PROCEEDINGS OF SCIENCE

Scalar Mesons on the Lattice Using Stochastic Sources on GPU Architecture.

Dean Howarth[*]
RENSSELAER POLYTECHNIC INSTITUTE
E-mail: HOWARD3@RPI.EDU

Joel Giedt
RENSSELAER POLYTECHNIC INSTITUTE
E-mail: GEIDTJ@RPI.EDU

We describe our studies of multi-pion correlation functions computed using stochastic propagators in quenched lattice QCD, harnessing GPUs for acceleration. We find that projecting onto momentum states for the pions becomes enormously expensive and has a poor scaling with the linear extent of the lattice. We will use Lüscher's method and look for the σ in the $\pi_+\pi_- \to \pi_+\pi_-$ channel. The result is that multi-pion correlation functions must be considered, which inevitably involve all-to-all propagators, which are quite expensive and require many inversions. For this reason, GPUs are ideally suited to accelerating the calculation. For this work we have integrated the Columbia Physics System (CPS) and QUDA GPU inversion library, in the case of clover fermions. We describe some other challenges that we have uncovered, in particular getting hit with Amdahl's law for the tying together and tracing of propagators in our calculations, as well as momentum projections. We have also accelerated these parts of our calculation using GPUs and show some benchmarks.

The 32nd International Symposium on Lattice Field Theory
23-28 June, 2014
Columbia University New York, NY

[*] Speaker.

PoS(LATTICE2014)096

1. A Pseudo-scalar Correlation Function

In order to study the 0^{++}, σ scalar state on the lattice, we must create a state on the lattice with the same quantum numbers. One such state is $\pi_+\pi_-$, which we can create by inserting a π_+ at $(0,0)$ and a π_- at $(\vec{x},0)$. We then destroy the pions at $(\vec{y},t)$ and $(\vec{z},t)$ respectively, and study the decay of its correlation function as a function of time,

$$C_{2\pi}(t) = \langle\Omega|T\{\widehat{\pi}_-(\vec{z},t)\widehat{\pi}_+(\vec{y},t)\widehat{\pi}^\dagger_-(\vec{x},0)\widehat{\pi}^\dagger_+(\vec{0},0)\}|\Omega\rangle, \tag{1.1}$$

where,

$$\widehat{\pi}^\dagger_+(\vec{x},t) = \bar{u}(\vec{x},t)\gamma_5 d(\vec{x},t) = \widehat{\pi}_-(\vec{x},t), \qquad \widehat{\pi}^\dagger_-(\vec{x},t) = \bar{d}(\vec{x},t)\gamma_5 u(\vec{x},t) = \widehat{\pi}_+(\vec{x},t). \tag{1.2}$$

For a more rigorous treatment, we will be using isoscalar projection where the pseudoscalar operators $\widehat{P}_a(\vec{x},t)$ take the more general form,

$$\widehat{P}_a(\vec{x},t) = \frac{1}{2}\overline{\psi}(\vec{x},t)\gamma_5\tau_a\psi \quad \text{for} \quad a = 1,2,3. \tag{1.3}$$

with τ_a being the Pauli matrices. For now, we have restricted our investigation to (1.1) to simplify the structure of the correlation functions and test our GPU code.

We are ultimately aiming to measure the scattering phase shift of the $\pi_+\pi_- \to \sigma \to \pi_+\pi_-$ channel using Lüsher's finite volume method [2, 3, 4, 5].

2. Preliminary Calculations

As a preliminary calculation, we use the quenched approximation on a lattice of size of $L = 4, T = 8$. We used both point sources and stochastic sources in order to gauge the performance of using stochastic sources and the accuracy of our GPU inverter. The numerical details of this calculation is given in table 1: m_0 is the bare mass parameter, a is the physical lattice spacing in femtometres, and m_π is is the corresponding physical pion mass being simulated. N is the number of (quenched) gauge fields sampled and E_0 is the ground state energy observed in the $\pi_+\pi_- \to \pi_+\pi_-$ system. The calculation was done with zero pion momentum and the physical lattice spacing was calculated using [6].

We used 2×1000 random source vectors at each timeslice in each calculation, as well as point sources. We found that for the $4^3 \times 8$ lattice, 2×700 stochastic sources gave comparable results to point sources. Figure 1 shows the point source and stochastic source correlation functions for the $4^3 \times 8$ for 2×700 sources. Although the error on the effective mass of the ground state calculated from stochastic sources is larger than the point source value, this error is acceptable. Further, the small number of timeslices prohibit the execution of an effective mass fit as a plateau does not appear in the data. We therefore perform a hyperbolic cosh fit with the caveat that the calculated ground state value is heavily contaminated by exited states. Indeed, the β coefficient in these simulations, given the lattice sizes, places the system in the deconfined phase, in that the quarks are weakly coupled due to the high temperature and asymptotic freedom is lost. This extra caveat means that physical interpretation of the ground state is not possible. However, the motivation for these calculations was driven more by the need to check our code and stochastic techniques. Larger volumes and more physical parameters will be employed in future calculations whereupon results will be more 'physical.'

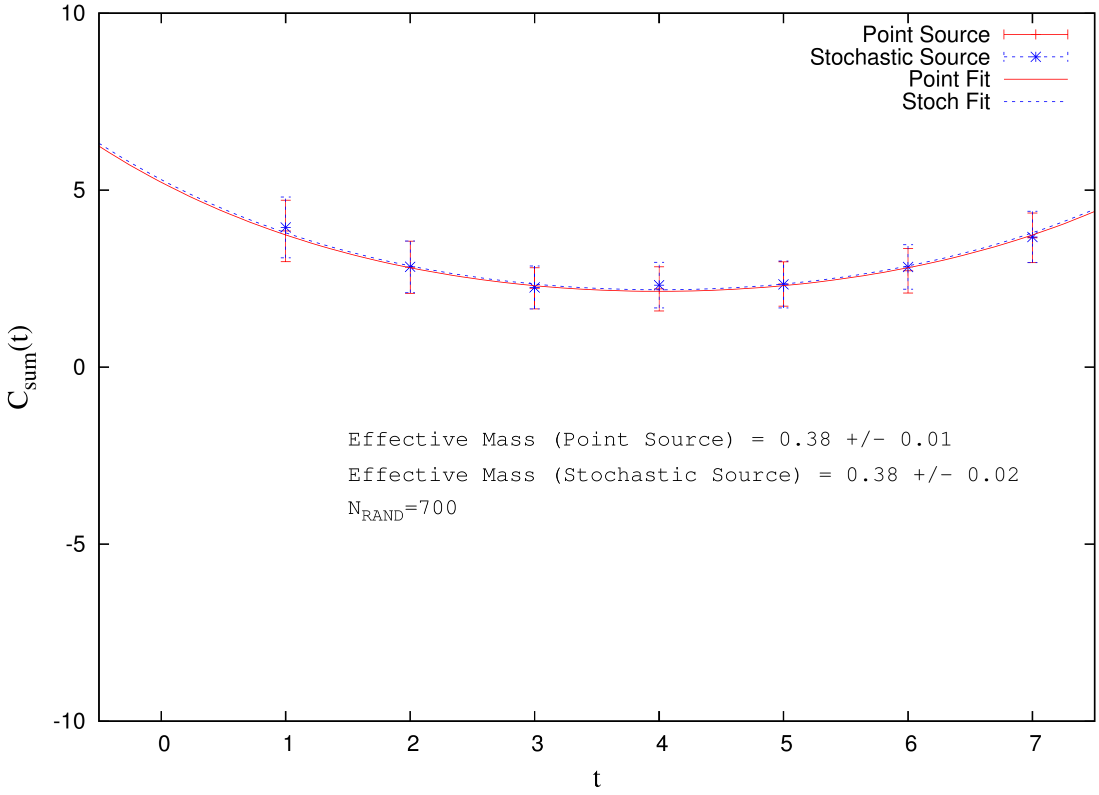

Figure 1: The sum correlation function for the $4^3 \times 8$ lattice. Although a very low ground state energy is seen in the plot, many caveats prevent us from making a physical interpretation. Nonetheless, stochastic sources are approximating point sources very well. [1]

3. Operator Improvement Techniques

Above the ground state energy, we need to improve our operators with smearing techniques and momentum projection. Furthermore, the number of stochastic sources need for good convergence can be reduced significantly by employing more sophisticated dilution schemes and operator construction. For the smearing, we choose to employ a Jacobi smearing operator[7],[8] ,

$$S(U,x,y) = \left(1 + \omega^2 \frac{\widetilde{\Delta}(x,y)}{4N}\right), \tag{3.1}$$

and smear the quark sources in order to reduce the level of contamination from excited states. The operator $\widetilde{\Delta}(x,y)$ is the usual discretised gauge covariant Laplacian operator. The prescription for using smearing is,

$$\widetilde{\psi}(\vec{x},t) = \left(1 + \omega^2 \frac{\widetilde{\Delta}(x,y)}{4N}\right)^N \widetilde{\psi}(\vec{y},t). \tag{3.2}$$

We will also use dilution schemes for the stochastic operators. On top of the spin-colour-time dilution, we can spatially dilute the sources on the time slice. Obviously, a great deal of different dilution schemes are possible and a systematic investigation is necessary to determine the optimal scheme.

Another method we will use is the so called stochastic LapH of Morningstar et al.[9] whereupon the lowest N_v eigenvectors of the smearing operator,

$$S'(U,x,y) = \left(\sigma_s^2 + \widetilde{\Delta}(x,y)\right), \tag{3.3}$$

are calculated, a matrix V_s whose columns are in one to one correspondence with the lowest N_v eigenvectors, and the smearing operator,

$$S' = V_s V_s^\dagger \tag{3.4}$$

is used.

4. Software

In order to invert the matrix M and perform the Fourier transforms, a significant amount of computing power is needed. In general, three architecture types are available: a single CPU, known as scalar, multiple CPUs, and GPU (Graphics Processing Unit.) The latter two are known as parallel as they can perform multiple floating point operations simultaneously. For the calculations proposed, code was developed on all three architectures and comparisons made between all three.

A large selection of open source code libraries are available and we chose to use CPS (Columbia Physics System) which has a comprehensive range of functions for QCD, and QUDA, which utilises the proprietary CUDA libraries for the GPU code. In order to use the functionality of CPS on GPUs, we needed to rearrange the data that CPS produces, specifically, the layout of the clover matrix and the layout of the gauge fields. his is done with two custom written functions. The rearranged gauge field and clover matrix arrays are then passed to the GPUs via QUDA. We have written the code so that a only a single file in CPS needs to be changed, QUDA need not be modified. Furthermore, we can utilise the GPUs to perform the Fourier transformations, which required us to write GPU kernels, though as we shall see, it is more efficient to perform inversion on the GPU and the Fourier transforms on BG/Q. In order to implement the various operator improvement techniques, several additions to the CPS source code had to be contracted. We did this by modification of existing class structures.

One can view and download the modified source code necessary to implement GPU inversion and the extra propagator classes at http://homepages.rpi.edu/~howard3/ where instructions on configuration flags and installation can also be found.

5. Hardware and Benchmarks

The calculations can be performed on three different types of architecture; scalar, multi-CPU, and GPU. The scalar machine we used had a quad core 2.40 GHz E5620 Intel Xeon Processor[1]. The GPU was a NVIDA Tesla C2075[2] which was connected to the scalar machine and used the host

[1] http://ark.intel.com/products/47925/Intel-Xeon-Processor-E5620

[2] http://www.nvidia.com/docs/IO/43395/NV-DS-Tesla-C2075.pdf

CPU. The multi-CPU was the BG/Q installed at the Center for Computaional Innovations[3] In tables 2 and 3 we decompose the calculation into separate parts and give the timings where appropriate for $4^3 \times 8$ and $8^3 \times 16$ lattices respectively. Instances where timing are not applicable are indicated with a dash (-). For instance, the heatbath updates are not performed on the GPU, rather they are done on the host CPU. Further, communications between CPUs is only applicable to the BG/Q architecture. The BG/Q benchmarks quoted are for the best possible QMP geometry available for the calculation i.e., each core is assigned two lattice points in every spacetime direction (16 lattice points per core). Other geometries are available where more lattice points are assigned to fewer cores, but naturally this will not give the best possible performance for the architecture.

The (Therm.) column gives the time for 5000 initial heatbath updates to thermalise the gauge configuration. (Comms.) is the amount of time that BG/Q spent transferring data between nodes, specifically for the Fourier transforms. (100 Inv.) is the average time taken for 100 inversions, (F.T) is the total time taken to perform all the Fourier transformations, (D.P) is the time taken to collect a single data point with 200 heatbath updates between data taking to minimise autocorrelation effects and (100 D.P.) is the time taken to collect 100 data points, enough to give a reasonably small error on the final result.

From these data one can see some striking differences. For the $4^3 \times 8$ lattice, one sees that the single GPU outperforms the BG/Q for inversion time, which we put down to overhead on the BG/Q. Fourier transforms are consistently faster in the BG/Q. This suggests that we would be able to split our calculation across two architectures and gain maximum efficiency from both if we:

- Simulate the gauge fields on BG/Q and save them to memory.
- Transfer the gauge arrays to a GPU system to perform the inversions.
- Transfer the propagator data back to BG/Q for Fourier transformation.

We predict that this will be the best use of resources available to us.

6. Outlook

The code written to implement operator improvement has been tested against point source calculation for small lattices and we find good agreement. When we move to larger lattice sizes, we will use of the available operator improvement techniques given here and give an exposé on their relative efficiencies can be drawn. We will also apply the finite volume method of Lüscher and and attempt to discern if the σ resonance at $\sim$500Mev in the 2π spectrum is indeed a distinct intermediate state, or simply a feature of the peak of the continuum spectrum.

References

[1] J. Giedt, D. Howarth: *Stochastic propagators for multi-pion correlation functions in lattice QCD with GPUs*, [ArXiV: hep-lat/1405.4524]

[2] M. Luscher, Commun.Math.Phys. 104, 177 (1986).

[3] https://secure.cci.rpi.edu/wiki/index.php/Blue_Gene/Q

[3] M. Luscher, Commun.Math.Phys. 105, 153 (1986).

[4] M. Luscher, Nucl.Phys. B354, 531 (1991).

[5] M. Luscher, Nucl.Phys. B364, 237 (1991)

[6] S. Necco and R. Sommer: *The $N_f = 0$ heavy quark potential from short to intermediate distances*, Nucl. Phys. B 622, 328 (2002) 65, 67 [ArXiV: hep-lat/0108008]

[7] S. Güsken et al., Nucl. Phys. B (Proc. Suppl.) 17 (1990) 361 and Phys. Lett. B227 (1989) 266.

[8] E. Eichten, G. Hockney, and H. B. Thacker, Nucl. Phys. B (Proc. Suppl.) 17 (1990) 529.

[9] M. Peardon et al.: *A novel quark-field creation operator construction for hadronic physics in lattice QCD*, Phys. Rev. D80, 054506 (2009) [ArXiV: hep-lat/0905.2160]

L^3T	β	m_0	a(fm)	m_π(GeV)	N	E_0(GeV)
$4^3 \times 8$	5.96	0.300	0.102	2.00	100	0.76

Table 1: Data for preliminary simulation of the $4^3 \times 8$ lattice.

$4^3 \times 8$ Lattice

Source	Arch.	Therm.(s)	Comms.(s)	100 Inv.(s)	F.T.(s)	D.P.(h)	100 D.P. (d)
Point	Scalar	75.1	-	115	67.3	0.18	0.76
	GPU	-	-	71.2	14.0	0.11	0.44
	BG/Q	72.9	0.15	132	4.16	0.19	0.79
2×700 Stoch.	Scalar	79.8	-	96.5	72.1	3.0	12.6
	GPU	-	-	55.6	14.3	1.73	7.23
	BG/Q	73.7	0.77	130	4.20	4.1	16.9

Table 2: Benchmarks for the $4^3 \times 8$ lattice.

$8^3 \times 16$ Lattice

Source	Arch.	Therm.(s)	Comms.(s)	100 Inv.(s)	F.T.(s)	D.P.(h)	100 D.P. (d)
Point	Scalar	1260	-	2520	8.24×10^4	80.24	334
	GPU	-	-	1020	2.92×10^4	31.3	131
	BG/Q	72.8	2.85	153	175	3.17	13.2
2×700 Stoch.	Scalar	1250	-	2050	6.80×10^4	146.6	610
	GPU	-	-	770	2.98×10^4	56.14	234
	BG/Q	73.7	8.95	153	175	10.13	42.2

Table 3: Benchmarks for the $8^3 \times 16$ lattice.