FROM QUARKS AND GLUONS TO NUCLEAR FORCES AND STRUCTURE

Lecture 2: Quantum Gauge Fields on a Lattice—A Primer

Thomas Luu

JÜLICH
FORSCHUNGSZENTRUM

UNIVERSITÄT BONN

JUST A RECAP OF THE LAST LECTURE

- Introduced Euclidean path integral and gave expression for expectation value of an operator O

$$\langle E_0|\hat{O}(\hat{x})|E_0\rangle = \frac{\int [d\boldsymbol{x}(t)]\, O(x) e^{-S[\boldsymbol{x}(t)]}}{\int [d\boldsymbol{x}(t)]\, e^{-S[\boldsymbol{x}(t)]}}$$

- Presented an algorithm to generate configurations with probability proportional to the Boltzmann factor

Metropolis-Hastings Algorithm

- Various properties of the system could be probed in the large time limit

WHAT YOU MIGHT HAVE NOTICED…

$$\langle E_0|\hat{O}(\hat{x})|E_0\rangle = \frac{\int [d\boldsymbol{x}(t)]\, O(x)e^{-S[\boldsymbol{x}(t)]}}{\int [d\boldsymbol{x}(t)]\, e^{-S[\boldsymbol{x}(t)]}}$$

$$= \lim_{t_f \gg t_i} \frac{\langle x_f, t_f|O(x)|x_i, t_i\rangle}{\langle x_f, t_f|x_i, t_i\rangle}$$

Independent of $\boldsymbol{x_f}$ and $\boldsymbol{x_i}$?

$$\langle x_f|e^{-H(t_f-t_i)}|x_i\rangle = \langle x_f|U(t_f,t_i)|x_i\rangle \approx \left(\frac{m}{2\pi a}\right)^{N/2} \int_{-\infty}^{\infty} dx[1]\ dx[2]\ dx[3]\ldots dx[N-1]e^{-S_{lat}[x]}$$

IT'S REALLY STATISTICAL PHYSICS

set $x_f = x_i \equiv x[0]$ and integrate over all values of x[0]

PBCs

$$\int [dx] \to \left(\frac{m}{2\pi a}\right)^{(N+1)/2} \int_{-\infty}^{\infty} dx[0]dx[1]dx[2]dx[3]\ldots dx[N-1]$$

$$\begin{aligned}\langle E_0|\hat{O}(\hat{x})|E_0\rangle =& \frac{\int [d\boldsymbol{x}(t)]\, O(x) e^{-S[\boldsymbol{x}(t)]}}{\int [d\boldsymbol{x}(t)]\, e^{-S[\boldsymbol{x}(t)]}} \\ =& \frac{1}{Z} \operatorname{tr}\left[\hat{O} e^{-\beta H}\right] \\ Z =& \operatorname{tr}\left[e^{\beta H}\right]\end{aligned}$$

HOW DO THINGS CHANGE WHEN DEALING WITH A QUANTUM FIELD THEORY?

- Recall that for field theories, degrees of freedom are represented as fields, e.g.

Quantum Mechanics

$$\boldsymbol{x}(t) \mapsto \phi(\boldsymbol{x}, t) = \phi(x)$$

Quantum Field Theory

- Field theories are defined by their Lagrangian—no problem, this is what we've been using all along!

Quantum Mechanics

$$\mathcal{L}[\dot{\boldsymbol{x}}(t), \boldsymbol{x}(t)] \mapsto \mathcal{L}[\dot{\phi}(x), \phi(x)]$$

Quantum Field Theory

LET'S TURN OUR 1-D HO INTO A FIELD THEORY!

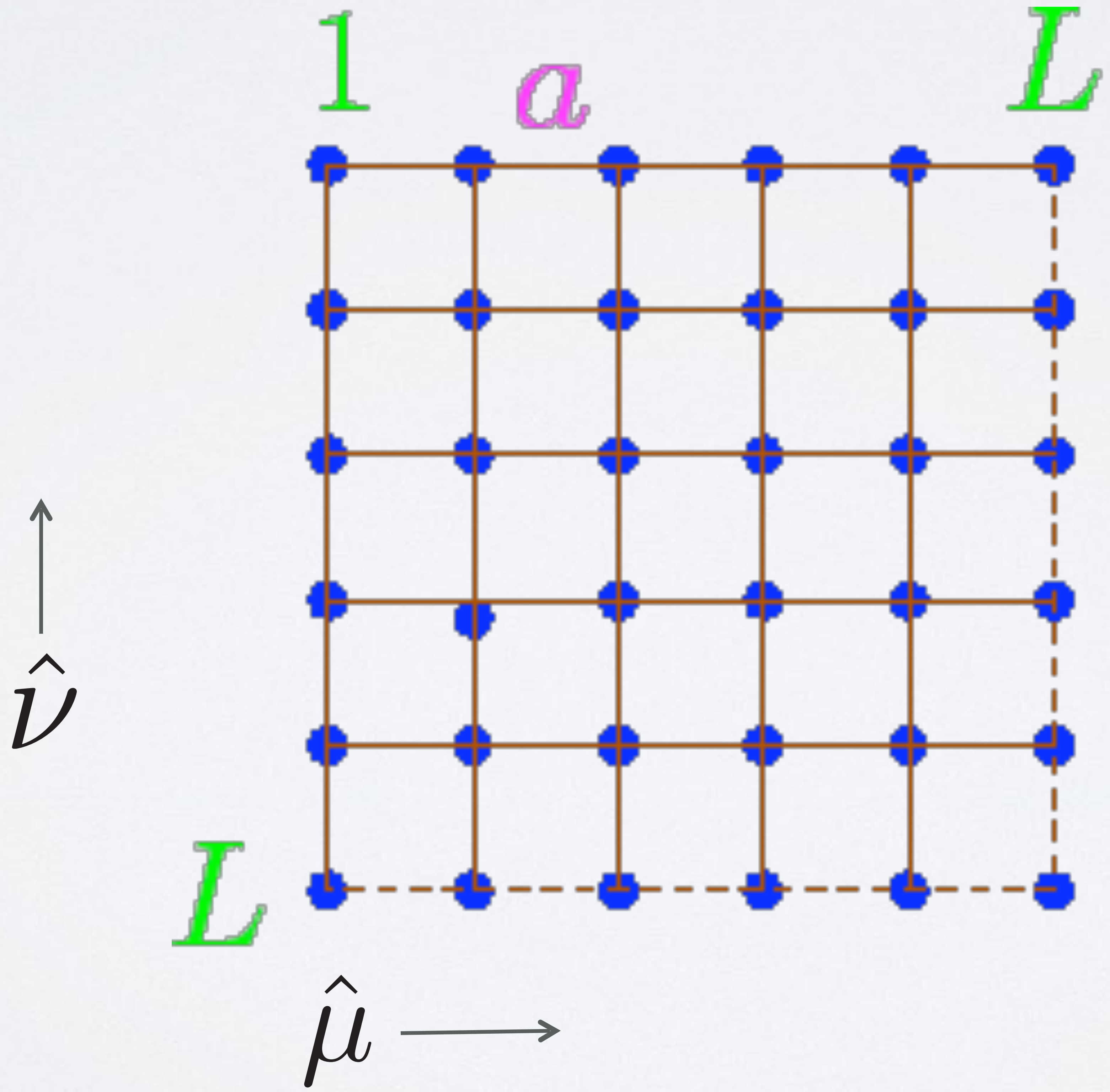
$$\frac{1}{2}\left(m\dot{x}^2 + m\omega^2 x^2\right) + \lambda m^2\omega^3 x^4$$

$$\mapsto \frac{1}{2}\dot{\phi}(x)^2 + \frac{1}{2}(\nabla\phi(x))^2 + \frac{m^2}{2}\phi(x)^2 + \lambda\phi(x)^4$$

phi-4 theory (Euclidean space)

AS BEFORE, WE DISCRETIZE THE COORDINATES

- Not only do we discretize the time coordinate, we discretize the spatial coordinates as well since the fields also depend on them



- Under this discretization, for example, we can approximate the Laplacean in the following manner:

$$\sum_{\mu=1}^{4} (\partial_\mu)^2 \phi(x) \approx \frac{1}{a^2} \sum_{\mu=1}^{4} (\phi(x+\hat{\mu}a) + \phi(x-\hat{\mu}a) - 2\phi(x))$$

- The action, defined as

$$S = \int d^4x \; \mathcal{L}[\dot{\phi}(x), \phi(x)]$$

can be approximated by discrete sums and differences over the field $\Phi(x)$ at the discrete lattice points

THERE ARE SOME FUNDAMENTAL DIFFERENCES, HOWEVER

- Generally, we are interested in calculating expectation values of various operators with respect to the ground state, but here the ground state of a field theory is simply the vacuum,

Quantum Mechanics

$$|E_0\rangle \mapsto \underset{\uparrow}{|0\rangle}$$

Quantum Field Theory

"vacuum state"

- What are the quantized excitations of a field theory?

Answer—The excitations of a field correspond to particles

THE PATH INTEGRAL FORMALISM IS AN INDISPENSIBLE TOOL FOR QF THEORISTS

- Expectation values are just as you would expect:

$$\langle 0|\hat{O}[\phi]|0\rangle \equiv \langle \hat{O}[\phi]\rangle = \frac{\int [d\phi]\, O[\phi] e^{-S[\phi]}}{\int [d\phi]\, e^{-S[\phi]}}$$

- We can extract information about the spectrum of particles (i.e. excitations) associated with a field by using various operators that act on the vacuum state

$$\langle \hat{\phi}(t')\hat{\phi}(t)\rangle = \frac{\int [d\phi]\, \phi(t')\phi(t) e^{-S[\phi]}}{\int [d\phi]\, e^{-S[\phi]}}$$

$$= \sum_n \mathbb{C}_n e^{-E_n(t'-t)} \xrightarrow{t' \gg t} \mathbb{C}_0 e^{-E_0(t'-t)}$$

"Correlator"

Problem #1 (moderate):

We can also extract information about excited states in our quantum mechanical 1-D oscillator problem from the previous lecture.

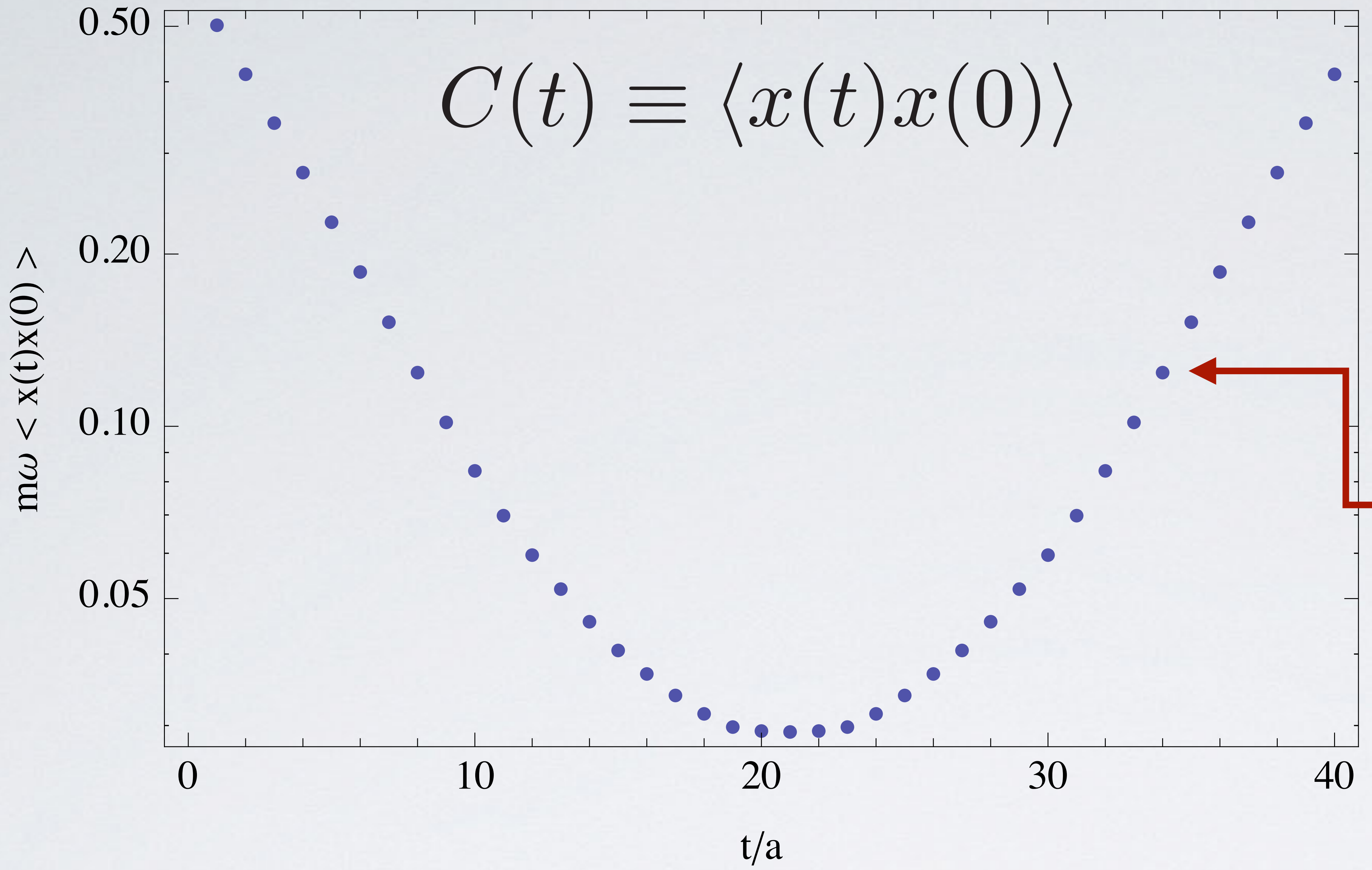
First show that the following relation holds:

$$\langle x(t')x(t)\rangle \xrightarrow{t'\gg t} \mathbb{C}_0 e^{-(E_1-E_0)(t'-t)}$$

What is $\boldsymbol{C_0}$?

Use the ensemble of configurations that you generated from the previous lecture to extract this energy shift. Use periodic boundary conditions in the time direction.

Correlator

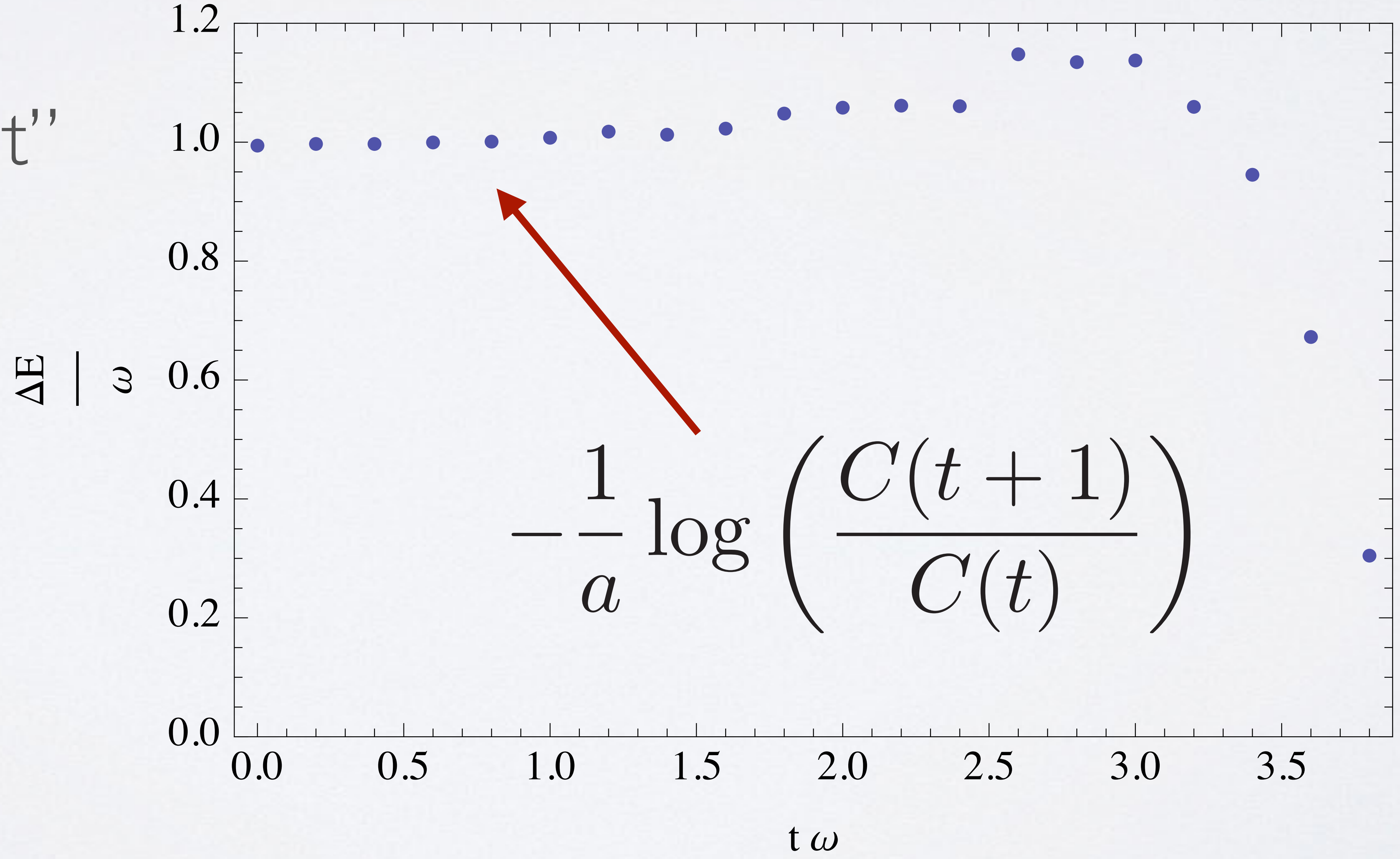


$$C(t) \equiv \langle x(t)x(0) \rangle$$

Periodic boundary conditions

Hey Tom! Where are your uncertainties?

"Effective Mass Plot"



$$-\frac{1}{a}\log\left(\frac{C(t+1)}{C(t)}\right)$$

SOME DISCUSSION POINTS REGARDING QUANTUM FIELD THEORIES ON A LATTICE

- Quantum field theories, from a mathematical point of view, are ill-behaved in the ultra-violet limit—i.e. they suffer from infinities
  - Many people have contributed to "removing" these infinities (e.g. Feynman, Dyson, etc.)
  - Introduce cutoff; express observables in terms of physical parameters, instead of bare parameters; remove cutoff dependence with relevant counter-terms, i.e. renormalization

- By putting QFTs on a discretized lattice, we are already introducing a cutoff: the lattice spacing α

- Removing lattice discretization effects, as well as taking the continuum limit ("renormalization"), is a tricky business!

- QFTs also respect certain symmetries
  - Lorentz invariance
  - Gauge symmetries

- These symmetries are important since conservation laws are derived from them

- How/Can these symmetries be preserved on a lattice?
  - Lorentz invariance? 🚫 But we understand its effects
  - Gauge symmetries? Yes, but does require some extra work—come back to this later

- Fermions obey particular statistics—How is this captured in our numerical path integral?
  - Good question! I will answer this question by avoiding it (for the time being)!

A FIRST LOOK AT QCD—BUT WITHOUT FERMIONS

- The QCD Lagrangian w/o fermions is sometimes referred to as Yang-Mills theory under SU(3):

$$\mathcal{L}_{ym} = \frac{1}{4}\,\mathrm{Tr}\left(F_{\mu\nu}(x)F_{\mu\nu}(x)\right)$$

$$F_{\mu\nu}(x) = \partial_\mu A_\nu(x) - \partial_\nu A_\mu(x) + ig\underbrace{\left[A_\mu, A_\nu\right]}$$

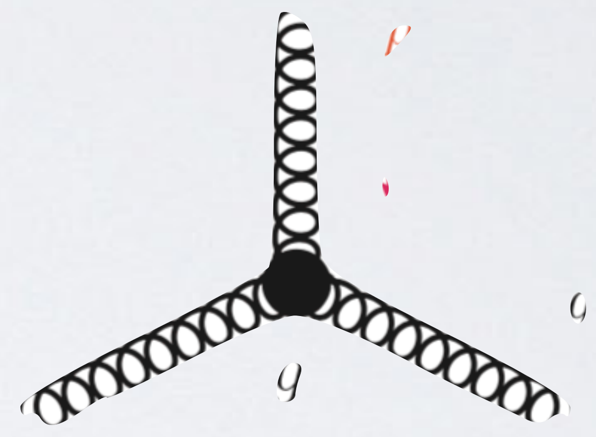


Fig 2

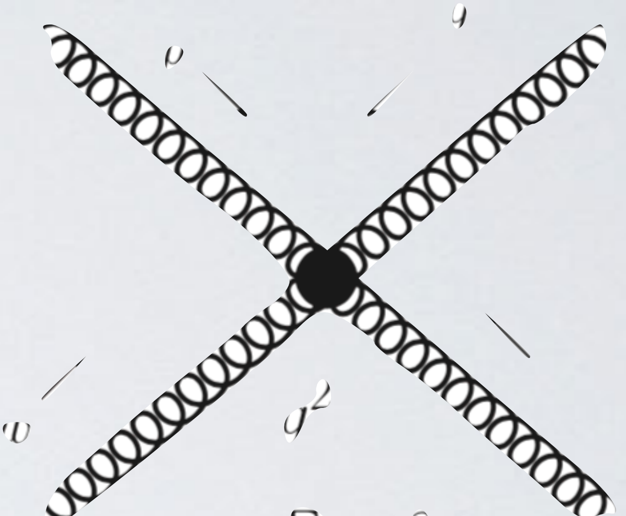


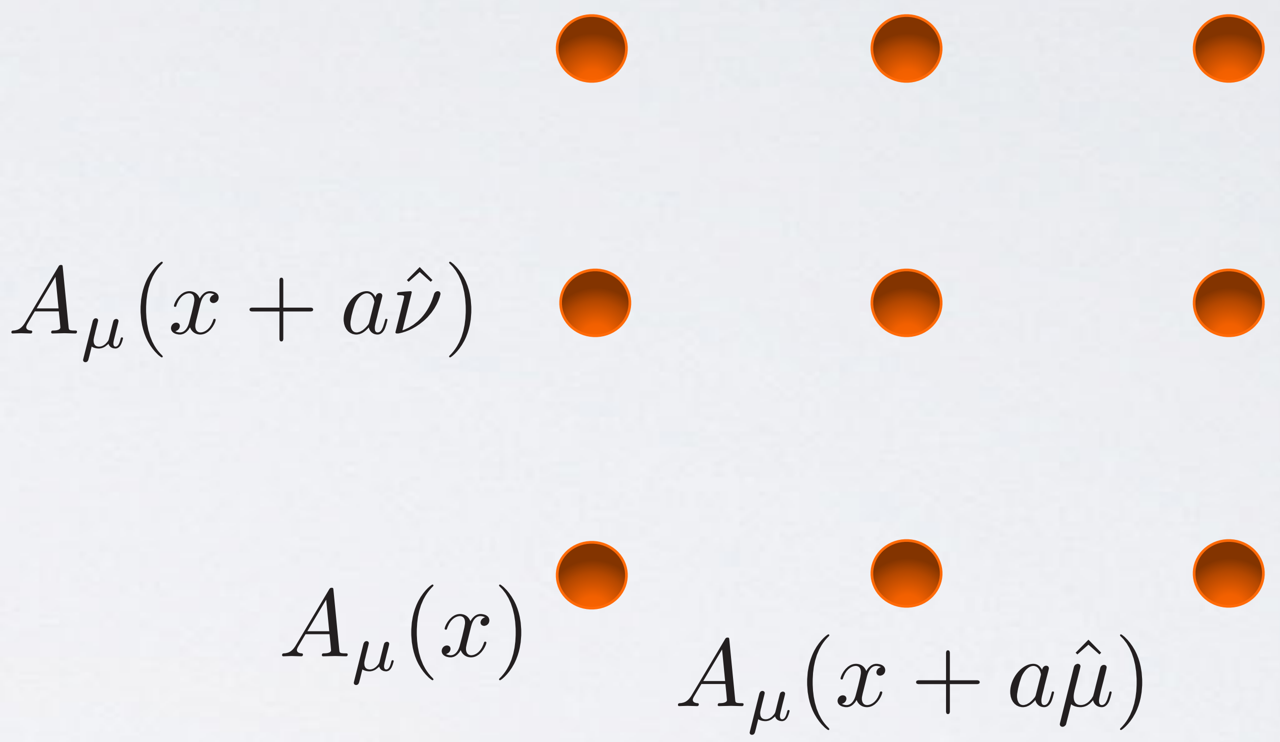
Fig 3

The gauge fields are actually 3x3 Hermitian matrices that are part of the Lie algebra of SU(3)

This commutator is not zero ➔ The gauge fields interact with each other! A consequence of the ***non-Abelian*** nature of SU(3).

SO LET'S DISCRETIZE THIS THEORY

- A naïve first step would be to just define values of the vector gauge field $A_\mu(x)$ at each lattice point



But this is really bad for the following reasons…

- A defining feature of QCD is that it is that the theory is invariant under SU(3) rotations—In other words, the Lagrangian doesn't change when

$$F_{\mu v}(x) \to \Omega(x) F_{\mu v}(x) \Omega^\dagger(x)$$

position-dependent SU(3) matrix

- This symmetry keeps the number of input parameters to QCD to a minimum and conserves color charge
    - From a numerical standpoint, we don't want many parameters in our simulations since they potentially all have to be tuned

- Unfortunately, finite differences and sums of the vector gauge fields **DO NOT** preserve this symmetry under the Lagrangian!

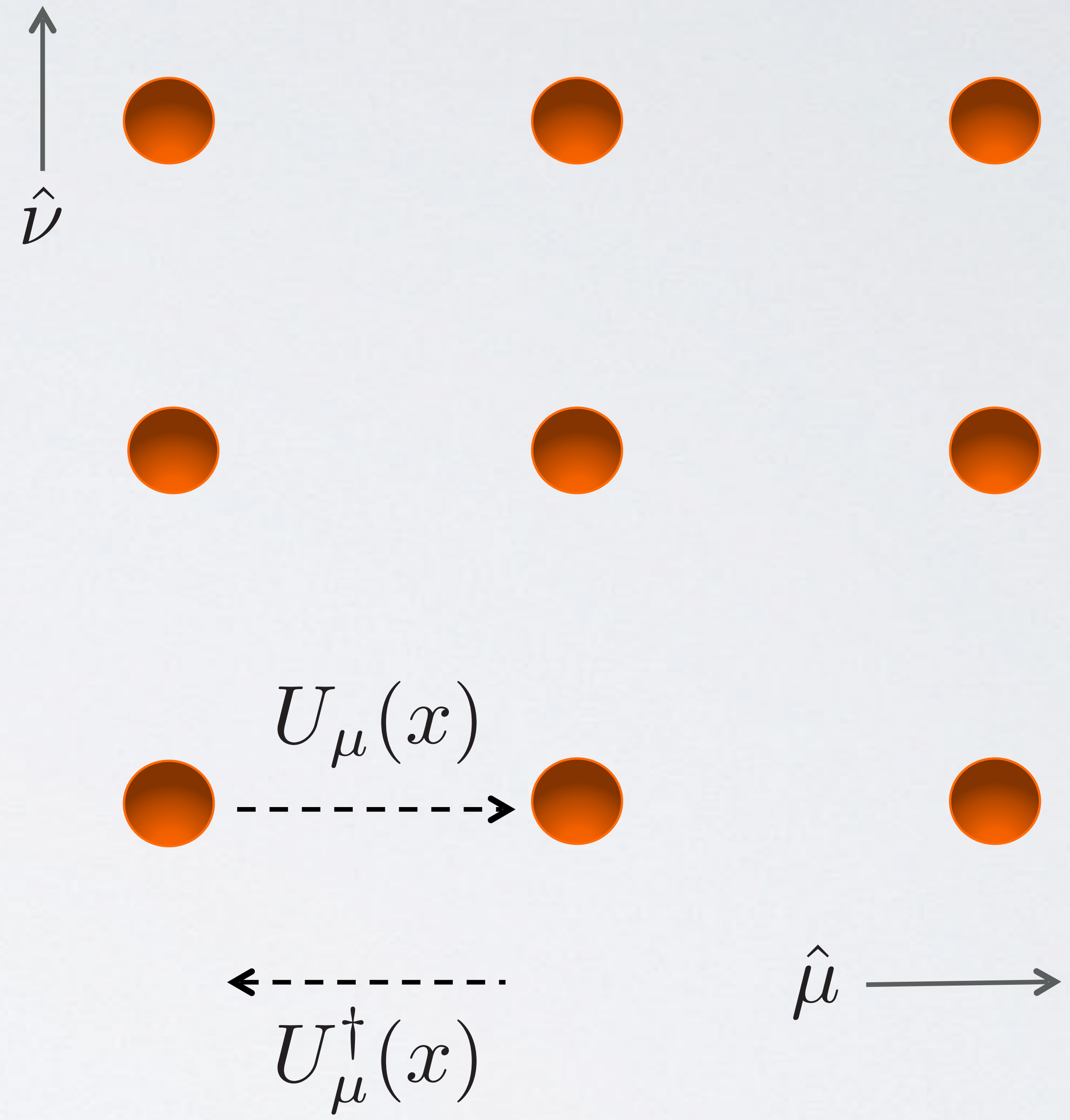
- The upshot: for gauge fields, we need an alternative discretization scheme that preserves gauge symmetry

PRESERVING GAUGE INVARIANCE ON A LATTICE

Kenneth Wilson

- Let us define the following objects:

$$U_\mu(x) = e^{-i\int_x^{x+a\hat{\mu}} gA_\mu(x')dx'} \approx e^{-igaA_\mu(x)}$$



- What are these objects?
  - They are simply 3x3 unitary matrices with det = 1, i.e. SU(3) matrices
  - They depend on position

- They do not reside at lattice points, but between them
- "link variables"

CONSTRUCTING GAUGE INVARIANT OBJECTS USING LINK VARIABLES

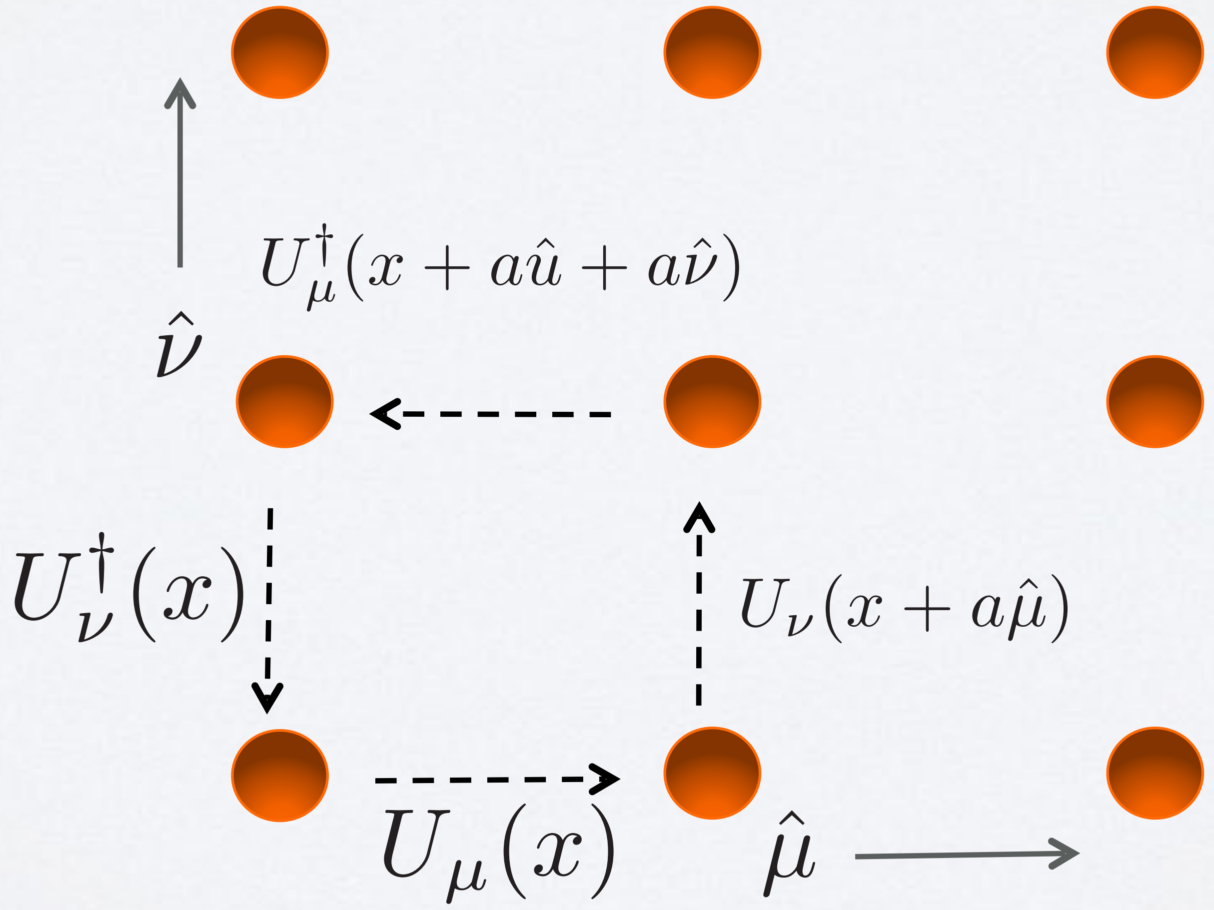
- How do these link variables transform under SU(3) rotations?

$$U_\mu(x) \rightarrow \Omega(x) U_\mu(x) \Omega^\dagger(x + a\hat{u})$$

- The simplest gauge invariant quantity that one can construct is the "plaquette"

$$P_{\mu\nu}(x) = \frac{1}{3} \mathrm{Re}\,\mathrm{Tr}\left[U_\mu(x) U_\nu(x + a\hat{\mu}) U_\mu^\dagger(x + a\hat{u} + a\hat{\nu}) U_\nu^\dagger(x)\right]$$

$$U_\mu(x) = e^{-igaA_\mu(x)}$$



AND FINALLY, THE ACTION…

- It turns out that the original action can be expressed in terms of the plaquettes

$$
\begin{aligned}
S_{ym} &= \int d^4x \; \frac{1}{4} \,\mathrm{Tr}\left[F_{\mu\nu}(x)F_{\mu\nu}(x)\right] \\
&= \frac{6}{g^2} \sum_{x,\mu>\nu} \left(1 - P_{\mu\nu}(x)\right) + O\left(a^2\right)
\end{aligned}
$$

"Wilson action"

Problem 1.5: prove this!

- Since the plaquettes are gauge invariant, our discretized action is gauge invariant!
  - It turns out that correction terms are gauge invariant also!

SO HOW DO WE GENERATE A CONFIGURATION OF THESE GAUGE FIELDS?

Procedure to generate $\{U\}_{n+1}$ from $\{U\}_n$:

```
def updateGaugeFields( U, beta, mu, nT, nX ):
    global num_of_updates, num_of_accepts

    for t in xrange( nT ):  # loop through all gauge links
      for x in xrange( nX ):
         for y in xrange( nX ):
            for z in xrange( nX ):
               for u in xrange( 4 ):
```

Loop through all links j

At link j, change $U(x)$ to $U(x)\ M$, where M is a random SU(3) matrix

Replace $U(x) \rightarrow U(x)\ M$ and compute the change in action ΔS

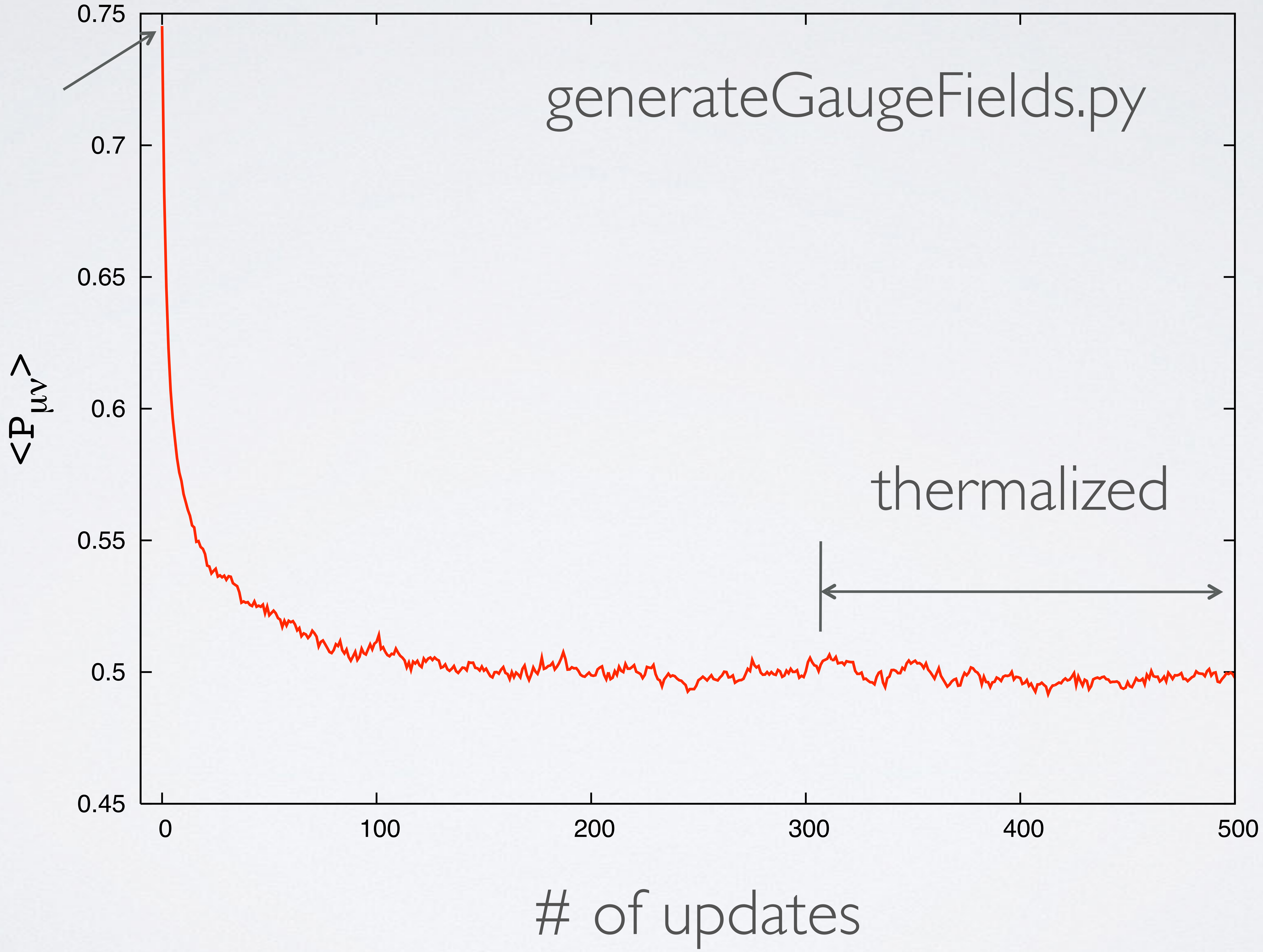
```
                  Gamma_u = actions.calc_Gamma_u( [t,x,y,z], U , u , nT, nX )

                  for j in xrange( 10 ): # at each site update 10 times
                     num_of_updates += 1
                     old_U = U[t][x][y][z][u]
                     U[t][x][y][z][u] = dot( generateSU3_matrix(mu), U[t][x][y][z][u] )

                     dS = actions.deltaS_Wilson( U[t][x][y][z][u], old_U , beta, Gamma_u )

                     if dS > 0 and exp(-dS) < uniform(0,1):
                        U[t][x][y][z][u]  = old_U           # don't accept change
                     else:
                        num_of_accepts += 1      # tally acceptance
                        actions.actionS += dS    # update action
```

If ΔS < 0, accept the new value of $U(x)$and continue to link j+1

If ΔS > 0, sample a number ρ uniformly distributed from 0 to 1. If exp(-ΔS) >ρ accept the new value of $U(x)$, otherwise reject change. Continue to link j+1

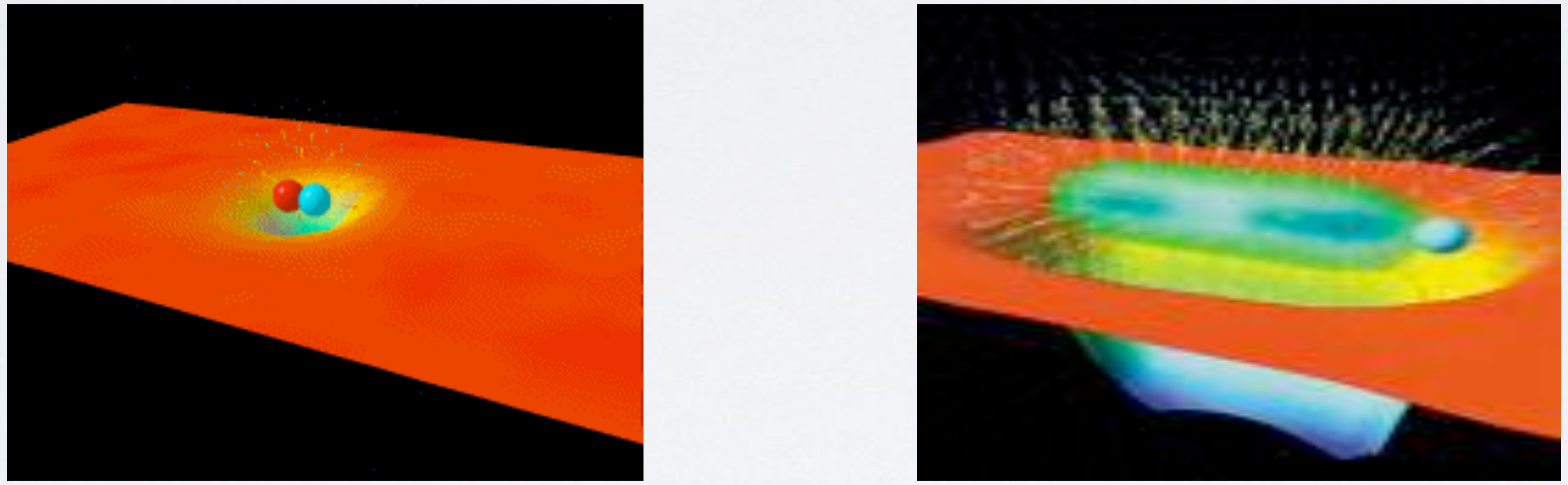
EXAMPLE OF THERMALIZING GAUGE FIELD CONFIGURATIONS

Random
configuration of
SU(3) matrices
“hot start”



CONFINEMENT WITHIN YANG-MILLS THEORY

- One of the great hallmarks of QCD is that it exhibits confinement: there is never a lonely quark in the universe



Indicative of a linear potential at large separations

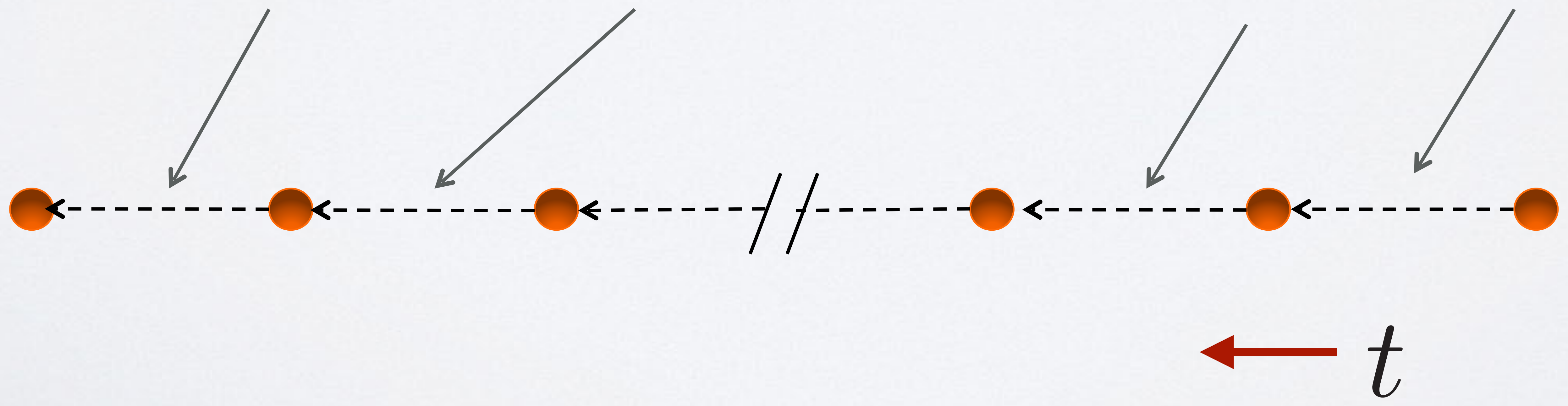
- Our theory of QCD w/o fermions already shows this feature!

HEAVY (INFINITE) QUARK POTENTIALS

- We can insert an infinitely massive quark and anti-quark within our soup of gauge fields and compute the potential between them as a function of separation distance

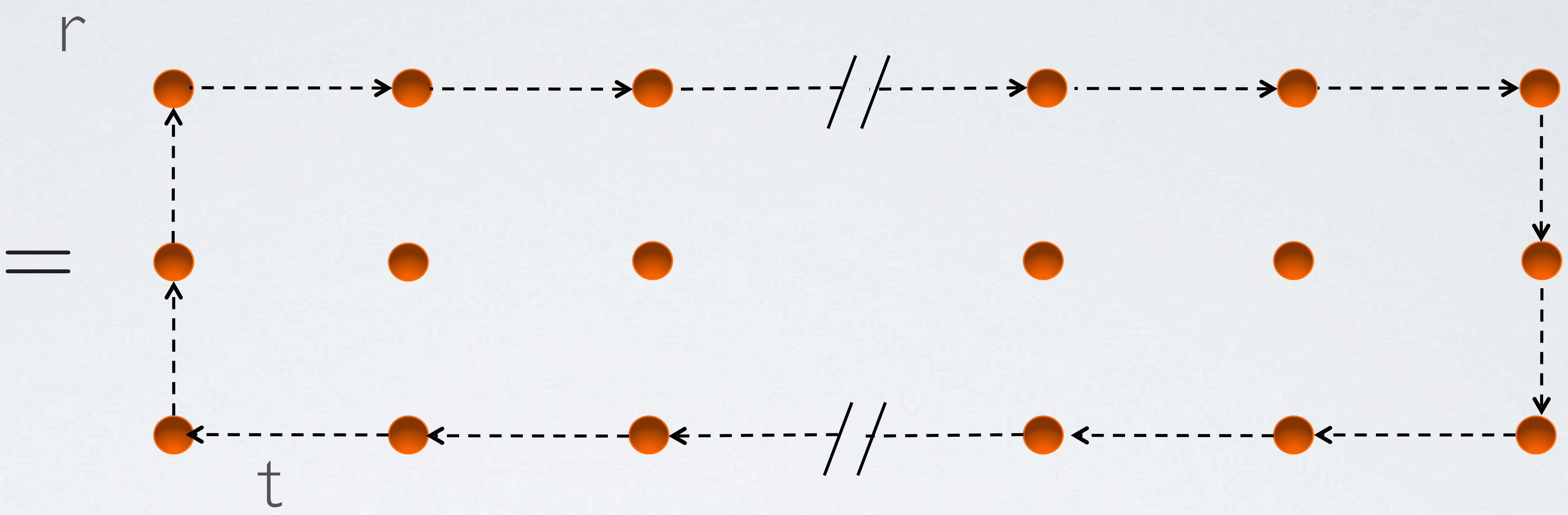
- Because the quarks are infinitely massive, they do not propagate spatially, i.e. they stay put

$$G(x,t) \approx U^\dagger(x,t-a)U^\dagger(x,t-2a)\cdots U^\dagger(x,a)U^\dagger(x,0)$$



- The time-propagation from 0 to T of an infinitely massive quark/anti-quark pair separated by distance R can be ascertained by looking at expectation values of the Wilson loop of dimension $R\,x\,T$

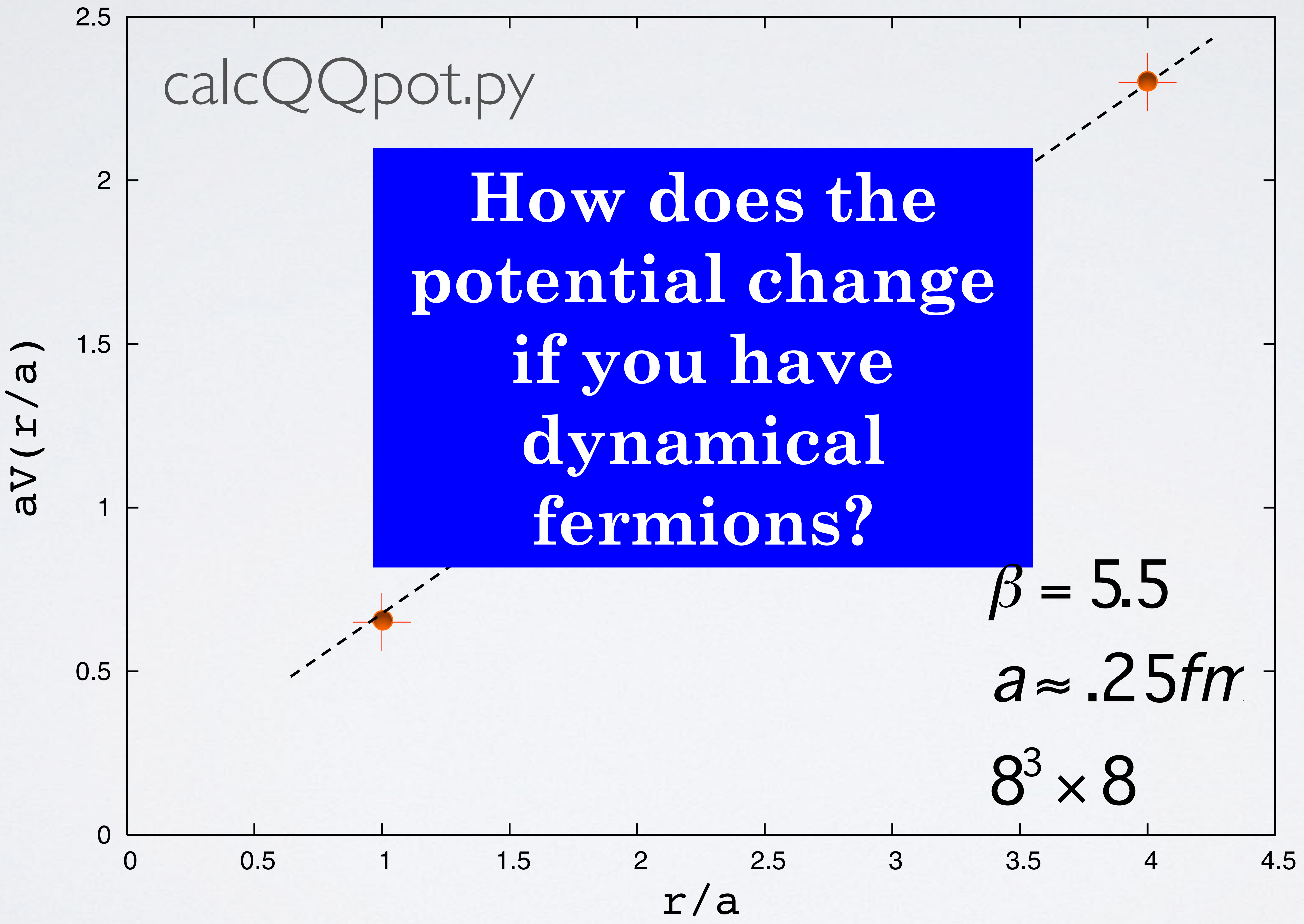
$$W(R = 2a, T) =$$



- The expectation value of the Wilson loop at large times behaves as

$$\langle \mathrm{Re}\,\mathrm{tr}\,W(R,T)\rangle \xrightarrow{T \gg 1} Ce^{-aV(R)T}$$

MY RESULTS FOR THE POTENTIAL



How does the potential change if you have dynamical fermions?

$\beta = 5.5$

$a \approx .25fm$

$8^3 \times 8$

Problem #2 (easy):

Download the python routines that generate gauge ensembles and calculate QQb-potential. Play with the codes to try to make sense of the concepts presented in this lecture.

Some things to note:

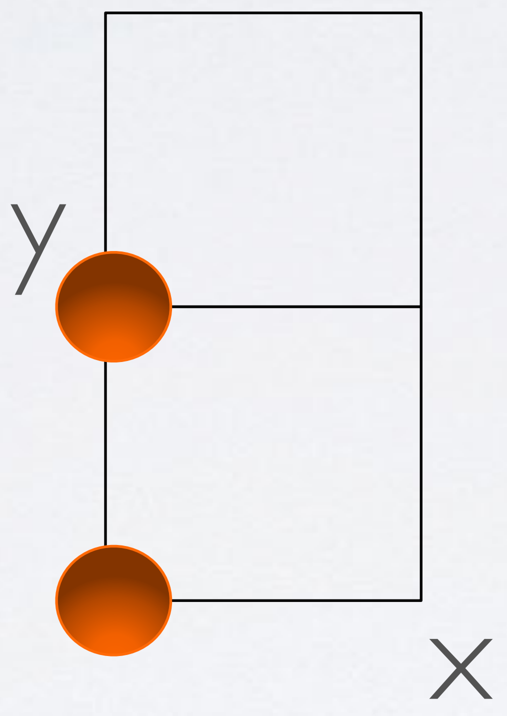
If you run ***generateGaugeFields.py***, the default option is to start from a "hot" (i.e. random) configuration. The configurations will be stored in the folder ***cnfgs***.

The routine ***calcQQpot.py*** needs thermalized configurations to perform measurements.

In the python routines, each configuration is stored as a list of 3x3 arrays: U[x][y][z][t][mu], where mu goes from 0 to 3.

FOR THE MASOCHIST IN YOU…

- The *calcQQpot.py* routine I've given you only calculates separation distances along axis lines, e.g.



- Try modifying the routine to make measurements along diagonals (be careful of boundary conditions!)