PROCEEDINGS OF SCIENCE

Future trends in lattice QCD simulations

Jacob Finkenrath[a,b,*]

[a] *Bergische Universität Wuppertal, Gaussstrasse 20, 42119 Wuppertal, Germany*

[b] *CERN, Esplanade des Particules 1, 1211 Geneva 23, Switzerland*

E-mail: j.finkenrath@cern.ch

This proceeding gives a short overview on current trends for Markov chain Monte Carlo simulations of lattice QCD on supercomputers. State-of-the-art lattice QCD calculations are becoming more and more essential in the search for new physics at the precision (intensity) frontier. Within this proceeding a brief discussion on methods is given for the generation of ensembles at larger lattice sizes and finer lattice spacings as well as methods to reach higher statical accuracy.

EuroPLEx Final Conference 11 - 15 September 2023, Berlin

*Speaker

https://pos.sissa.it/

PoS(EuroPLEx2023)009

1. Introduction

With the entrance of Frontier at Oakridge in the United States of America in Fall 2022 within the Top500 ranking [1], High Performance Computing (HPC) officially entered the Exascale era. With Aurora a second machine recently broke the barrier and soon other machines will follow, like Jupiter at the Jülich Supercomputing Center. This milestone in HPC will enable computations in lattice Quantum Chromodynamics (QCD) at unprecedented high precision and will allow for new scientific discoveries at the foundation of physics. Algorithms, which are suited to perform on the novel HPC architectures, will play here a pivotal role. Within this proceedings, we will take a brief look into a handful of algorithms, which are promising to play a part in this endeavour to shed some light on the unknowns of fundamental physics. Based on recent reviews on algorithms for dynamical fermions and machine learning approaches [11–14], we will give an overview on promising trends in algorithms for lattice QCD.

1.1 Search for new physics

Lattice QCD is a versatile tool in the understanding of the low energy regime of the standard model. This was discussed in several contribution at the workshop [2]. Here, we will use the measurement campaigns for anomalous magnetic moment of the muon as an example to point out the next steps required to reach higher precision. With the new results from the Muon experiment at Fermilab, which improved the error by a factor 2 in August 2023, the mismatch between the data-driven approach [15–18, 20] and the experimental value [21–23], increased to more than 5 standard deviations. To resolve this puzzle, lattice QCD calculations need to decrease the error by at least a factor two. The major contribution comes from the calculation of the leading order (LO) term of the HVP contribution a_μ^{HVP-LO}, which reaches an accuracy of 0.8% in case of the result by BMW [24]. Here, the error splits roughly up into 75% associated to the isosymmetric contribution, which is dominated by uncertainties of the continuum extrapolation and large time distance contributions, 20% associated to finite size effects (FSE) and the remaining part associated to the contribution of isospin breaking effects. To reduce these effects larger and finer lattices (see section 3 and 4) as well as methods for higher precision measurements (see section 5), are needed.

A lot of other EuroPLEX contributions outlined similar needs, as discussed for neutrino-nucleon scattering [4], in thecorrections of isospin breaking effects [6] or in application beyond the electroquenched approximation [7].

2. Machines

With the end of Moore's law complexity in computing architectures is increasing, i.e. now-a-days most of the top HPC systems are equipped with GPU accelerator cards. With Intel, which is equipping the systems Aurora and SuperMUC-NG2, a third vendor after Nvidia and AMD enters the market of cutting-edge GPU devices. Additionally the gap between available FLOPs and bandwidth on- and inter-node is further increasing over the last decade , see Fig. 3 of [14] . While the computational power of a single node is increasing, the strong scaling window of algorithms is shrinking, see Fig. 1, where the coarse grid operators is the bottleneck.

The increase of the complexity of the hardware, e.g. introducing different concepts to address memory and computing layouts, make the optimization of software and algorithms challenging. An example is the newest high-end solution by NVIDIA, Grace-Hopper, a combination of a powerful ARM CPU with several memory lanes connected to an even more powerful CUDA GPU chip. It has to be seen how computational kernels can be designed to utilize such hybrid CPU+GPU in an optimal manner.

Additional the high demand for computational power of large language model has impact on the usage model of supercomputing centers and likely on the design of novel architecture. Namely, if machine learning workloads prefer low precision arithmetics, the importance of high precision arithmetic, which is needed by scientific workloads, might drop. This leads to challenges, which need to be addressed by lattice QCD software solutions and algorithms, e.g. mixed precision solvers are an option to utilize the available hardware [25].

Due to that, ideally, algorithms for novel machines consist of computational kernels with higher algorithmic intensity, i.e. need less bandwidth for computing, and techniques which can avoid communication and use low precision arithmetics.

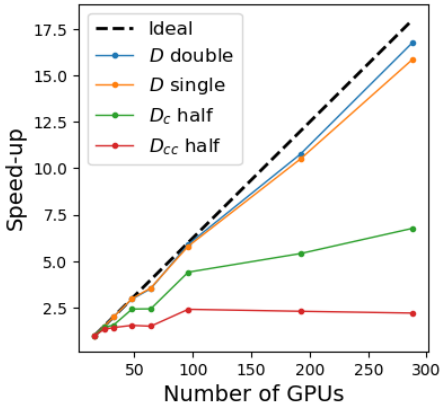

Figure 1: Scaling of components used in a 3 level multi-grid procedure on the A100 GPUs of Juwels-Booster.

3. Algorithms for larger lattices

The state-of-the-art method for simulating large lattices is given by the Hybrid Monte Carlo (HMC) algorithm. Towards large lattices the challenge is given by the sequential nature of the molecular dynamics, which requires continuous updates based on the previous states. This can be only speed up via strong scaling. If the strong scaling window does not scale as well as the computational cost of the HMC, simulation time increases such that ensembles generation requires several years.

The major computational part of the HMC simulation, is solving the Dirac equation $Dx = b$, with $D \in \mathbb{C}^{12V\times 12V}$ a complex matrix which scales with the volume V of the lattice, $b \in \mathbb{C}^{12V}$ the right hand side and $x \in \mathbb{C}^{12V}$ the solution vector. Extending the strong scaling window can be done now via communication avoiding algorithms for linear solvers. A basic approach is given by the Schwarz-Alternating procedure [44]. Here, domain decomposition is used to divide the lattice into domains and introducing a black-white order

$$\det D = \det(1 - D_w^{-1} D_{wb} D_b^{-1} D_{bw}) \prod_j \det D_{b,j} \prod_k \det D_{w,k} \tag{1}$$

with the block operators $D_{b,j}(D_{b,k})$ define on the jth (kth) block. The idea is to apply the inverse of white and black blocks in an alternating procedure. The procedure can be used to preconditioning the iterate in a Krylov solver application via $x_i \rightarrow (1-KD)x_i + KD$ with $K = D_w^{-1} + D_b^{-1} - D_b^{-1} D_{bw} D_w^{-1}$. If we assign at least two blocks per node (or core), there is no (inter-node) communication required

while the inversion of the blocks. If the operator has only next-neighbour interaction, the method is simple to implement and it acts effectively on the large eigenvalues of the operator. Due to that it is used as a smoother in multi-grid methods, see e,g. [28].

The other part of an effective multi-grid procedure, which solves efficiently the Dirac equation, is a correction of the iterate which acts on the low modes. As pointed out in [26] low modes of the Dirac operator can be captured via local coherence and approximated via a coarse grid operator which can be build via suitable projection operators $D_c = RDP$. Now, a flexible GMRES/GCR Krylov method preconditioned with such coarse grid correction is highly effective for fermion discretizations like Wilson clover or twisted mass fermions, outperforming standard methods like the CG solver by more than two magnitudes [27–30]. For other fermion discretization such methods are under development and do not reach similar speed-ups yet [31–34].

Multigrid solvers can be used to significant speed-up HMC simulations. However one needs to take into consideration some points, e.g. the limited scalability. At the large scale the scaling is limited by the coarse grid size while at the lower limit by the memory. Additional it comes with an overhead due to the update of the multi-grid subspace during MD.

To overcome limitations, multiple efforts are ongoing to improve the multigrid performance, e.g. kernel improvements based on multiple right hand sides. This increase the algorithmic intensity, leading to a potential for speed up on GPU architecture roughly by a factor 2 - 4. Such kernel are now available in the software packages QUDA. Additional improvements can be achieved by using communication avoiding Krylov iterative procedures, such as pipelined GCR [46] or communication avoiding CG, see e.g. [36]. Additional coarse grid deflation at physical pion masses have the potential to further speedup the solver. As pointed out in [35], the deflation space can be build up during the solver, via GCR-DR methods. These developments of *DDalphaAMG* are currently ported to a modern C++ layout, to enable prototyping of novel algorithms within the complex multi-grid setup, see for performance results with open boundary condition [37].

3.0.1 Higher order integrators

The HMC algorithm requires for the MD integration a volume preserving, reversible integrator. An collection of higher order schemes mainly based on force gradient integrators can be found in [38] while for hessian free versions in [41].

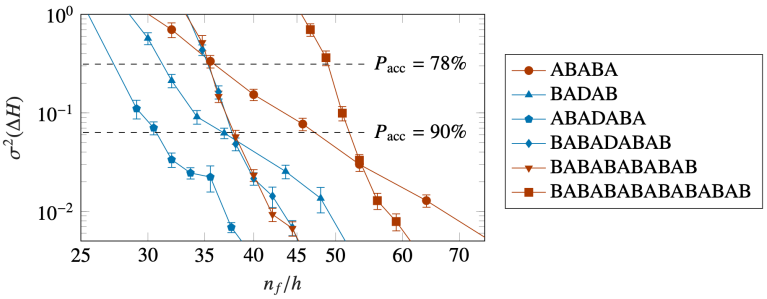

Figure 2: Scaling of different hessian free force gradient integrators with the cost per MDU. Details can be found in [41].

An example is the extension of the second minimal norm scheme, which becomes a fourth order integrator if a force gradient term is added

$$\Delta(h) = e^{h\frac{1}{6}\hat{B}} e^{h\frac{1}{2}\hat{A}} e^{h\frac{2}{3}\hat{B}-172h^3\hat{C}} e^{h\frac{1}{2}\hat{A}} e^{h\frac{1}{6}\hat{B}} \tag{2}$$

where $C = 2\sum_{x=1,\nu=0}^{V,3} \frac{\partial S}{\partial U_\nu(x)} \frac{\partial^2 S}{\partial U_\nu(x)\partial U_\mu(x)}$ [40]. The required calculation of the Hessian can be

express via two applications of the gradient. Namely by following the trick introduced to lattice QCD by Lin and Mawhinney [39] the term can be approximate numerically by

1. Compute a temporary position update via $\mathbf{Q}' = \exp\left(-\frac{2h^2}{m} d^{(n)} \mathbf{e}^j(V)\mathbf{T}_j\right)\mathbf{Q}$
2. Compute a usual momentum update using Q' : $\mathbf{P}_{b^{(n)}h} = \mathbf{P}_0 - b^{(n)} h \mathbf{e}_i(V(\mathbf{Q}'))\mathbf{T}^i$

Note, that this introduces additional higher order terms, which needs to be taken into account for six or even higher order integrators [41]. The method is available in different software suites, recently also in openQCD-2.4 [42].

3.1 HMC on GPUs

In the last decade HMC simulations could be significantly speed up by utilizing multigrid methods by roughly a magnitude and by utilizing computations on the GPU by an additional magnitude [43]. Extrapolating the cost towards a large lattice of size L=192, this gives roughly $\sim$ 1000 Node hours per MDU (4x A100). This seems to be in reach with exascale computing because with current preconditioning techniques, like usage of multigrid solvers and Hasenbusch mass-preconditioning, simulation at physical quark masses are possible. Note that further investigation beyond scalability, like reversibility of the HMC need to be investigated to reach very large lattices. This might require to use quarter precision in parts of the algorithms and might shift the application towards an SME based algorithm [50, 51].

3.2 Additional ways to accelerate HMC-simulations ?

A possible way to utilize the single node performance in the HMC is to exploit the locality of the lattice action. This can be achieved by domain decomposition techniques, pioneered in [44]. Now with the improve node-level performance and larger memory on the GPU large local lattice of size 32^4 are possible. The required operator is currently implemented within the modern software framework *grid* [45]. The general idea is to preconditioning the fermions via domain decomposition:

$$\det D = \det S \cdot \prod_b \det D_b \tag{3}$$

with the Schur complement $S = 1 - D^{-1}_{ww} D_{wb} D^{-1}_{bb} D_{bw}$.

Now the integration of the local parts $\det D_b$ is independent of the global part if the boundary links between the domains, i.e. which are contained in the hopping terms D_{wb} are freezed during the MD integration. By increasing to sufficient large size, this effectively utilizes the single-node performance and avoids communication in the local part dominated by UV fluctuation [45].

Another way of making use of large scale machines is to exploit the parallelism introduced by the rational HMC [47, 48]. Namely, by simply split the fermion determinate into N-pieces, via

$$\det D = \prod^n \det D^{1/n} \tag{4}$$

and distribute each nth root on a different local partitions. Now the inversions require only to exchange force terms during HMC integration. Note that rational approximation can be combined with multigrid solvers, e.g. by using initial guesses the heavy quark sector of ETMCs sector could be speed up by more than a factor 2 [49]. In a similar manner multi-level sampling or multi-tempering techniques can make use of large HPC machines.

4. Algorithms for finer lattices

To reach higher precision it is important to minimize systematic effects arising from the continuum extrapolation. A simple way here is to simulate at finer lattice spacings. This is challenging for HMC simulations due to critical slowing down [52]. At fine lattice spacings the HMC algorithms can not sample efficiently different topological sectors which make simulation below $a < 0.04$ fm on current supercomputers very expensive and in case of periodic boundary condition unfeasible. As investigated by [53] the local fluctuation of the topological charge can be understood by a tunneling term and a diffusion term, with

$$\tau_{\text{tunn}}(Q) = C_{\text{tunn}} \exp(0.9/a) \qquad \text{and} \qquad \tau_{\text{diff}}(Q) = T/8D_{\text{diff}} \tag{5}$$

where the first term describes the spontaneous creation or destruction of a topological charge and the second term describes the *diffusion timescale* through the lattice, see also Fig. 3. If charges can enter the lattice via open boundaries the autocorrelation is dominated by the time the charge diffuses from the boundary to the physical region. This leads to $D_{\text{diff}} \propto O(1)a^2 - O(1)a^4$, which corresponds to $\tau_{int} > O(100)$ for lattice spacings $a < 0.05$ fm [54]. Here, tunneling of topological charges is highly suppressed with $C_{\text{tunn}} \propto O(1)$, see for details [53] .

To overcome this limitations novel algorithms have to improve eq. (5), by minimizing the coefficients or changing the scaling laws. Lets take a look to the general structure of a MC step, which is given by

1. Propose U' according to $T_0(U \to U')$
2. Accept-reject $P_{acc}(U \to U') = \min\left[1, \frac{\tilde{\rho}(U)\rho(U')}{\rho(U)\tilde{\rho}(U')}\right]$.

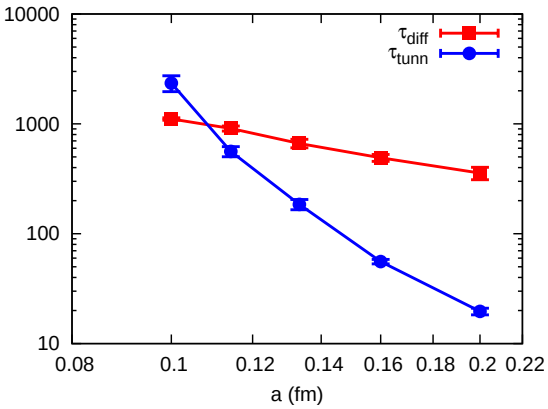

Figure 3: The figure, taken from [53], shows the scaling of the tunneling and diffusion timescales in case of HMC simulation with the DWB2 action.

Obviously the proposal 1.) has to have an improved autocorrelation but for an efficient algorithm, the acceptance rate of the second step should be high. Here, we will discuss two possible approaches which allow tunneling by 1.) independent sampling via gauge flows and by 2.) effectively diffuse fluctuations into the target region via multi tempering.

4.1 Flow

4.1.1 Flows approximating trivial maps

One possibility to allow tunneling between different topological sectors is by propose a new configuration independently of the previous one. This, in principle, can be done via a trivializing map, see [8]. Here we will discuss its application to pure gauge theories, i.e. described by a plaquette action. The idea of a trivializing map is to start with gauge links distributed uniformly with $r(U_0)$ and define a map $f^{-1}(U_0) \to U$, which flows the gauge links to the non-trivial target distribution.

A possible way is to approximate such map via gauge-equivarient coupling layers with a traceable Jacobian, which update a subset of gauge links [57–60]. The flow distribution is given by the Jacobian over the coupling layers $\tilde{\rho}(U) = r(f(U)) \cdot \left|\det \frac{\partial f(U)}{\partial U}\right|$. Now, we can introduce convolutional networks within the layers and train the maps via the Kullbeck-Leibler diversion, such that $\tilde{\rho}(U) \approx \rho(U)$.

Now this approach allows for generating gauge configurations, via direct sampling and accept-reject the proposal via

$$P_{acc} = \min\left[1, \frac{\rho(U')\tilde{\rho}(U)}{\tilde{\rho}(U')\rho(U)}\right] \tag{6}$$

The proposed gauge configurations are independent, because each new one is drawn from an uniform distribution. However, larger autocorrelation are introduced if the corresponding acceptance rate is small. Indeed this is the case for larger volumes, where the acceptance rate breaks down with $\propto e^{-V}$.

Ways to overcome the break down of the acceptance rate is currently under intensive investigation. Note, that there are different ways to set up such flow, a possible alternative is given by continuous flows [61]. Here, a differential equation is used to transform the gaugefield. This has the advantage that symmetries of the theory are exactly preserved, such as translational invariance. However the tuneable parameters are introduced as linear coupling terms, which limits the optimization potential and the introduction of higher loop terms becomes a computational challenge.

In general, as long as the Jacobian of the transformation is tracktable, flows can be also applied to deform the path integral. This can minimize introduced fluctuations when first derivatives or small corrections with respect to observables has to be calculated [65, 66], e.g. of applications of the Feynman-Hellmann method or reweighting.

4.1.2 Localization of flows

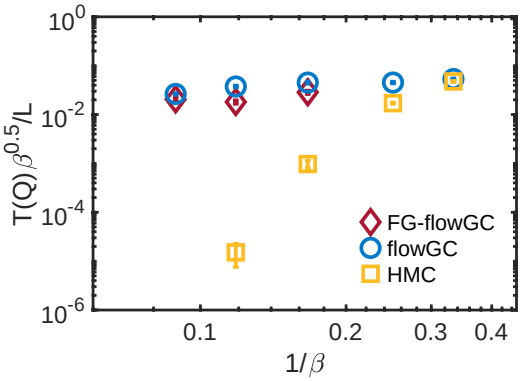

Figure 4: The figure shows the tunneling frequency of the topological charge in the 2D-U(1) model for different MCMC algorithms, see also [67].

A possible way to overcome the breakdown of the acceptance rate $\propto e^{-V}$ is given by utilizing the locality of the gauge model and propose updates within a finite domain. This was successfully tested in the 2D-U(1) model [67], by localizing the flow to a domain. This can be done by fixing the boundary links of a domain and only transform links within the domain. Due to the ultra locality of the gauge action larger volumes can be than trivial generated.

A possible drawback of the flow approaches, is however, that during the flow transformation, the lattice spacing is effectively changed while the number of links is kept constant. This changes the physical box size while intuitively it is more ideal to keep the physical size. How this can be introduced into the flow approaches is under investigation [64] and is based on adding new physical degree while mapping towards the target region. Towards models with higher

dimensions, such as 4D, this becomes hard, because the number of missing links increases drastically. Successful application are given in lower dimensional models applied within renormalization group theory transformation, see [9].

Another way is to effectively fine grain local defects into the target region. The idea is to place a local defect and smooth it out into the surrounding region. This can be done by using flow transformations based on localized and center symmetric coupling layers [68]. Now, one can train such fine graining map by introducing a topological aware training condition. In 2D-U(1) proposals which allows topological transition can be achieved by starting with a defect localized to a single plaquette, i.e. the four links .

Test in 2D U(1) or the Schwinger model, shows that the approach can mild down the behaviour of critical slowing down. The tunneling rate drops only mildly towards fine lattices, while the HMC breaks down, see Fig. 4. Application to 4D are so-far limited to relative small lattice sizes [63], however roughly $L/a > 0.5$ fm should be reached to allow tunneling in 4D-SU(3). While direct sampling of these sizes are not yet in reach, combination with tempering approaches are looking promising. Moreover flows are currently under intensive research, so that advances in the near future are likely.

4.1.3 Localization with global corrections

So far the here discussed flows were only applied to pure gauge theories. In order to utilize them within lattice QCD simulations, the fermion contributions has to be taken into account. This can be done by extending the sampling space by including pseudofermions [69].

Another possible way is to include fermions via correction steps. This is done by the introduction of an additional accept-reject step with the fermion determinant [44, 70]. Obviously, this step suffers from the extensive nature of the determinant, i.e. the acceptance rate drops proportional with $\propto e^{-V}$. This can be overcome by making use of the locality of the action. The determinant can be decomposed via $\det D = \det S \cdot \det D_{b,b} \det D_{r,r}$, where the Schur complement is given by $S = 1 - D^{-1}_{r,r} D_{r,b} D^{-1}_{b,b} D_{b,r}$ where the red and black block operators $D_{r,r}$ and $D_{b,b}$ are defined on domains. Now, one can introduce a hierarchy of filter steps, by first accep-reject local parts followed by a global correction step, which involves the Schur complement [70]. By using correlations acceptance rates of $> 90\%$ can be reached in case of the Schwinger model. The applications to 4D-SU(3) are possible by using stochastic methods for estimating the determinant ratios [70] and requires implementation of domain decomposition techniques, which are being conducted within the software packages QUDA and grid.

4.2 Multi-tempering algorithms

Another algorithmic approach, which potentially can overcome the critical slowing of topological tunneling, is given by multi-tempering [71, 72]. The basic idea is to run several MCMC chains in parallel using different sampling distributions, at least one with a mild scaling $\tilde{\rho}$ and one ρ in the target space from which configuration are sampled. To unfreeze the topology, a swapping step between the distribution is introduced

$$P_{swap}(U_1 \leftrightarrow U_2) = \min\left[1, \frac{\tilde{\rho}(U_1)\rho(U_2)}{\rho(U_1)\tilde{\rho}(U_2)}\right] \tag{7}$$

where U_1 is switched with U_2 if the swap is accepted. Tempering works if the acceptance rate is reasonable high, i.e. if $\tilde{\rho}$ and ρ are similar to each other.

4.3 Multi-tempering algorithms with a meta potential

A possible choice for $\tilde{\rho}$ is by adding a meta-potential to ρ, which allows for topological transitions [73]

$$V_t(s) = \sum_{t \geq t} \prod_{i=1}^{N} g(s_i - s_i(t')) \tag{8}$$

with Gaussian $g(s_i) = \omega \exp\left(-\frac{s_i^2}{2\delta s_i^2}\right)$ where for s an approximation to the topological charge is used. Note, the build up of the meta-potential comes with an larger computational overheads and requires to generate a rather long MCMC chain. However the corresponding costs can be likely minimized by educated guesses.

By adding only the meta-potential, the swapping probability reduces to

$$P_{swap} = \min\left[1, e^{-V_t(U_{meta})+V_t(U_0)}\right]. \tag{9}$$

The effectiveness of the approach is studied in SU(3) 4D pure gauge theory and tested on lattices with spatial extend of $L = 22$ (see Fig. 5). Note this approach is significant improved to the previous used reweighting approach. The topological transition occurs more frequent but the decoupling time is still sizeable. Additional the volume scaling of the approach is still under investigation as well as its application to dynamical fermion simulations.

4.3.1 Multi-tempering sampling with a defect

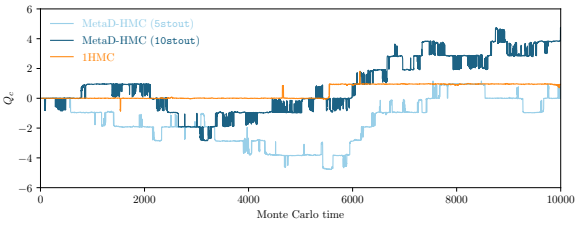

Figure 5: The figure, taken from [73], shows the evolution of the topological charge using the HMC with and without meta potentials.

Another possible choice for $\tilde{\rho}$ is given by utilizing the a^{-2} scaling of the topological diffusion. This can be done by *opening* the lattice via a defects in the lattice. Effectively the gauge coupling β of the pure gauge action within a region, for example in a 3D cube of size l=2, is changed . Namely for $\tilde{\rho}$ the gauge coupling β of the defect is set to zero and interpolate towards the target chain with ρ back to the target β. The swapping step is given by $P_{swap}(l \leftrightarrow (l-1)) = \min\left[1, \frac{\exp(-S_{l-1}(U^{(l)}) - S_l(U^{(l)}))}{\exp(-S_l(U^{(l-1)}) - S_{l-1}(U^{(l-1)}))}\right]$. As demonstrated in [75], this works very well for SU(N) theories. Here 16 levels between target and defect needs to be introduced to reach a swapping rate of 20%. Note due to swapping of the chains the autocorrelation function becomes rather non-trivial, e.g. longer autocorrelation modes are potentially hidden due to frequent chain swaps.

As shown recently [77], due to the localization of the defect, it is possible to include fermions, i.e. using staggered fermions at physical pion masses. By only include the defect within the gauge action, the fermion weight drops out of the swapping step and decent swapping rates of 20% were reached by introducing 10 chains. This leads to unfreezing of the topological charge however by increasing the computational cost proportional with the number of tempering levels. So far

the approach is limited to one defect, i.e. scaling towards larger lattices might be not optimal and further investigations are needed to understand the scaling of the diffusion-tunneling effects. Overall the approach allows topological sampling of lattices without the need of open boundaries even at physical pion masses and is a very promising step towards enabling simulations at very fine lattice spacings.

4.3.2 Multi-tempering with flows

To reduce the cost of multi-tempering, i.e. the number of tempering levels, a combination with gauge flows are promising. Namely, the standard swapping steps do not act on the gauge links but only changes the probability distributions. Introducing the gauge flow within the swapping steps, enables to act on the gauge links and increase so the overlap towards the proposed distributions. If the Jacobian of the transformation is tracktable, for the swapping step follows

$$P_{swap}(l \leftrightarrow (l-1)) = \min\left[1, \frac{p_{l-1}(U'^{(l)})\, p_l(U'^{(l-1)})}{p_l(U'^{(l)})\, p_{l-1}(U^{(l-1)})} J_f(U^{(l-1)}) J_{f^{-1}}(U^{(l)})\right] . \tag{10}$$

For the gauge flow, one can utilize the techniques used for the trivializing maps and train neural networks in order to achieve high swapping rate. As shown in [62], this effectively minimizes the number of required level, i.e. from O(10) to O(1). Further investigations are required to make use of the full potential, i.e. optimal size of the defect, training procedure or inclusion of fermions.

5. Algorithms for higher statistics

Another major challenge is to measure observables to a high precision. This is often notorious difficult due to an exponential growing of the noise relative to the signal, e.g. in case of the nucleon $\propto \exp\{-(m_N - 3/2m_\pi)t\}$. This allows only to extract quantities, such as the effective mass, within a small time-window, i.e. examples are discussed in [4, 5, 79].

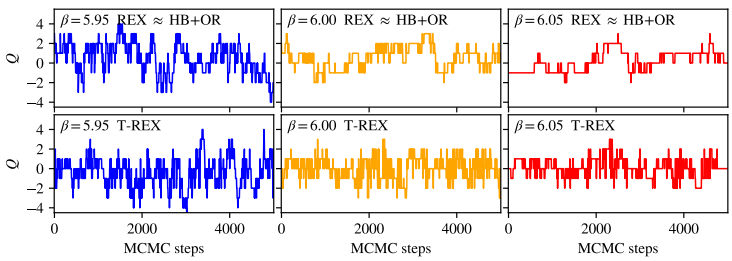

Figure 6: The figure, taken from [62], shows the comparison between the application of the T-REX algorithm compare to the MCMC chains without tempering.

Method which acts on the width are improving the achievable precision. At the lower bound, for short time distances, excited states contaminates the signal and the region to extract the quantity of interest starts at larger time distance. A way to minimise the contamination is to resolve the excited states, e.g. by extending the correlation matrix with appropriate operators. A method, which proved to be effective here, is to utilize distillation by introducing optimised profile to the basis [78], see also [10].

The signal at large time distance is usually dominated by noise and its onset limits the region. Due to the statistical nature, it can be suppressed via increasing the number of measurements. Obviously, due to the exponential gap between noise and signal, simply increasing statistic is rather limited.

5.1 Multi-level sampling

A promising methods to further minimize this gap, is given by using multi-level sampling. The general idea, is to sample domains of the lattice independently from each other and recombine them in the measurements later. If the decomposition is effectively utilizing the locality of the model, e.g. by freezing links between the active domains, the statistical error can be minimized with $N^{-n_d/2}$ where n_d is the number of measurements.

In pure gauge theories, multi-level techniques were successfully applied [80–83]. Recently, the effectiveness were studied in case of glueball measurements [84]. Here, it was sufficient to freeze the links within one time slice to observe a reduction of the error. In detail, the reduction effect depends on the lowest mode in the channel and the distance to the frozen region [85].

Multi-level sampling can be also applied to models with fermions. This, however, is more challenging due to character of the fermions. Using domain decomposition technique, it is also possible to effectively decouple the fermion parts. If the local parts are dominating the signal a reduction effect similar to the pure gauge model can be achieved [86, 87]. One main difference is that the lowest modes in models with fermions are naturally an order of magnitude smaller than in pure gauge models, which makes multi-level sampling less effective. Additional the fermion decomposition requires a larger frozen region, i.e. the distance between active regions needs to be sufficient large that they can effectively be sampled independently. For example in case of 300 MeV Pions the frozen region requires a distances of ~ 0.5 to 0.8 fm, as investigated in [86, 87]. For physical quark masses, this distance becomes larger ~ 1.5 to 2.5 fm. In general this is exactly the distance where statistical noise starts to dominantes the signal, i.e. effective mass plateaus of baryons becomes very noisy for $t > 1.5$ fm similar to the long distance signal for the HVP [88], which makes further investigation in this direction attractive. However multi-level sampling at physical pion masses are computational challenging and requires effective software solution, i.e. an open source software version is currently missing. Additional multi-level sampling is very well suited for parallel computing, i.e. generation of the local domains and calculation on the local domains can be done independently from each other and can be used in masterfield-like applications.

6. Conclusion

To conclude the discussion on future trends, a well suited algorithm might be given by a method based on multi-level sampling accelerated with local updates, which uses 4D Domain decomposition in combination with localized updates which enables topological transitions. Such methods is promising to enable simulation at very fine lattice spacing and is meeting hardware requirements at the same time. For example multiple chains can be run in parallel (also very suitable for multiple tempering), communication overheads can be avoided and localization can be used to increase acceptance and statistics.

The development of such methods are very challenging. New software solutions are needed and require major development efforts. Currently, there are no open source codes for multi-level sampling as well as SU(N) flows available especially in combination with flexible kernels for novel GPU architectures. However implementations of such solutions are currently ongoing. An example is the software package *grid* [89, 90], where methods to enable Domain Decomposition

HMC on GPUs are currently implemented. Similar efforts exist for the package QUDA, e.g. via implementaion efforts like within an innovation study [91]. These methods will be handy when the European Exascale machines, Jupiter and Alice Recoque, are coming online and will help to further push the precision frontier in elementary particle physics.

Acknowledgments:

The author would like to thank the organisers for hosting a productive and splendid workshop. The author received financial support by the German Research Foundation (DFG) research unit FOR5269 "Future methods for studying confined gluons in QCD" and is supported by Inno4scale under grant agreement No 101118139. Additional the author want to thank for helpful discussions with S. Bacchio, C. Bonanno, T. Eichhorn and J. Urban.

References

[1] Top 500 List, https://www.top500.org/

[2] EuroPLeX Final Conference, Berlin, September 2023 https://europlex.unipr.it/network-events/europlex-final-conference-2023/

[3] T. Blum, Status of hadronic contributions to the muon anomalous magnetic moment from lattice QCD, EuroPLeX Final Conference, Tuesday, 9:45

[4] R. Gupta, Lattice QCD input for neutrino-nucleus scattering, PoS(EuroPLEx2023)010

[5] L. Barca, G. Bali and S. Collins, PoS(EuroPLEx2023)002, [arXiv:2405.20875 [hep-lat]].

[6] M. Di Carlo, Isospin breaking corrections to QCD observables, EuroPLeX Final Conference , Tuesday 11:00.

[7] T. Harris, Beyond the electroquenched approximation, PoS(EuroPLEx2023)011

[8] J. Urban, Properties and uses of approximate trivializing maps in lattice QCD, EuroPLeX Final Conference, Thursday, 9:00.

[9] D. Bachtis, PoS(EuroPLEx2023)001, [arXiv:2405.16288 [hep-lat]].

[10] F. Erben, Resonances from LQCD via Distillation, EuroPLeX Final Conference, Friday, 9:45.

[11] G. Kanwar, [arXiv:2401.01297 [hep-lat]].

[12] P. A. Boyle, [arXiv:2401.16620 [hep-lat]].

[13] J. Finkenrath, PoS **LATTICE2022** (2023), 227 [arXiv:2402.11704 [hep-lat]].

[14] P. Boyle, D. Bollweg, R. Brower, *et al.*[arXiv:2204.00039 [hep-lat]].

[15] T. Aoyama, N. Asmussen, M. Benayoun, J. Bijnens, *et al.*Phys. Rept. **887** (2020), 1-166

[16] M. Davier, *et al.* Eur. Phys. J. C **80** (2020) no.3, 241 [erratum: Eur. Phys. J. C **80** (2020) no.5, 410]

[17] A. Keshavarzi, *et al.* Phys. Rev. D **101** (2020) no.1, 014029

[18] M. Hoferichter, B. L. Hoid and B. Kubis, JHEP **08** (2019), 137

[19] G. Colangelo, M. Hoferichter and P. Stoffer, JHEP **02** (2019), 006

[20] G. Colangelo, M. Hoferichter and P. Stoffer, JHEP **02** (2019), 006

[21] Bennett, G. W. and others Phys. Rev. D 73 (2006) 072003

[22] Abi, B. and others, Phys. Rev. Lett. 14,(2021) 141801

[23] Aguillard, D. P. and others, Phys. Rev. Lett. 113 (2023) 16

[24] S. Borsanyi, *et al.* Nature **593** (2021) no.7857, 51-55

[25] M. A. Clark, D. Howarth, J. Tu, *et al.*PoS **LATTICE2022** (2023), 338

[26] M. Luscher, JHEP **07** (2007), 081

[27] R. Babich, J. Brannick, R. C. Brower, *et al.* Phys. Rev. Lett. **105** (2010), 201602

[28] A. Frommer, K. Kahl, S. Krieg, *et al.* SIAM J. Sci. Comput. **36** (2014), A1581-A1608

[29] C. Alexandrou, S. Bacchio, J. Finkenrath, *et al.* Phys. Rev. D **94** (2016) no.11, 114509

[30] M. A. Clark *et al.* [QUDA], doi:10.5555/3014904.3014995 [arXiv:1612.07873 [hep-lat]].

[31] R. C. Brower, M. A. Clark, *et al.* Phys. Rev. D **97** (2018) no.11, 114513

[32] V. Ayyar, R. Brower, M. A. Clark, et al.,[arXiv:2212.12559 [hep-lat]].

[33] R. C. Brower, M. A. Clark, D. Howarth *et al.* Phys. Rev. D **102** (2020) no.9, 094517

[34] P. Boyle and A. Yamaguchi, [arXiv:2103.05034 [hep-lat]].

[35] J. Espinoza-Valverde, A. Frommer, *et al.*Comput. Phys. Commun. **292** (2023), 108869

[36] M. Hoemmen, Communication-Avoiding Krylov Subspace Methods, U. of California Berkeley

[37] A. Frommer,*et al.* DDalphaAMG Solver for Lattice QCD on GPUs, in preparation

[38] Omelyan, Mryglod, and Folk, Phys. Rev. Lett. 86(5), 898. (2001).

[39] H. Yin and R. D. Mawhinney,

[40] A. D. Kennedy, P. J. Silva and M. A. Clark, Phys. Rev. D **87** (2013) no.3, 034511

[41] K. Schäfers, J. Finkenrath, M. Günther and F. Knechtli, [arXiv:2403.10370 [math.NA]].

[42] K. Schaefer, J. Finkenrath, https://github.com/KevinSchaefers/openQCD_force-gradient

[43] B. Kostrzewa *et al.* [ETM], PoS **LATTICE2022** (2023), 340

[44] M. Luscher, Comput. Phys. Commun. **165** (2005), 199-220

[45] P. A. Boyle, D. Bollweg, C. Kelly and A. Yamaguchi, PoS **LATTICE2021** (2022), 470

[46] P. Ghysels, T. Ashby, K. Meerbergen, W. Vanroose, J. Sci. Comput. 35 (1) (2013) C48–C71.

[47] M. A. Clark and A. D. Kennedy, Phys. Rev. Lett. **98** (2007), 051601

[48] P. de Forcrand and L. Keegan, Phys. Rev. E **98** (2018) no.4, 043306

[49] C. Alexandrou, S. Bacchio and J. Finkenrath, Comput. Phys. Commun. **236** (2019), 51-64

[50] R. G. Edwards, I. Horvath and A. D. Kennedy, Nucl. Phys. B **484** (1997), 375-402

[51] A. Francis, P. Fritzsch, M. Lüscher and A. Rago, Comput. Phys. Commun. **255** (2020), 107355

[52] S. Schaefer *et al.* [ALPHA], Nucl. Phys. B **845** (2011), 93-119 [arXiv:1009.5228 [hep-lat]].

[53] G. McGlynn and R. D. Mawhinney, Phys. Rev. D **90** (2014) no.7, 074502

[54] A. Athenodorou *et al.* [ALPHA], Nucl. Phys. B **943** (2019), 114612

[55] M. Luscher, Commun. Math. Phys. **293** (2010), 899-919

[56] G. P. Engel and S. Schaefer, Comput. Phys. Commun. **182** (2011), 2107-2114

[57] M. S. Albergo, G. Kanwar and P. E. Shanahan, Phys. Rev. D **100** (2019) no.3, 034515

[58] G. Kanwar, M. S. Albergo, D. Boyda, *et al.*

[59] D. Boyda, G. Kanwar, *et al.* Phys. Rev. D **103** (2021) no.7, 074504

[60] M. S. Albergo, D. Boyda, *et al.* [arXiv:2101.08176 [hep-lat]].

[61] S. Bacchio, P. Kessel, S. Schaefer and L. Vaitl, Phys. Rev. D **107** (2023) no.5, L051504

[62] R. Abbott, D. Boyda, *et al.* PoS **LATTICE2023** (2024), 011

[63] R. Abbott, M. S. Albergo, *et al.* Eur. Phys. J. A **59** (2023) no.11, 257

[64] R. Abbott, M. S. Albergo, *et al.* PoS **LATTICE2023** (2024), 035

[65] R. Abbott, A. Botev, *et al.* [arXiv:2401.10874 [hep-lat]].

[66] S. Bacchio, Phys. Rev. D **108** (2023) no.9, L091508 [arXiv:2305.07932 [hep-lat]].

[67] J. Finkenrath, [arXiv:2201.02216 [hep-lat]].

[68] J. Finkenrath, PoS **LATTICE2023** (2024), 022 [arXiv:2402.12176 [hep-lat]].

[69] R. Abbott, M. S. Albergo, *et al.* Phys. Rev. D **106** (2022) no.7, 074506

[70] J. Finkenrath, F. Knechtli and B. Leder, Comput. Phys. Commun. **184** (2013), 1522-1534

[71] B. Joo *et al.* [UKQCD], Phys. Rev. D **59** (1999), 114501 [arXiv:hep-lat/9810032 [hep-lat]].

[72] M. Hasenbusch, Phys. Rev. D **96** (2017) no.5, 054504 [arXiv:1706.04443 [hep-lat]].

[73] T. Eichhorn, G. Fuwa, C. Hoelbling and L. Varnhorst, [arXiv:2307.04742 [hep-lat]].

[74] T. Eichhorn, C. Hoelbling, P. Rouenhoff and L. Varnhorst, PoS **LATTICE2022** (2023), 009

[75] C. Bonanno, C. Bonati and M. D'Elia, JHEP **03** (2021), 111 [arXiv:2012.14000 [hep-lat]].

[76] C. Bonanno, M. D'Elia, B. Lucini and D. Vadacchino, Phys. Lett. B **833** (2022), 137281

[77] C. Bonanno, G. Clemente, M. D'Elia, L. Maio and L. Parente, [arXiv:2404.14151 [hep-lat]].

[78] F. Knechtli, T. Korzec, M. Peardon *et al.* Phys. Rev. D **106** (2022) no.3, 034501

[79] C. Alexandrou, S. Bacchio, G. Christou *et al.* Phys. Rev. D **108** (2023) no.9, 094510

[80] G. Parisi, R. Petronzio and F. Rapuano, Phys. Lett. B **128** (1983), 418-420

[81] M. Luscher and P. Weisz, JHEP **09** (2001), 010

[82] H. B. Meyer, JHEP **01** (2004), 030

[83] H. B. Meyer, JHEP **01** (2003), 048

[84] L. Barca, F. Knechtli, M. J. Peardon, S. Schaefer *et al.* PoS **LATTICE2023** (2024), 030

[85] L. Barca, F. Knechtli, M. J. Peardon, S. Schaefer *et al.* In preparation.

[86] M. Cè, L. Giusti and S. Schaefer, Phys. Rev. D **93** (2016) no.9, 094507

[87] M. Cè, L. Giusti and S. Schaefer, Phys. Rev. D **95** (2017) no.3, 034503

[88] M. Dalla Brida, L. Giusti, T. Harris and M. Pepe, Phys. Lett. B **816** (2021), 136191

[89] P. A. Boyle, G. Cossu, A. Yamaguchi and A. Portelli, PoS **LATTICE2015** (2016), 023

[90] A. Yamaguchi, P. Boyle, G. Cossu, *et al.* PoS **LATTICE2021** (2022), 035

[91] Innovative Algorithms For Applications On European Exascale Supercomputers https://www.inno4scale.eu