PROCEEDINGS OF SCIENCE

Twisted-Mass Lattice QCD using OpenCL

Matthias Bach* **and Volker Lindenstruth**
Frankfurt Institute for Advanced Studies / Institut für Informatik, Goethe-Universität Frankfurt am Main
E-mail: bach@compeng.uni-frankfurt.de
E-mail: voli@compeng.de

Christopher Pinke and Owe Philipsen
Institut für theoretische Physik, Goethe-Universität Frankfurt am Main
E-mail: pinke@th.physik.uni-frankfurt.de
E-mail: philipsen@th.physik.uni-frankfurt.de

Graphics Processing Units (GPUs) are by now an established tool for Lattice QCD applications. I present an update on our OpenCL based code for Lattice QCD with twisted-mass fermions. On current generation AMD GPUs we now reach over 100 GFLOPS in double-precision $\not{D}$ and 70 GFLOPS in our double-precision inverter. For the hybrid Monte-Carlo (HMC) we improve energy-efficiency by a factor of four over a plain CPU system. We also found one 4-GPU node to provide about 12 times the throughput of a pure CPU system of comparable cost.

31st International Symposium on Lattice Field Theory - LATTICE 2013
July 29 - August 3, 2013
Mainz, Germany

*Speaker.

PoS(LATTICE 2013)032

1. Introduction

Originating in the high-end computer gaming market, Graphics Processing Units (GPUs) nowadays offer highest computing capabilities at a very attractive price-per-flop and flop-per-watt ratios. Thus, in the last years there has been a lot of development to utilize GPUs for Lattice QCD computations. However, while the pioneer work in the field [1] was in principle platform agnostic, most later developments are based on NVIDIA CUDA. This limits these applications to hardware by this single vendor[1].

In the last years, two High-Performance Computing (HPC) systems with GPUs have been built by groups from Frankfurt. LOEWE-CSC [2] provides 786 AMD Radeon HD 5870 GPU and showed excellent energy-efficiency, ranking 8th in the Green500 [3] list of November 2010. SANAM—built by a cooperation of the Frankfurt Institute for Advanced Studies (FIAS) and the King Abdulaziz City for Science and Technology (KACST)—is equipped with two AMD FirePro S10000—that is, four GPUs—in each of its 300 nodes and placed 2nd in the Green500 of June 2012.

To utilize clusters like LOEWE-CSC and SANAM—and to get out of the vendor lock—an application which can utilize GPUs but does not rely on NVIDIA CUDA is required. Here, we present our progress in the development of CL^{2}QCD, an OpenCL based application for Lattice QCD using twisted-mass Wilson fermions [4]. We show how the performance of our applications scales from LOEWE-CSC to SANAM, and how it currently scales to multiple GPUs.

GPUs are commonly viewed as a cheap solution for HPC. However, unlike dedicated systems they have not been optimized specifically for Lattice QCD computations. Thus, it is not necessarily clear whether dedicated hardware or GPUs provide the best performance-per-price. Therefore, we also present a cost-per-flop comparison of CL^{2}QCD on GPUs in comparison to Lattice QCD applications running on other systems.

The benchmark used for the Green500 is quite different from Lattice QCD. Where Lattice QCD is bandwidth bound, the High-Performance Linpack (HPL) is compute bound. Thus, it is not clear how the energy-efficiency transfers to Lattice QCD computations. Therefore, we also present the energy-efficiency of CL^{2}QCD utilizing GPUs and that of a reference application running on Central Processing Units (CPUs).

2. Performance

In terms of compute time, the $\not{D}$ kernel dominates both the analysis and the configuration generation stage. Thus, its performance is of highest importance for the overall application performance. Figure 1 shows the performance of the $\not{D}$ kernel on four different GPUs for a variety of lattice sizes.

The elderly AMD Radeon HD 5870 performs at about 70 GFLOPS for a wide variety of lattice sizes [4]. For smaller lattices, the performance drops slightly but is still above 50 GFLOPS.

The performance scales well with the higher memory bandwidth of the AMD FirePro S10000. It provides about 100 GFLOPS in double-precision $\not{D}$ for a wide variety of lattice sizes. The AMD

[1] We are aware that NVIDIA opened the standard for implementation by other vendors. However, so far no other vendor does support NVIDIA CUDA.

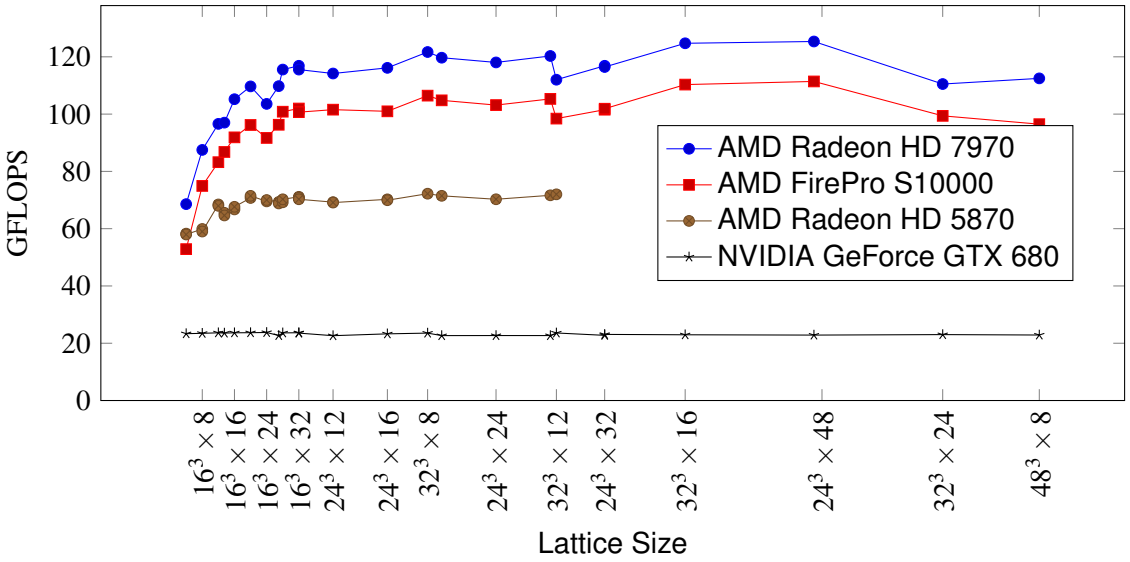


Figure 1: Performance of the double-precision $\not{D}$ kernel

Radeon HD 7970, equipped with slightly more memory bandwidth than the AMD FirePro S10000, achieves about 120 GFLOPS and peaks up to 125 GFLOPS.

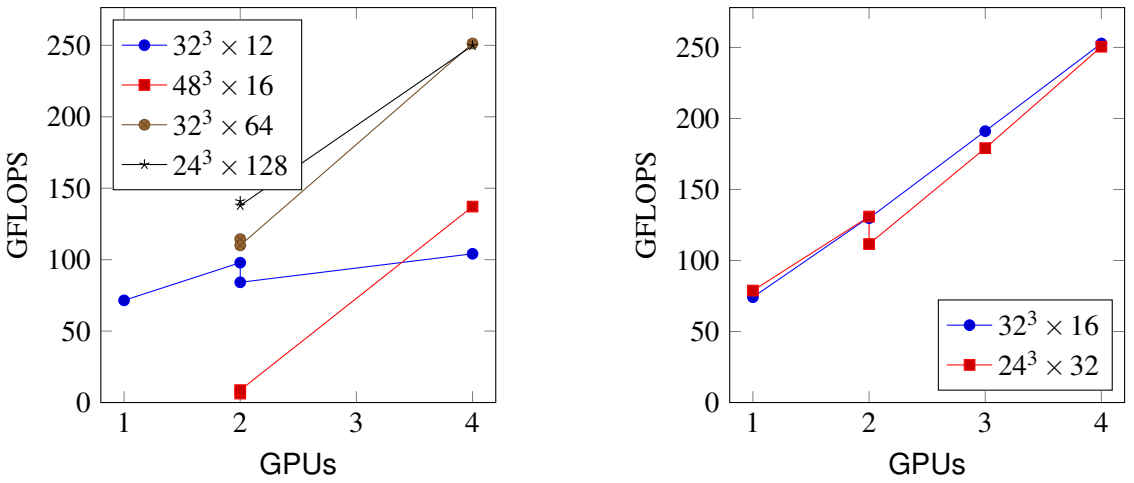
On the NVIDIA GeForce GTX 680 the $\not{D}$ kernel does not perform as well. There, it is currently hindered by register spilling. Contrary to CUDA, NVIDIA's OpenCL implementation does currently not allow to request more registers per thread from the compiler, which might solve the issue. In addition, the NVIDIA GeForce GTX 680 is a suboptimal platform for the double-precision kernel, as it has only 1/8th of the double-precision performance of the Tesla series GPUs. Performance-wise, on the NVIDIA platform our code cannot compete with QUDA, for which performances exceeding 300 GFLOPS in single-precision have been reported [5]. The same is true for the Intel Xeon Phi [6]. Yet, our code provides the advantage of portability and provides excellent double-precision performance.

The Conjugate Gradients (CG) solver achieves about 75 % of the $\not{D}$ performance. In double precision, about 50 GFLOPS are achieved on the AMD Radeon HD 5870. For the AMD FirePro S10000 this increases to about 73 GFLOPS.

The scaling to multiple GPUs in SANAM is shown in Figure 2. Figure 2a shows the strong scaling for the inverter. Due to memory constraints single-GPU data is only available for the $32^3 \times 12$ lattice. For vacuum lattices with a large number of time slices, two GPUs can reach 140 GFLOPS, nearly twice the single-GPU peak performance in the CG. Reaching 250 GFLOPS, four GPUs achieve an efficiency of 85 % versus the single-GPU peak.

In the strong-scaling case the local lattice-size reduces with each added GPU. This can result in reduced performance of the $\not{D}$ kernel execution. Therefore, Figure 2b shows the scaling behaviour given a fixed local lattice size. The results match that of the strong scaling case. Obviously the reduction in local lattice size is not an issue for the lattices analysed.

Especially for thermal lattices with a small number of time slices, throughput scaling is also important because of the need for finites size analyses and the large statistic required. Running separate instances of the hybrid Monte-Carlo (HMC) on each GPU in a node of SANAM, no sig-



(a) Strong scaling for multiple lattice sizes

(b) Weak scaling for two local lattice sizes

Figure 2: Scaling of the CG solver in SANAM

nificant slowdown compared to the single-GPU case is observed. Thus, throughput scales perfectly to all four GPUs.

As Figure 3 shows, the HMC performance of the AMD FirePro S10000 is about twice that of the AMD Radeon HD 5870 and about four times that of two AMD Opteron 6172 running tmlqcd.

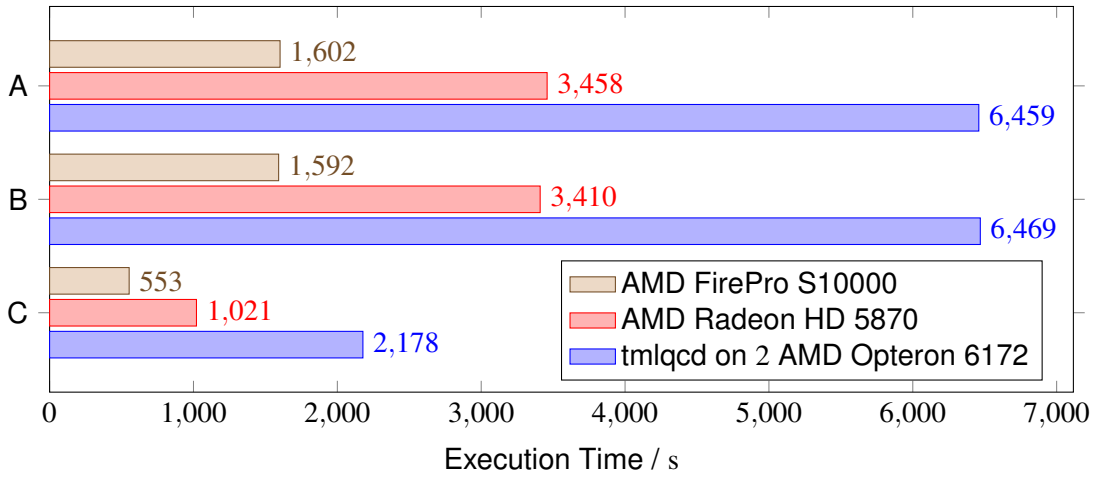


Figure 3: HMC runtime in seconds for the set-ups from [4]

3. Total Cost of Acquisition

Given the fact that budgets are usually limited, the Total Cost of Acquisition (TCA) is an important metric for every system. Different lattice sizes and studies require different size systems. As both, performance and cost approximately scale with system size, it is reasonable to compare the TCA normalized to the system performance in Lattice QCD.

Based on publicly available data [7, 8], we estimate the normalized TCA of Sequoia, the currently largest BlueGene/Q system, to be about \$0.040/MFLOPS in the CG solver. SANAM is able to match this with a normalized TCA of \$0.033/MFLOPS. However, Sequoia is still in the advantage of being capable of PFLOPS calculations, which are out of SANAM's reach. In any case, TCA has come long way. It dropped by a factor of 400 since the Gordon Bell Award for the \$13.2/MFLOPS of the QCDSP machines in 1988 [9].

As capacity systems are not always available, it is also of interest how far always-available local systems could be optimized. Therefore we have configured several hypothetical systems based on a high end gaming PC[2]. Using the single-precision CG performance values from [6], we get a normalized TCA of \$0.018/MFLOPS for the Intel Xeon Phi 5110P and of \$0.023/MFLOPS for the NVIDIA Tesla K20. These are just back-of-the-envelope calculations, though. For the AMD FirePro S10000 and the AMD Radeon HD 5870 we achieve normalized TCAs of \$0.027/MFLOPS and \$0.019/MFLOPS for the double-precision CG solver. In conclusion, all accelerator platforms provide a similar normalized TCA.

A proper comparison should actually be based on HMC trajectories and not only on the CG or the solver [10]. With its four GPUs a SANAM node has 16 times the throughput of a CPU node. But, it costs only 33 % more. Thus, the normalized TCA of the CPU system is 12 times that of a GPU-equipped node.

4. Energy Efficiency

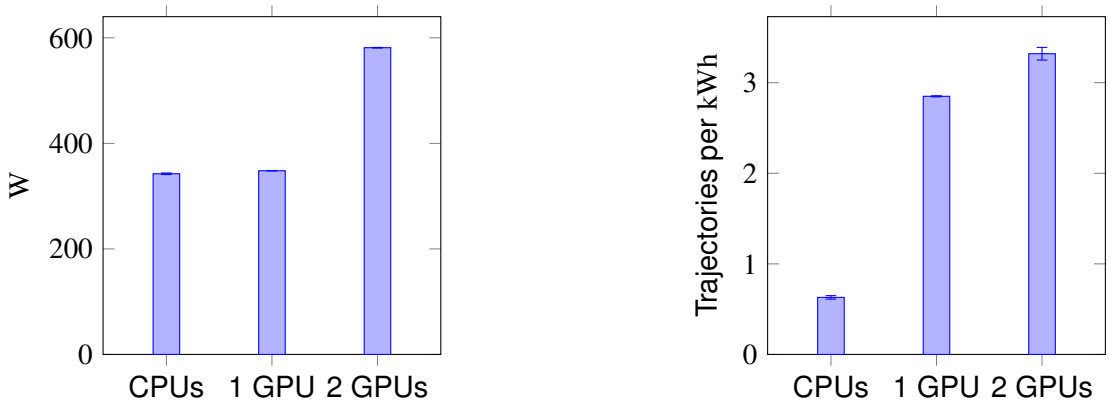
Given today's energy costs and the cooling challenge, the energy required to solve a given problem is one of the key performance indicators. Therefore, we compared the energy consumption of running CL^{2}QCD with the energy consumption of running tmlqcd [11].

There are two metrics of interest. The average power consumption while running the application defines the required cooling. The total energy required to get to the result is important for the ecological impact of the system and the monetary costs. For the purpose of this comparison we introduce an efficiency-metric which is defined as the number of trajectories the HMC performs per kWh.

The HMC algorithm is run on a given configuration of a $32^3 \times 12$ lattice using two flavours of twisted-mass Wilson fermions and $m_\pi \approx 270\,\mathrm{MeV}$. We performed the comparison using two different systems: The CPU system is equipped with two AMD Opteron 6220 and no GPUs, while CL^{2}QCD runs on a system equipped with a varying number of AMD Radeon HD 7970 GPUs and two AMD Opteron 6172. For the measurement with two GPUs, we ran two instances of CL^{2}QCD. The energy consumption is measured for the whole system over the whole application runtime using an LMG95 [12] power meter. As the runtime of the HMC algorithm is influenced by the random numbers used, all measurements were performed for ten random number seeds and the analysis was performed on the average result.

Figure 4a shows the power consumption of the systems averaged over the duration of one step of the HMC algorithm. Equipped with one GPU, the system requires 348 W running CL^{2}QCD.

[2] The calculation is based on a price of 1299 € for the XL Core system from the web shop at `http://www.one.de` on 5 July 2013. It includes a value added tax (VAT) of 19 % and an AMD Radeon HD 7970. The price of the latter has been deducted, of course.



(a) Average power consumption during an HMC trajectory

(b) Energy-efficiency in the HMC

Figure 4: Energy consumption of CPU and GPU systems in comparison

This is hardly more than the 343 W required by the pure CPU system running tmlqcd. Adding the second GPU to run a second instance of the HMC increases the power consumption by about 235 W.

Figure 4b shows the energy efficiency of the different systems when performing HMC calculations. The CPU system performs about 0.63 HMC trajectories per kWh. The single-GPU system is 4.5 times as efficient. It can perform 2.85 HMC trajectories per kWh. The two-GPU system shows the best energy efficiency, performing 3.3 HMC trajectories per kWh. This is 5.25 times the energy efficiency of the CPU system.

5. Conclusion

CL^{2}QCD is an OpenCL based Lattice QCD framework which allows to perform the whole HMC on the GPU. On the AMD FirePro S10000, its $\not{D}$ kernel provides 100 GFLOPS in double precision. Based on OpenCL it can also be used on NVIDIA GPUs and on CPUs.

So far, development has been focused on thermal lattices optimising single-GPU performance. As long as a thermal lattice fits on one unit, only a minimal speed-up can be reached using multiple GPUs within a single node. Yet, for vacuum lattices the CG solver shows linear scaling with an efficiency of about 85 %

In terms of normalized TCA, GPU-based clusters match conventional large-scale Lattice QCD systems. Contrary to those, however, they can be scaled up from a single node. In comparison to a CPU system of comparable cost, the GPU system achieves 12 times the throughput.

The usage of GPUs provides a significant advantage in terms of energy efficiency. A single-GPU system operates at a similar power consumption level as a typical CPU system. Taking into account its higher performance, the GPU based system is four times as energy-efficient.

Currently, support for staggered fermions is being added to CL^{2}QCD. To further improve performance, mixed-precision solvers as well as parallelization to multiple nodes should be evaluated. Also—to improve the platform independence—CL^{2}QCD will have to be further optimized for the NVIDIA and the Intel platforms.

Acknowledgements

O. P. and C. P. are supported by the German BMBF grant *FAIR theory: the QCD phase diagram at vanishing and finite baryon density*, 06MS9150. M. B., O. P, and C. P. are supported by the Helmholtz International Center for FAIR within the LOEWE program of the State of Hesse. M.B. and C.P. are supported by the GSI Helmholtzzentrum für Schwerionenforschung. C.P. acknowledges travel support by the Helmholtz Graduate School HIRe for FAIR.

The authors want to thank Lars Zeidlewicz for his participation in the early stages of this project. Some of the calculations have been performed on LOEWE-CSC and Sanam. The authors thank the respective teams for all the support.

References

[1] G. I. Egri, Z. Fodor, C. Hoelbling, S. D. Katz, D. Nogradi, and K. K. Szabo, "Lattice QCD as a video game", *Computer Physics Communications*, vol. 177, no. 8, p. 11, Nov. 2006, ISSN: 0010-4655. DOI: 10.1016/j.cpc.2007.06.005. arXiv: 0611022 [hep-lat].

[2] M. Bach, J. de Cuveland, H. Ebermann, D. Eschweiler, J. Gerhard, S. Kalcher, M. Kretz, V. Lindenstruth, H.-J. Ludde, M. Pollok, and D. Rohr, "A Comprehensive Approach for a Power Efficient General Purpose Supercomputer", in *2013 21st Euromicro International Conference on Parallel, Distributed, and Network-Based Processing*, IEEE, Feb. 2013, pp. 336–342, ISBN: 978-1-4673-5321-2. DOI: 10.1109/PDP.2013.55.

[3] W.-c. Feng and K. W. Cameron, *The Green500*. [Online]. Available: http://www.green500.org.

[4] M. Bach, V. Lindenstruth, O. Philipsen, and C. Pinke, "Lattice QCD based on OpenCL", *Computer Physics Communications*, p. 19, Mar. 2013, ISSN: 00104655. DOI: 10.1016/j.cpc.2013.03.020. arXiv: 1209.5942.

[5] M. A. Clark and R. Babich, "High-efficiency Lattice QCD computations on the Fermi architecture", in *2012 Innovative Parallel Computing (InPar)*, IEEE, May 2012, pp. 1–9, ISBN: 978-1-4673-2633-9. DOI: 10.1109/InPar.2012.6339591.

[6] B. Joó, D. D. Kalamkar, K. Vaidyanathan, M. Smelyanskiy, K. Pamnany, V. W. Lee, P. Dubey, and W. I. Watson, "Lattice QCD on Intel Xeon Phi Coprocessors", in *Supercomputing*, J. M. Kunkel, T. Ludwig, and H. W. Meuer, Eds., Springer Berlin Heidelberg, 2013, pp. 40–54, ISBN: 978-3-642-38750-0. DOI: 10.1007/978-3-642-38750-0_4.

[7] P. Boyle, "The BlueGene/Q supercomputer", in *The 30 International Symposium on Lattice Field Theory - Lattice 2012*, 2012. [Online]. Available: http://pos.sissa.it/archive/conferences/164/020/Lattice%202012_020.pdf.

[8] J. Brodkin, *With 16 petaflops and 1.6M cores, DOE supercomputer is world's fastest*, 2012. [Online]. Available: http://arstechnica.com/information-technology/2012/06/with-16-petaflops-and-1-6m-cores-doe-supercomputer-is-worlds-fastest/ (visited on 07/06/2012).

[9] D. Chen, P. Chen, N. H. Christ, R. G. Edwards, G. Fleming, A. Gara, S. Hansen, C. Jung, A. Kahler, S. Kasow, A. D. Kennedy, G. Kilcup, Y. Luo, C. Malureanu, R. D. Mawhinney, J. Parsons, C. Sui, P. Vranas, and Y. Zhestkov, "QCDSP machines: design, performance and cost", in *Proceedings of the 1998 ACM/IEEE conference on Supercomputing (CDROM)*, ser. Supercomputing '98, Washington, DC, USA: IEEE Computer Society, 1998, ISBN: 0-89791-984-X.

[10] M. A. Clark, "QCD on GPUs: cost effective supercomputing", p. 14, Dec. 2009. arXiv: 0912.2268.

[11] K. Jansen and C. Urbach, "tmLQCD: a program suite to simulate Wilson Twisted mass Lattice QCD", *Quantum*, no. May 2009, pp. 1–44, 2009. arXiv: 0905.3331.

[12] *ZES Zimmer LMG 95*. [Online]. Available: http://www.zes.com/english/products/single-phase-precision-power-analyzer-lmg95.html (visited on 09/25/2013).