-point functions

$$\langle \hat{\mathrm{O}}^p(t) \hat{\mathrm{O}}^p(0)^\dagger \rangle = \frac{1}{N} \sum_i^N \hat{\mathrm{O}}_i^p(t)\; \hat{\mathrm{O}}_i^p(0)^\dagger \;\sim\; e^{-m_p t}$$

Extract m_p from an exponential fit

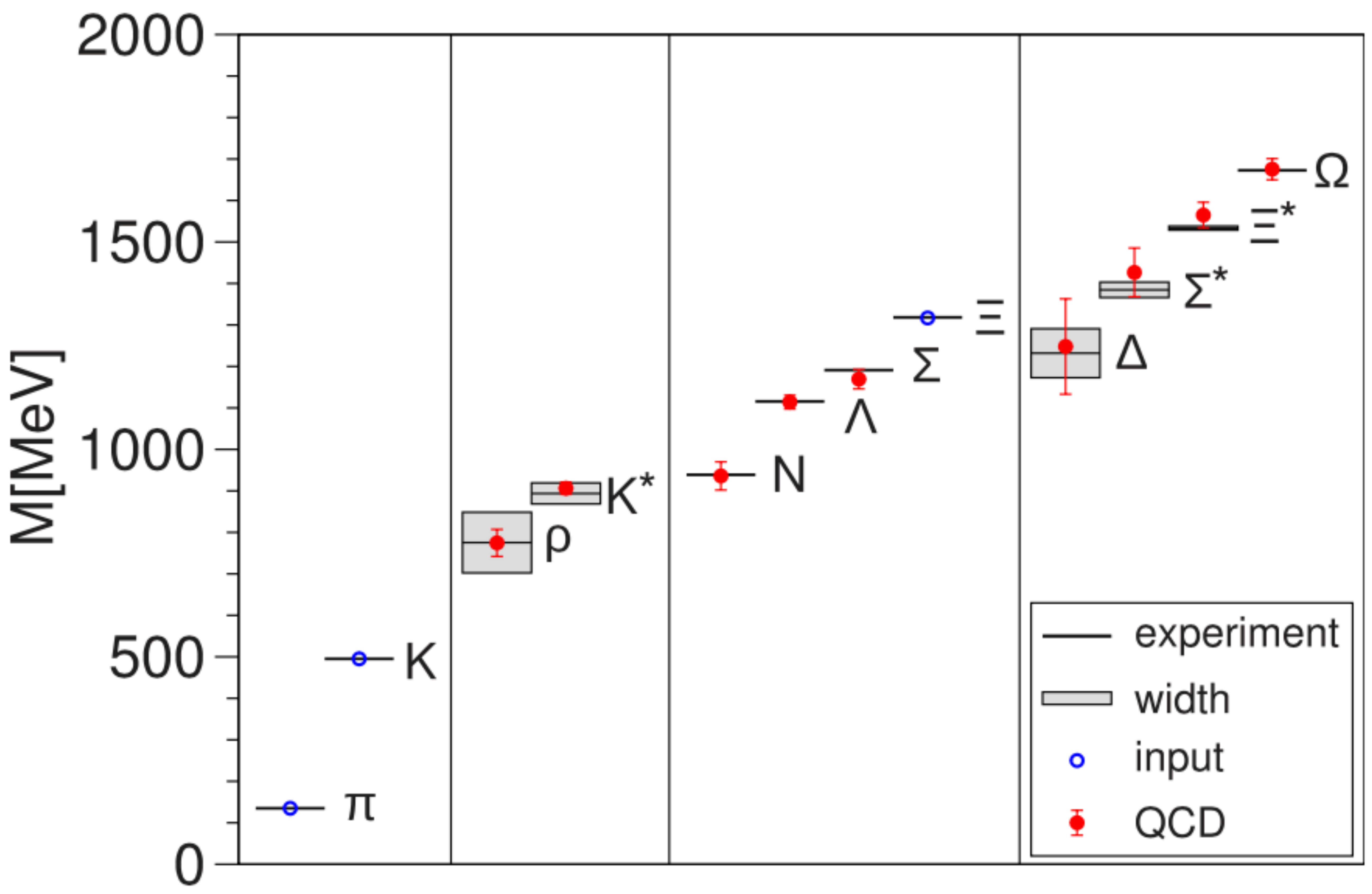
Proton three-point functions

$$\langle \hat{\mathrm{O}}^p(t) J_\mu^{EM}(\tau) \hat{\mathrm{O}}^p(0)^\dagger \rangle = \frac{1}{N} \sum_i^N \hat{\mathrm{O}}_i^p(t)\; J_{\mu,i}^{EM}(\tau)\; \hat{\mathrm{O}}_i^p(0)^\dagger \;\propto\; G_E, G_M$$

Information on internal structure

Extract form factors G_E, G_M

Low-lying hadron spectrum



BMW Collaboration

[Science 322, Issue 5905 2008]

High dimensional integrals

Exact result

$$\langle \mathrm{O} \rangle = \frac{1}{\mathcal{Z}} \int [dU][d\psi][d\bar{\psi}]\ e^{-S[U,\psi,\bar{\psi}]}\ \mathrm{O}[U,\psi,\bar{\psi}]$$

Integrating out fermions

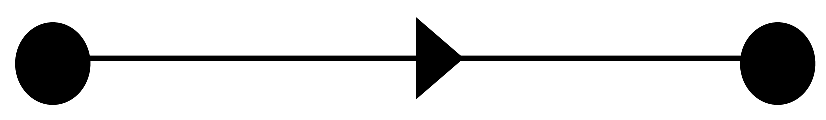
$$= \frac{1}{\mathcal{Z}_g} \int [dU]\ \det(\mathrm{D}^{-1})^{\mathrm{N_f}}\ e^{-S_g[U,\psi,\bar{\psi}]}\ < \mathrm{O}[D^{-1}(U)] >_W$$

After integrating fermionic fields

Dependence on quark propagators $D^{-1}(U)$

Quark propagators are **high-dimensional** objects $V^2 \times N_c^2 \times N_s^2$ (Space-time, color, spin)

$$D^{-1}(\vec{x}, x_0; \vec{y}, y_0)^{ab}_{\alpha\beta}$$



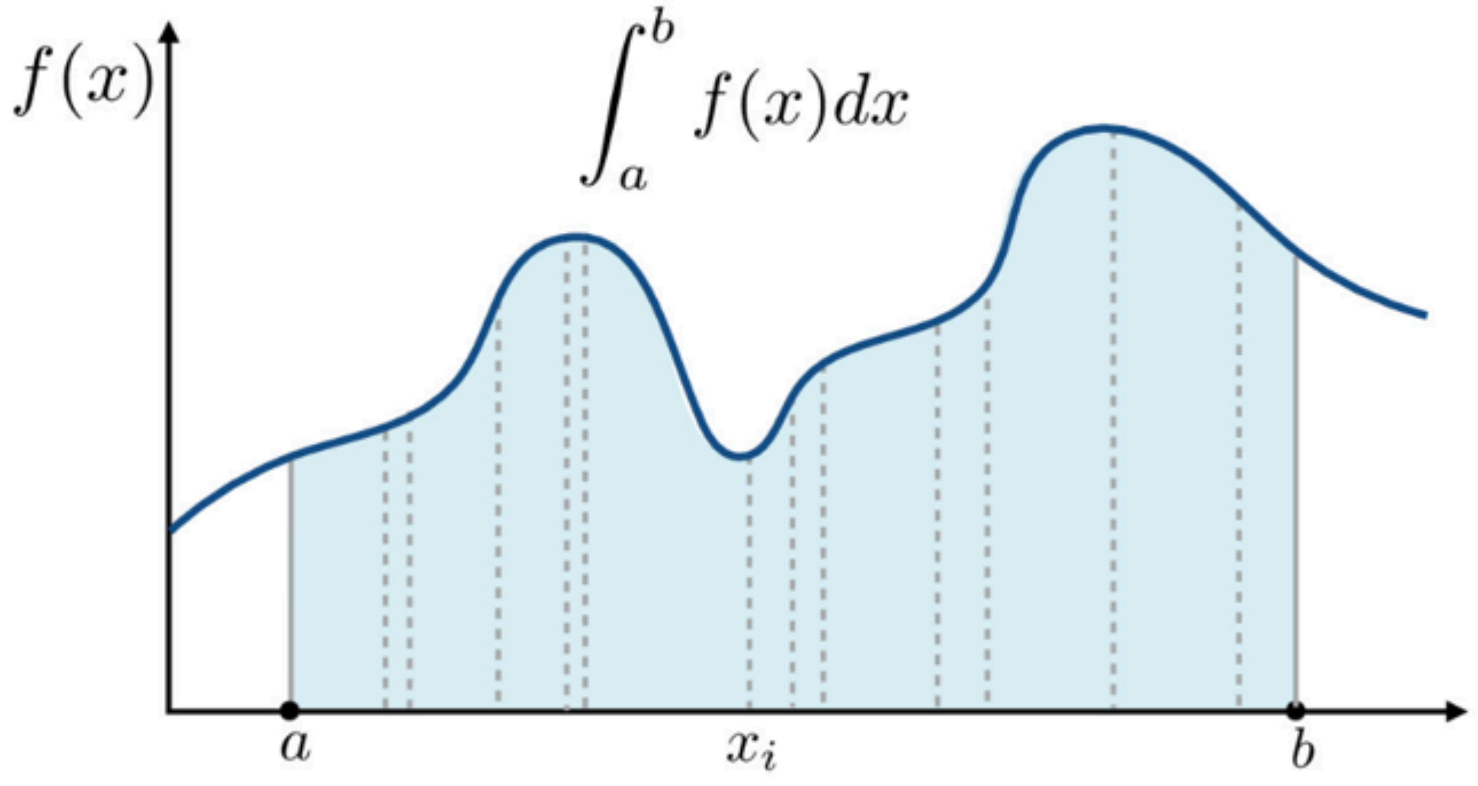
$x = (x_0, \vec{x})$ $\quad$ $y = (y_0, \vec{y})$

Use Monte Carlo to compute high-dimensional integrals

Monte Carlo samplings for High dimensional integrals

$$\frac{1}{b-a}\int_a^b dx\, f(x) \quad = \quad lim_{N\to\infty}\frac{1}{N}\sum_i^N f(x_i)$$

x_i randomly chosen from distribution $\rho(x_i) = \dfrac{1}{b-a}$



$$\frac{1}{\mathcal{Z}_g}\int [dU]\ \det(\mathrm{D}^{-1})^{\mathrm{N_f}}\ e^{-S_g[U,\psi,\bar{\psi}]}\ < \mathrm{O}[D^{-1}(U)] >_W \quad = \quad lim_{N\to\infty}\frac{1}{N}\sum_i^N f(U_i)$$

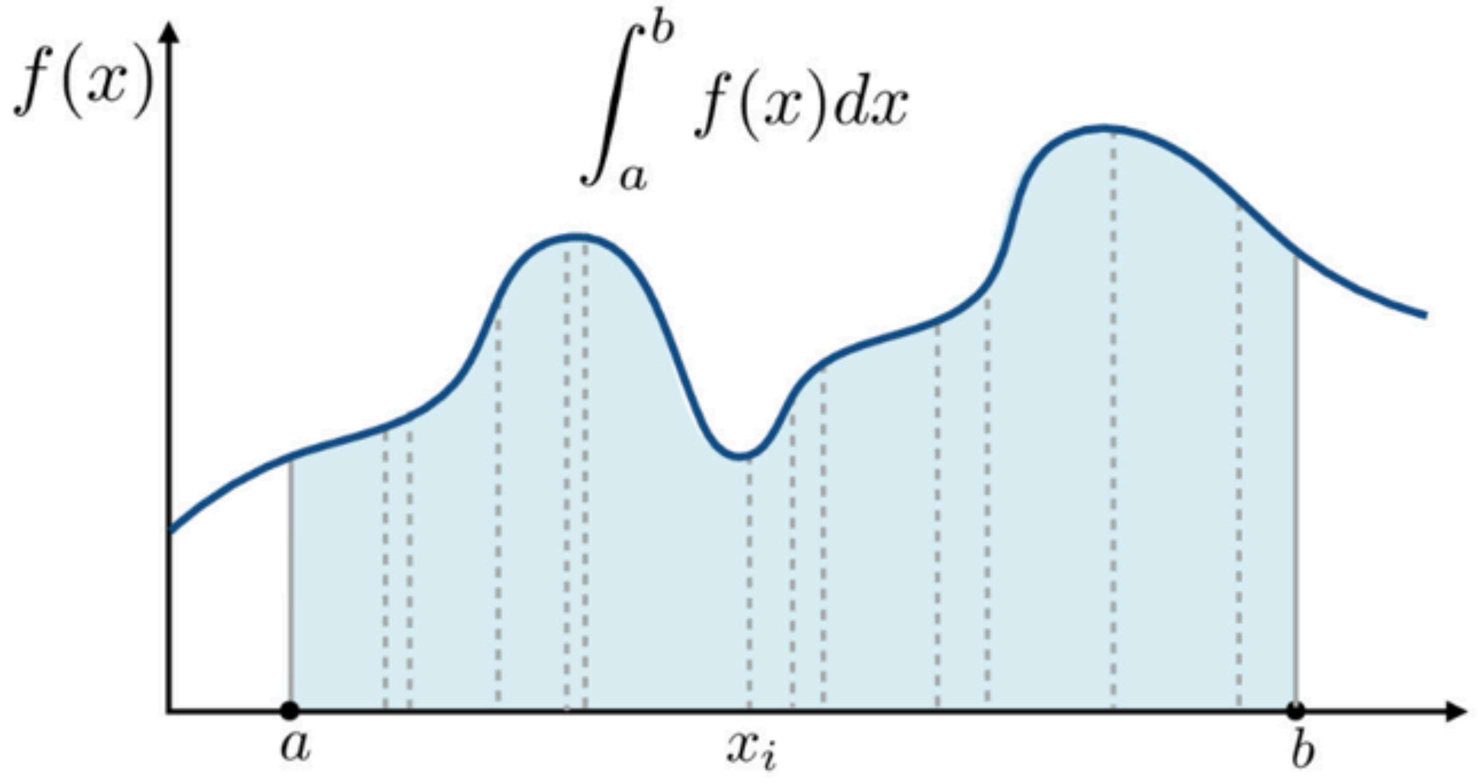
U_i randomly chosen from probability distribution

U_i are multi-dimensional vectors

$$\rho(U_i) = \frac{e^{-S_g}(U_i)}{Z_g}$$

Monte Carlo samplings for High dimensional integrals

$$\frac{1}{b-a}\int_a^b dx\, f(x) \quad = \quad \cancel{lim}_{N\to\infty}\frac{1}{N}\sum_i^N f(x_i) + \mathcal{O}\left(\frac{\sqrt{var(f)}}{\sqrt{N}}\right)$$



x_i randomly chosen from distribution $\rho(x_i) = \dfrac{1}{b-a}$

$$\frac{1}{\mathcal{Z}_g}\int [dU]\ \det(D^{-1})^{N_f}\ e^{-S_g[U,\psi,\bar\psi]}\ \ < O[D^{-1}(U)] >_W \quad = \quad \cancel{lim}_{N\to\infty}\frac{1}{N}\sum_i^N f(U_i) + \mathcal{O}\left(\frac{\sqrt{var(f)}}{\sqrt{N}}\right)$$

U_i randomly chosen from probability distribution

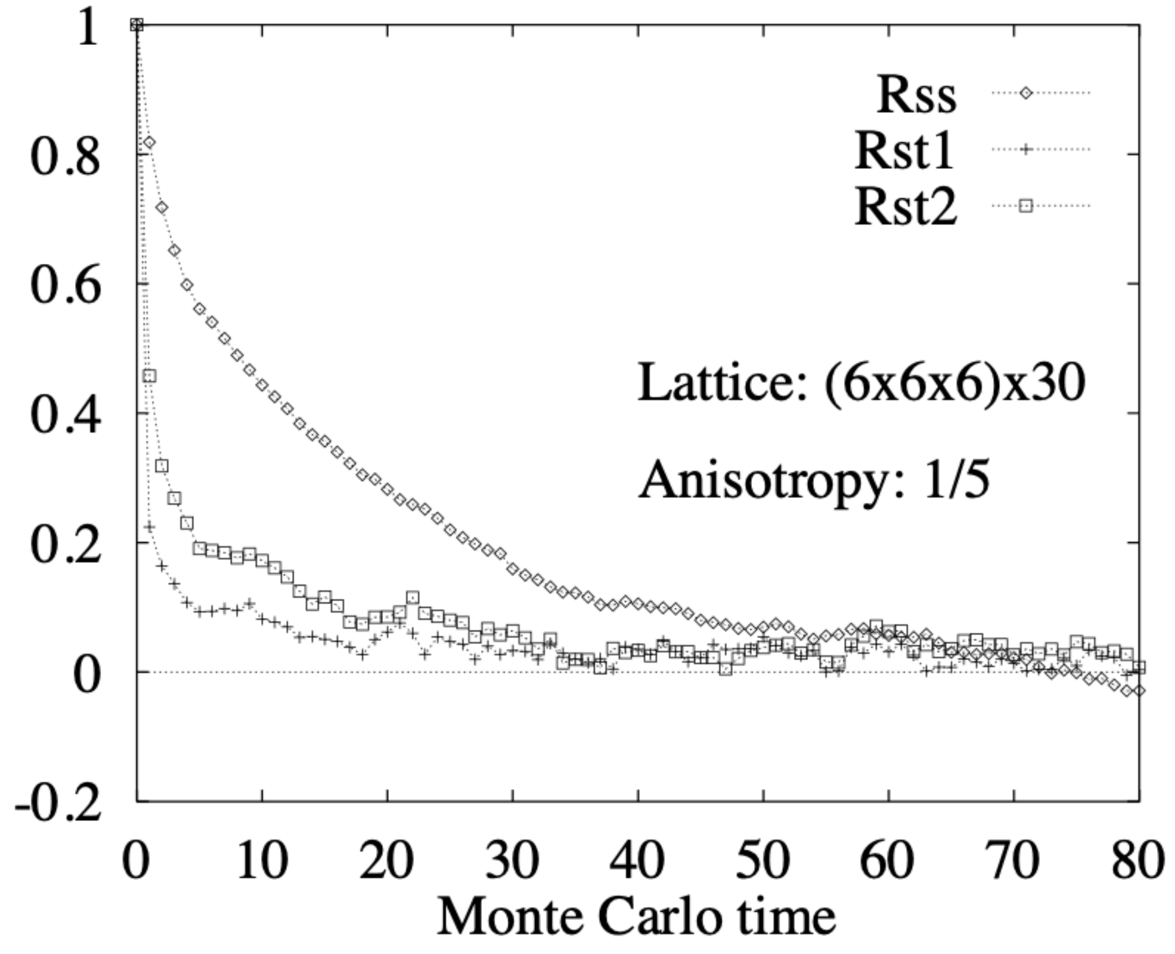
U_i are multi-dimensional vectors

$$\rho(U_i) = \frac{e^{-S_g}(U_i)}{Z_g}$$

Standard Algorithm

well spaced = **autocorrelations**

Since subsequent configurations are built from previous ones, U_i are correlated



Generate N thermalised gauge configurations (HMC)

Compute $O_i(t)$ and correlate them

$$C(t) = \frac{1}{N} \sum_{i}^{N} O_i(t) O_i(0)$$

High dimensional integrals demand High Performance Computing

$$\frac{1}{N}\sum_{i}^{N} f(U_i) + \mathcal{O}\left(\frac{\sqrt{var(f)}}{\sqrt{N}}\right)$$

Bottlenecks:

Generation of many U_i to reach good accuracy

Computation of $D^{-1}(U_i)$ $\quad V^2 \times N_c^2 \times N_s^2$

$V = T \times L^3 = 192 \times 96^3$ lattice

Example of expenses for 1 config U

CPU	78 kch	(8192 cores x 9.5 h)
Nominal cost	780 Eur	(1 cent/ch)
Energy	780 kWh	(10 W/core)
CO_2	550 kg	(0.7 kh/kW)

Need **supercomputers**	**Nodes**	**Cores**
Large scale: SuperMUC-NG (Munich)	6336	48
Small scale: e.g. PAX (Zeuthen)	16	64

Some computations are too expensive

noise

$$\langle \hat{\mathrm{O}} \rangle = \frac{1}{N}\sum_{i}^{N} \mathrm{O}_i \pm \boxed{\mathcal{O}\left(\frac{\sqrt{var(\mathrm{O})}}{\sqrt{N}}\right)}$$

For some $\langle \hat{\mathrm{O}} \rangle$, $N \sim \mathcal{O}(10^6 - 10^9)$ Monte Carlo samples are required

to reduce noise and extract physical information with precision $\lesssim 5\,\%$.

(huge expenses on previous lattice sizes)

Better algorithms: Multilevel Monte Carlo

[G. Parisi 1983]

[M. Lüscher, P. Weisz 2001]

[H. Meyer 2003]

If action S is local and operators are local, the correlation function

$$\langle \mathrm{O}(t_1)\mathrm{O}(t_0)\rangle = \frac{1}{\mathcal{Z}} \int [dU] e^{-S[U]}\, \mathrm{O}(U,t_1)\mathrm{O}(U,t_0)$$

can be factorised into a product of integrals

$$\langle \mathrm{O}(t_1)\mathrm{O}(t_0)\rangle = \frac{1}{\mathcal{Z}} \int [dU_B] e^{-S_B[U_B]}\, \left[\mathrm{O}^{(2)}(U_B,t_1)\right] \left[\mathrm{O}^{(1)}(U_B,t_0)\right]$$

$$\left[\mathrm{O}^{(r)}(U_B,t)\right] = \int [dU^{(r)}]\; e^{-S_r[U^{(r)}|U_B]}\; \mathrm{O}(U^{(r)},t)$$

Requirements:

- S is local
- $\mathrm{O}(U,t_1)\mathrm{O}(U,t_0)$ can be factorised

Multilevel Algorithm: Summary

$$[\mathrm{G}(U_i, t)] = \frac{1}{N_1}\sum_{k}^{N_1} G(U_{ik}, t)$$

When integral can be factorised

When integral cannot be factorised

Standard HMC algorithm

$$\hat{C}(t) = \frac{1}{N}\sum_{i}^{N} \mathrm{O}_i(t)\ \mathrm{O}_i(0)$$

Effectively doing N measurements

with N gauge fields (cost)

$$\sigma(t) \sim \frac{1}{\sqrt{N_0 N_1}}$$

Multilevel HMC algorithm

$$\hat{C}(t) = \frac{1}{N_0}\sum_{i}^{N_0} [\mathrm{O}_i(t)]\ [\mathrm{O}_i(0)]$$

Effectively like doing $N_0 \times N_1^2$ measurements

with $N_0 \times N_1$ gauge fields (cost)

$$\hat{C}(t) = \frac{1}{N_0}\sum_{i}^{N_0}\left[\frac{1}{N_1}\sum_{j}^{N_1}\mathrm{O}_{ij}(t)\right]\left[\frac{1}{N_1}\sum_{j}^{N_1}\mathrm{O}_{ij}(0)\right]$$

$$\sigma(t) \sim \frac{1}{\sqrt{N_0 N_1^2}}$$

Signal/Noise w/ two algorithms

Exponential degradation of signal/noise

Standard algorithm

$$\frac{\hat{C}(t)}{\sigma(t)} \sim e^{-mt} \sqrt{N}$$

$$N \sim e^{2mt}$$

Multilevel algorithm

$$\frac{\hat{C}(t)}{\sigma(t)} \sim e^{-mt} N$$

$$N \sim e^{mt}$$

Exponential improvement in computer time

Less samples N are needed to keep $\text{signal} \gg \text{noise}$.

$N \sim 10^8$ vs $N \sim 10^4$

Successfully employed the Multilevel HMC algorithm in Yang-Mills theory (QCD w/o quarks)

[arXiv:2406.12656] *L. B. (DESY), S. Schaefer (DESY), et al.*

Joint effort of Lattice Community

Several lattice groups coordinate to generate gauge configurations that can be used for different projects:

CLS (coordinated lattice simulations) is one example with DESY operating as one of the main nodes

European level

Some HPC routines are publicly available (e.g. openQCD)

FAIR principles are becoming mandatory requirement to receive European fundings

Findable
Accessible
Interoperable
Reusable

The International Lattice Data Grid (ILDG)

ILDG Goals

- Make sharable, reproducible, usable and citable data for the community
- Ensure basic quality standards
- Provide the framework for the organisation
- Operate at international level (Fermilab, JLAB, DESY, RIKEN, …)

first proposed in 2002

2022 ➡ Start of community efforts to re-activate and modernise ILDG

https://hpc.desy.de/ildg/ ILDG Board: Hubert Simma, and others…

Summary

- Non-perturbative correlation functions contain physical information

- Computing correlation functions with HMC require large number of gauge configurations and HPC

- Correlation functions are high-dimensional objects

Lots of expenses

(money, electricity, CO_2 emission, time)

- Lattice groups join efforts to store and share data and metadata according to **FAIR** principles

- Several HPC routines are publicly available (GitHub, GitLab, …)

- Ongoing work on Multilevel HMC algorithms which require exponentially less resources

Thank you for your attention!

Multilevel Algorithm

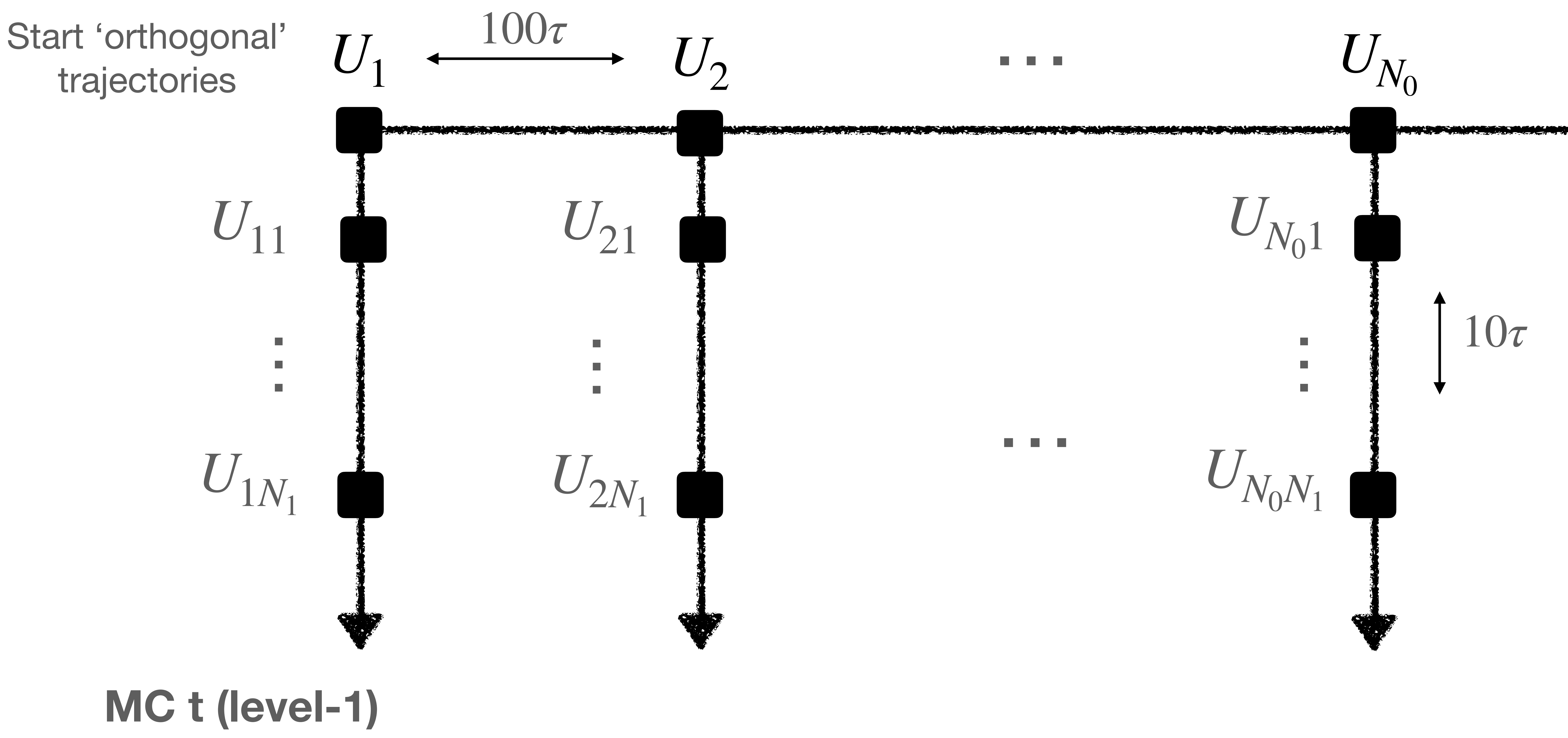
In the same way as with the standard algorithm

we generate N_0 gauge configurations (HMC)

Multilevel Algorithm

$N_0 \times N_1 = 101 \times 1000$

gauge fields



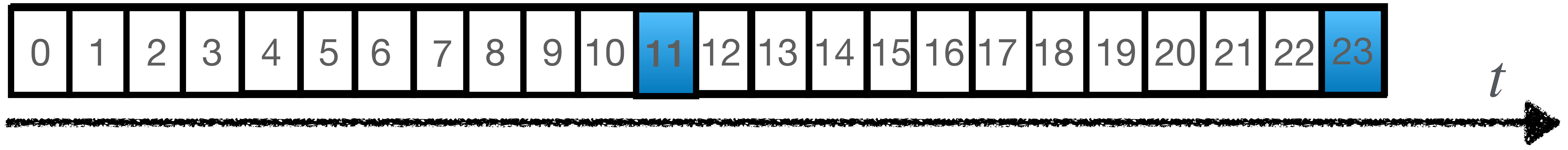
Local updates of gauge fields

gauge fields

boundaries (fixed)

dynamical (updated)

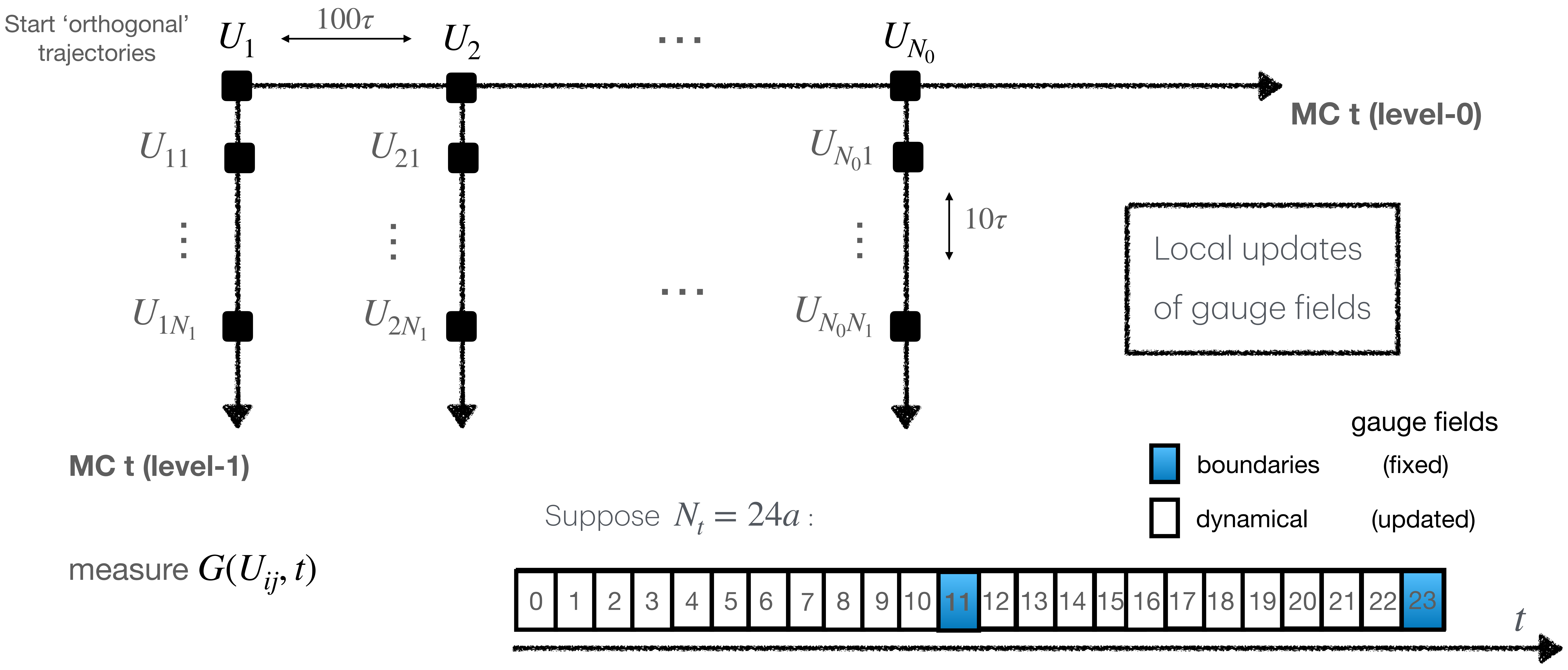
Suppose $N_t = 24a$:



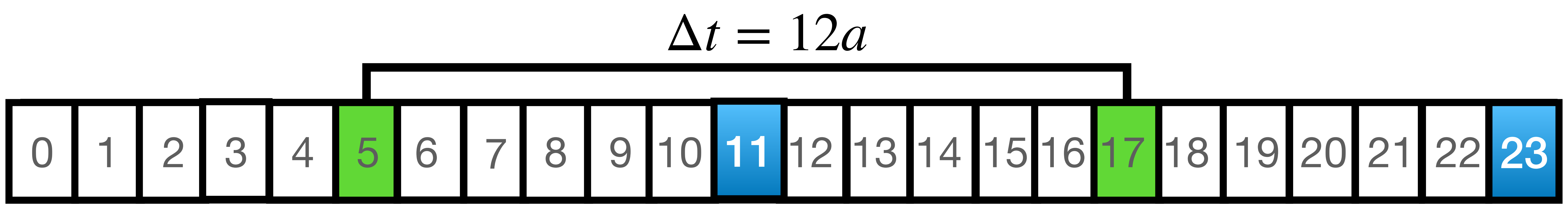
Multilevel Algorithm

$N_0 \times N_1 = 101 \times 1000$

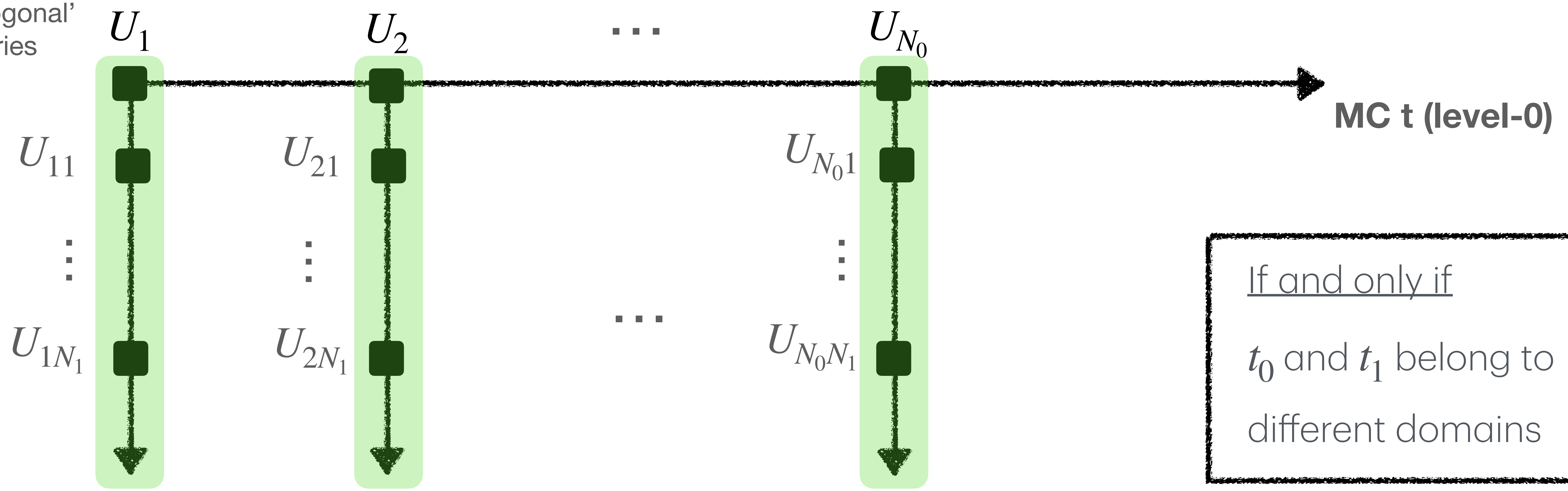
gauge fields



Multilevel Algorithm



Start ‘orthogonal’ trajectories



If and only if

t_0 and t_1 belong to different domains

MC t (level-1)

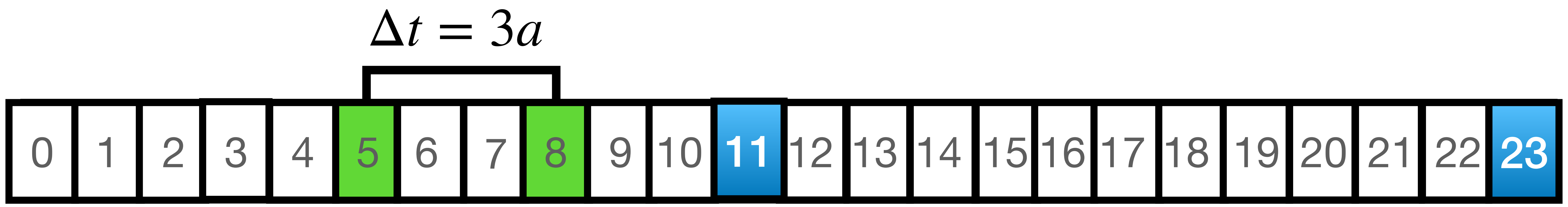
Consider level-1 averages

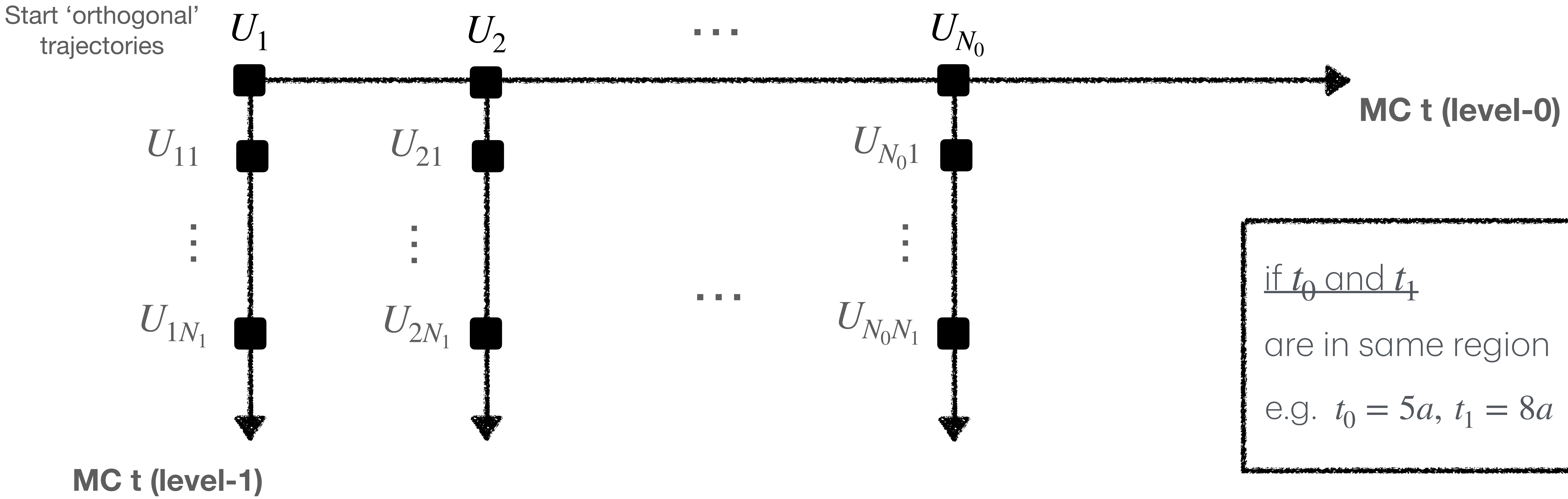
$$[G(U_i, t)] = \frac{1}{N_1} \sum_{j=1}^{N_1} G(U_{ij}, t)$$

and correlate level-1 averages

$$C(t_1 - t_0) = \langle [G(U_i, t_1)]\ [G(U_i, t_0)] \rangle = \frac{1}{N_0} \sum_{i=1}^{N_0} [G(U_i, t_1)]\ [G(U_i, t_0)]$$

Multilevel Algorithm





if t_0 and t_1

are in same region

e.g. $t_0 = 5a,\ t_1 = 8a$

Correlate over $N_0 \times N_1$ samples

$$C(t_1 - t_0) = \langle G(U_{ij}, t_1)\ G(U_{ij}, t_0) \rangle = \frac{1}{N_0} \sum_{i=1}^{N_0} \sum_{j=1}^{N_1} G(U_{ij}, t_1) G(U_{ij}, t_0)$$