PROCEEDINGS OF SCIENCE

https://pos.sissa.it/

PoS(ISGC2022)015

Large Scale ARM Computing Cluster and its Application in HEP

Yaosong Cheng[a,*], Yujiang Bi[a,c], WeiSun[a], Yaodong Cheng[a,b,c], Jingyan Shi[a] and Minxing Zhang[a,b]

a Institute of High Energy Physics, CAS

100049 Beijing, China

b University of Chinese Academy of Sciences

100049 Beijing, China

c Tianfu Cosmic Ray Research Center

610213 Chengdu, China

E-mail: chengys@ihep.ac.cn, biyujiang@ihep.ac.cn

With the advancement of several large scientific experiments in the field of high-energy physics, the speed of data generation continues to increase and will reach exabyte(EB) levels in the next decade. This puts forward higher requests for data storage and computation processing. Reducing reliance on a single chip architecture can provide more cost-effective storage and computing solutions. We built a supercomputer cluster with 9,600 computing cores based on the ARM architecture in Dongguan, China. On this ARM cluster, we ported some common software for high-energy physics and made some optimizations for Lattice quantum chromodynamics(LQCD). In addition, we compared the performance of ARM and X86 in many application scenarios, which is beneficial to improve computational efficiency.

International Symposium on Grids & Clouds 2022 (ISGC 2022)
21 - 25 March, 2022
Online, Academia Sinica Computing Centre (ASGC), Taipei, Taiwan

*Speaker

1 Introduction

ARM is a Reduced Instruction Set Computing(RISC) microprocessor that is characterized by fixed instruction length, simple addressing, and extensive use of registers. Compared to Intel's x86 architecture, it has a smaller size, lower power consumption, and lower cost. These features give ARM a certain advantage in the high-performance computing field that requires a lot of power costs. Nowadays, most ARM chips use multi-core architecture, which brings it better parallelism and compatibility. However, it has the disadvantage that the performance of individual cores is weaker than X86 and the application ecology is not perfect.

In the early days, ARM chips were mainly used in mobile terminal products. But with the introduction of the new generation ARMv8 architecture, more and more vendors are developing arm-based server-level chips. ARM chips are also increasingly used in distributed storage, big data analytics, cloud computing, and other scenarios. For example, Cavium ThunderX in the United States [1], Fujitsu A64FX in Japan [2], Huawei Kunpeng 920 in China, etc.

In 2021, the Japanese supercomputer Post-k was officially operated. It is the highest performing supercomputer in the world today, using the A64FX chip with ARM architecture. In 2018, AWS announced the Amazon Graviton Processor, a 64-bit ARM processor based on the Cortex-A72 micro-architecture[3]. This processor is made available on Amazon Elastic Compute Cloud (EC2) and improved the cost effectiveness of computing instances by 55%.

As a classic application scenario of high-performance computing, high-energy physics experiments pay great attention to the software ecology based on the ARM architecture. Many commonly used software have ported from x86 to ARM, such as LHCb stack, LCG software stack, and ATLAS software stack. These works are of great significance to the development of software in the field of high-energy physics. Marek et al. ported the LCG software stack to ARM[4], and used the program "multicore/mtbb201_ parallelHistoFill.C" in ROOT tutorials for testing. The results show that while single-core ARM servers do not perform as well as x86_64, the CPU time for X86 and ARM tend to be similar as the number of cores increases in multi-core benchmark tests. And the advantage of ARM processor lies mainly in the good cost performance, rather than high performance. Laura Promberger et al. Work on porting LHCb stack from x86 to aarch64 and ppc64le[5]. At the same time, they analyzed and proved the importance of cross-platform support for vectorization.

2 Large Scale ARM Computing Cluster

2.1 Hardware facilities

We have built the largest ARM computing cluster in the field of high-energy physics. Traditional computing clusters include many different parts, so our ARM cluster includes 100 ARM Compute nodes with 9600 CPU cores. And each node is interconnected by a 100Gb RoCE network. The storage space includes 5.7PB of file storage and 1.6PB of block storage. In addition, it includes management nodes, login nodes, etc. The hardware architecture of the cluster is shown as follows.

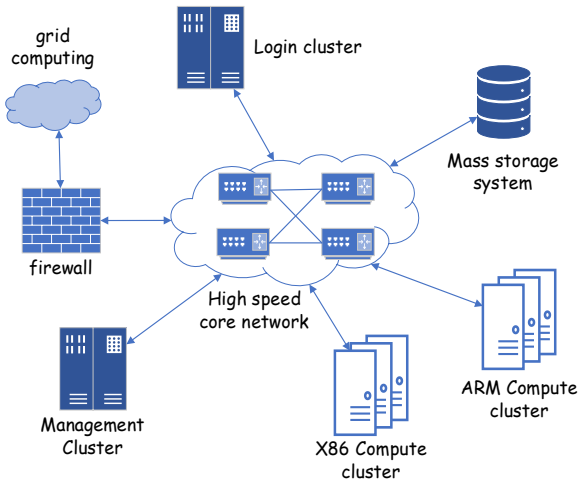

Fig.1 Hardware architecture for ARM computing cluster

The cluster is powered by a Huawei Kunpeng 920 CPU with ARMv8.2 architecture. Kunpeng has three models of CPU with 32, 48, and 64 cores. The maximum main frequency is 3.0GHz, and it also supports for eight-way DDR4 controller and two 100Gb RoCE network cards. These components are integrated on the Taishan 200K high-performance computing server. One compute node is configured with two CPUs (48 cores) and a 256 GB memory card. It also supports an Atlas 300 acceleration card for AI computing. And it is worth mentioning that the server additionally includes a hardware acceleration subsystem that supports some encryption and compression algorithms.

2.2 Computing environment

This ARM computing cluster is built in Dongguan, Guangdong Province, China, and its computing environment is shown in the following figure.

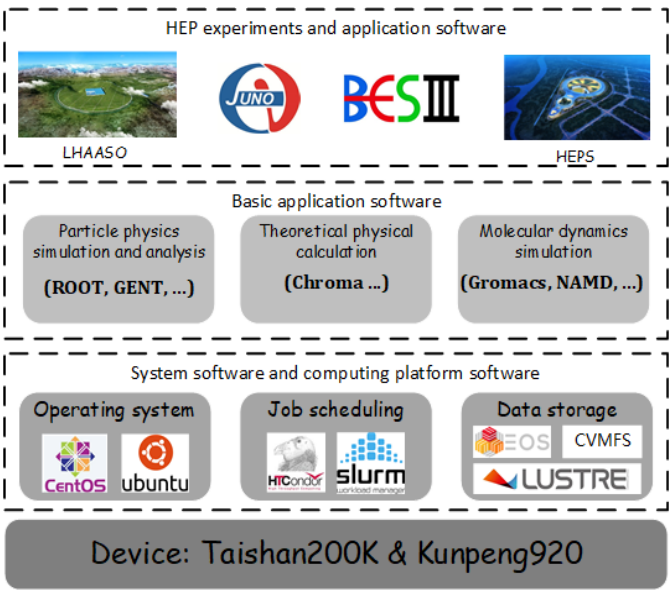

Fig.2 Environment for ARM computing clusters

The bottom layer is an arm-based hardware device. We first installed some system software such as CentOS, HTCondor, Slurm, and some data storage software such as Lustre, EOS, and CVMFS. The third layer is some middleware, mainly basic applications of high-energy physics, including ROOT, GEANT, Chroma, etc. At the top level, it supports some high-energy physics experiments and applications, including LHAASO, JUNO, BES3, and HEPs.

We use a cloud management platform, a distributed HTC, and a unified authentication system to ensure consistent usage patterns between the ARM computing cluster and the IHEP computing cluster. In addition, this ARM computing cluster provides the basis for our research on heterogeneous computing.

3 Application in HEP

3.1 Software porting

The ARM cluster currently supports several experiments, such as “Large High Altitude Air Shower Observatory (LHAASO)” and “Lattice quantum chromedynamics (LQCD)”. So we ported some common software for high-energy physics, such as EOS, ROOT, GEANT, etc. The difficulty of porting is that there is no official arm version of many dependencies. We needed to recompile the source code and solve some parameter configuration problems at the same time. EOS has many dependencies, including XRootD, RocksDB, Protobuf 3, ISA-L, etc. Sometimes, we also have to modify some assembly instructions in the program [6]. For example, the data verification part of EOS has a small amount of Intel assembly code related to the crc32 function, and we need to replace the assembly instructions with the corresponding ARM assembly instructions.

3.2 Performance Test

We performed some classical tests to compare the performance of ARM and X86 in the field of high-energy physics. Because some of the experiments were conducted early, the processors tested were not up to date, but they were versions widely used in data centers. Certainly, we also welcome evaluation and discussion of different processors.

The benchmarks include single-core and entire server. In the single-core test, we focus on the time to complete a task that only calls only one CPU core. In the entire server test, all cores of the server are used (excluding the hyper-threaded processor). Although it was not determined that each CPU core reached its performance limit in the test, comparing the time they spent on the same task can still indicate which CPU architecture is more helpful for HEP jobs.

3.2.1 HEP-SPEC06 benchmark test

The first is the HEP-SPEC06 (or HS06)benchmark test, which is the standard tool for evaluating CPU performance in HEP.Run one copy of the benchmark per logical CPU core, and the final score is a geometric mean of seven C++ benchmarks.

For this test, the ARM CPU is Kunpeng 920 with 64 cores of Huawei. The entire server contains two CPUs with a total of 128 cores. For comparison, we chose two X86 CPUs, Intel E5-2620@2.0G Hz, which contains 12 cores, and Intel Gold 6230@2.1G Hz, which contains 40 cores. The test results are as follows.

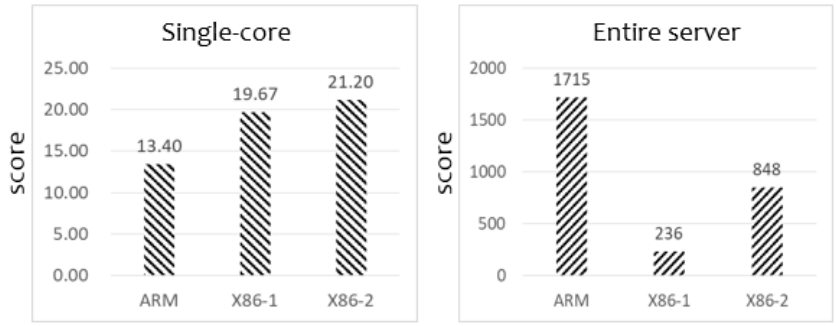

Fig.3 Results of the HEP-SPEC06 benchmark test

We can see that the performance of the ARM single core is lower than the X86. But for the entire server, the performance of the Kunpeng920 is 7.27 times that of the first X86 server (E5-2620) and 2.02 times that of the second (Gold 6230).

3.2.2 Experimental application evaluation

We next test for LHAASO's job on ARM. The first is the LHAASO-WCDA case reconstruction job, which runs based on the ROOT program and is a kind of data-intensive computational task. Similar to the actual case, the job under test contains 418,816 cases, and one job uses only one CPU core. *Minhit* is an important parameter in the case reconstruction job. So we counted the computing time of a single CPU core at different *minhit* separately (*minhit=1500 & minhit=800*). The results are shown in the following figure.

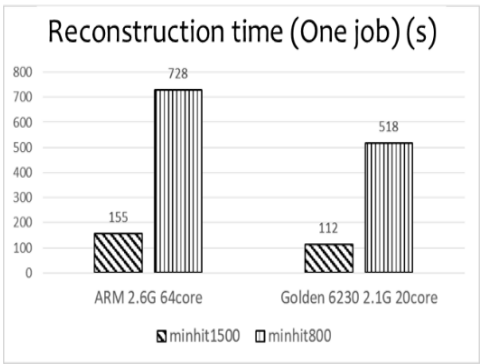

Fig.4 Computing time of single LHAASO-WCDA case reconstruction job

From the figure, we can see that the ARM single core (ARM 2.6G 64core) runs about 40% slower

than the Intel X86 (Gold 6230@2.1G) in different *minhit*. But for the entire server, the Taishan 200K (128 cores) is 320% of the CPU core number of Intel Golden 6230 (40 cores). Since the LHAASO reconstruction process has good data parallelism, Taishan 200K can complete more jobs at the same time.

Then, we tested 5,000 Corsika jobs and compared the Kunpeng 920 to a better X86 CPU (Intel 6240R@2.4 GHz). As You can see in table 1, it took less time for X86 to complete the same number of jobs, but more servers were needed. If the number of servers is equal, the ARM will perform better.

Table 1. Run time of 5000 Corsika jobs

Node Type	Num. of Jobs	Num. of servers	Job Run Time (s)
X86	5000	209	982.073
ARM	5000	**105 (199%↑)**	**1661.432（169.2%↓）**

3.2.3 LQCD testing and optimization

LQCD (Lattice QCD) is a non-perturbative method for solving QCD from the first principles. It does not have any additional parameters other than the standard model. So its computational results are considered to be a reliable description of strong interaction phenomena and are of great importance for the study of QCD theory. LQCD is a typical computationally intensive application in the field of theoretical physics. We benchmarked LQCD applications on a Kunpeng 920 7260 @ 2.60 GHz and an Intel Xeon Gold 6248R CPU @ 3.00 GHz. The results are as follows.

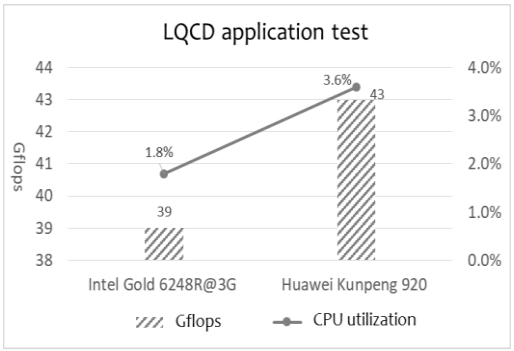

Fig.5 LQCD calculation speed and CPU utilization

For the entire server, on the one hand, X86 computes at 39Gflops and the arm computes at 43Gflops, with the arm being about 10% higher than X86. On the other hand, the CPU utilization of the arm is twice that of X86, because the single X86 CPU core has higher performance.

For LQCD computing tasks, higher memory bandwidth means faster data processing. So, we next tested the difference between the two server memory bandwidths. Our test tool was STREAM, and we configured both the ARM and X86 servers with the same 256GB DDR4 memory, which has a data transfer rate of 2933MB per second. The test result shows that Huawei Kunpeng ARM processors have more than 20% higher memory bandwidth than Intel Xeon Gold 6248R processors.

Table 2. STREAM memory bandwidth test results

Num. of processes	Memory bandwidth (MB/s)			
	Copy	Scale	Add	Triad
48 (Intel)	201 232.0	201 855.5	221 104.4	220 547.3
128 (HW)	264 284.0	264 468.4	273 077.8	274 061.3

We also perform some optimizations of LQCD based on the Taishan 200K server. Since the LQCD computation requires solving a large number of sparse linear equations Dx=b, we can speed up the computation by optimizing the MPI communication process. Our first optimization is to optimize the distribution of parallel MPI processes. Secondly, we changed some compilation parameters, for example, "Allreduce" was selected as "Segmented Ring" and "bcast" was selected as "Binary Tree". Finally, we replaced the standard C math library *libm* with the Kunpeng math library *kml*. By comparing the computation time before and after optimization, you can see that the overall performance is improved by about 7%.

Table 3. Compile instructions and program runtime

gcc, openmpi -O3 -march=native, -np 64 -geom 1 1 4 16				
	total_time(s)	total apply/invapply (s)		
1	1991.25163	179.62093	6.632009	186.252939
2	1991.37653	181.45771	6.684231	188.141941
avg	1991.31408	180.53932	6.658120	187.197440
gcc, openmpi -O3 **-Ofast** -march=native **kml**, -np 64 **-x UCX_TLS=sm --map-by core -geom 4 4 4 1 allreduce=1, bcast=5**				
	total_time(s)	total apply/invapply (s)		
1	1855.82170	162.17470	6.048500	168.223200
2	1855.76260	166.85740	6.044250	172.901650
avg	1855.79215	164.51605	6.046375	170.562425
improvement	**7.30%**	**9.74%**	**10.12%**	**9.75%**

Another optimization is the porting of multigrid to the ARM architecture, which is the cutting-edge algorithm for solving sparse linear equations in the LQCD domain and can be used to improve the Arithmetic Intensity (AI) of LQCD. After this work, we compare the scalability of chroma when solving sparse linear equations, which can indicate the level of server parallelism performance. From the following figure, we can see that Arm is better than X86 in terms of both weak and strong scalability.

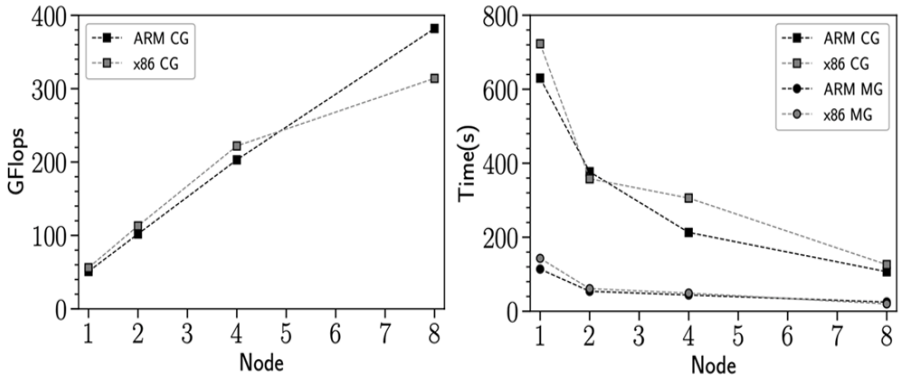

Fig.6 Chroma weak & strong scaling comparison

Then we compare multigrid with the traditional calculation method BiCGStab. We can see that the ARM-based multigrid architecture increases the computational speed by 3.6 times.

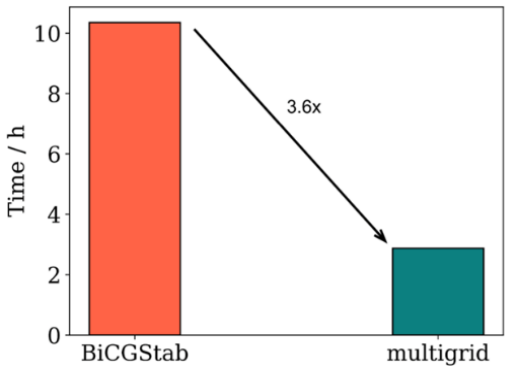

Fig.7 ARM-based multigrid and BiCGStab runtime comparison

4 Summary

We have built the largest arm computing clusters in Dongguan, Guangdong Province, China. On this

cluster, we completed some basic software porting and provided the computing environment for experiments such as LHAASO. Finally, we tested and compared the performance difference between Huawei Taishan servers and some X86 servers, and the results show that the single core of the X86-based CPU has higher performance, but for the entire server, ARM is more advantageous.

In the future, we will plan to conduct multi-ARM node parallel application testing. We will also continue to improve the system to enhance stability and support more experiments.

5 Acknowledgments

This work is supported by the Science and Technology Innovation Project of Institute of High Energy Physics, Chinese Academy of Sciences (No.E15451U2). This work is partly supported by the National Natural Science Foundation of China (No.12075268). Yujiang Bi is partly supported by National Natural Science Foundation of China (No. 12075253).

References

[1] S.M. Smith, J. Price, T. Deakin, A. Poenaru, *A performance analysis of the first eneration of HPC-optimized Arm processors. CCPE*, **31** (2019)

[2] T. Yoshida, *Fujitsu high performance CPU for the Post-K Computer. HCS* (2018)

[3] https://aws.amazon.com/ec2/instance-types/a1/

[4] M. Marek, G. Ganis, P. Mato, I. Razumov, *Porting the LCG software stack to the ARM architecture.* September 2019, https://cds.cern.ch/record/2690233

[5] L. Promberger, M. Clemencic, B. Couturier, A.B. Iartza, N. Neufeld, *Porting the LHCb Stack from x86 (Intel) to aarch64 (ARM) and ppc64le (PowerPC). EPJ,* **214**, 05016 (2019)

[6] Cheng Y, Bi Y, et al. *Porting the EOS from X86 (Intel) to aarch64 (ARM) architecture[J]. The European Physical Journal Conferences*, 2021, 251:02040.