Final Architecture Proposal - NTHU

Hardware Configuration

Item	Configuration	Power Consumption Per Item	Quantity
CPU[1]	Intel Xeon Gold 6148 2.4 GHz 20 cores	150 W	2 per node
	Intel Xeon Platinum 8176 2.1 GHz 26 cores	165 W	2 per node
Memory	32 GB 2666 MHz	7.5 W	12 per node
Storage	Intel SSD DC S3500 240 GB	active: 2.6 W idle: 0.9 W	2 per node
	Micron Crucial MX500 500GB[2]	< 4.5 W	7
	NVMe SSD 2TB	active: 12 W idle: 5 W	1
GPU[3]	NVIDIA Tesla V100 PCIe 32GB	max: 250 W idle: ~25 W	8
	NVIDIA Tesla V100 SXM2 32GB	max: 300 W idle: ~40 W	8
Infiniband Switch	Mellanox EDR	100 W	1
Infiniband NIC	EDR NIC	13.76 W	1 per node
Ethernet Switch[4]	QuantaMesh T4048-IX8D	max 378 W	1

The table above describes the hardware configuration of our HPC cluster system, including each component's power consumption. Some components in the above table might not be used during contest (to meet the 3000 W limit), and these decisions will be base on each application's benchmark result and cluster system stability.

Our cluster will contain two types of machines:

1. QCT QuantaGrid D52G-4U[5] (GPU node)

[1] Final choice on CPU model will be made before contest begin. We will use either 6148 or 8176 as our CPU. The choice will depend on how the applications perform after our optimization.

[2] For IO-500 benchmark.

[3] Final choice on GPU type will be made before contest begin. We will use either SXM2 or PCIe as our GPU. The choice will depend on how well we can port applications to GPU.

[4] Depends on the result in testing phase before the contest, this component might be removed for power saving.

[5] https://www.qct.io/product/index/Server/rackmount-server/GPGPU-Xeon-Phi/QuantaGrid-D52G-4U

2. QCT QuantaPlex T42S-2U[6] (CPU node)

Tesla V100 GPUs provide strong computation power, but it is often affected by CPU-GPU, inter-process and even inter-node communication. To reduce communication overhead, we decide to place GPUs on certain nodes only. That's why we use two types of machines: one with many GPUs, the other without any GPU.

We choose **Xeon Gold 6148** and **Xeon Platinum 8176** as our CPU, since they both have good single-core and multi-core performance. They also support latest powerful instructions, including AVX2 and AVX512.

- Xeon Gold 6148[7]
    - single-core: up to 3.7 GHz
    - multi-core: up to 3.1 GHz on 20 cores
- Xeon Platinum 8176[8]
    - single-core: up to 3.8 GHz
    - multi-core: up to 2.8 GHz on 26 cores

For storage system, we use **SSDs** to provide high I/O performance and saving power at the same time. On the other hand, we will build software-based RAID with multiple SSDs to provide high bandwidth for the newly introduced IO-500 benchmark.

Although the maximum power consumption of our cluster will exceed 3000 Watt power limit, we have several strategies to prevent from go over the limit.

- **HPL consumes high GPU power**, thus we will underclock our CPUs during HPL runs. Also we found that lowering the GPU frequency will give a *higher per-watt-performance*. With these strategies, we can stay under the 3000 Watt limit.
- Remaining benchmarks and applications do not consume high GPU power, so we can run applications with usual CPU and GPU configurations.
- For each application, we measure the per-watt-performance, and sometimes we can *lower the CPU and GPU frequency* to make power consumption lower.

Software Configuration

System Environment

Operating System	CentOS 7.6_1810 x86_64
Compiler	GNU GCC 4.8.5 / 7.4 / 8.3 Intel Parallel Studio 2018 / 2016
CUDA	Driver: 410.48 CUDA 10.0
Infiniband driver	MLNX_OFED 4.6-1.0.1.1

[6] https://www.qct.io/product/index/Server/rackmount-server/Multi-node-Server/QuantaPlex-T42S-2U-4Node
[7] https://en.wikichip.org/wiki/intel/xeon_gold/6148
[8] https://en.wikichip.org/wiki/intel/xeon_platinum/8176

MPI	OpenMPI 3.1.4 Intel MPI 2018 / 2016 MVAPICH2 2.3
Profiling Tools	Intel Vtune Amplifier 2018 nvprof / nvvp
Miscellaneous	clustershell 1.8.1 ipmitool 2.0.23

We choose CentOS as our operating system with the following reason:

- **Stable, reliable packages**
  CentOS provides a massive number of high-quality and stable packages in its official repositories. Using a reliable system allows us to benefit from performance improvements introduced in stable versions.
- **Easy to use package manager**
  CentOS features an easy-to-use package manager: yum. With yum, we can install packages, track dependencies with a single command. Also, when the official packages do not fit our needs, we can use the rpm utility to build packages compatible with yum, making everything integrate well.

Second, we use clustershell to manage cluster nodes, reducing the time cost if we want to run commands on multiple nodes at the same time. For server management, we use IPMI to control some hardware behaviors, including fan speed control and power.

Last, we use CUDA to fully utilize our NVIDIA GPUs. We not only use CUDA to run HPL/HPCG benchmarks, but also develop the GPU version of VPIC using CUDA toolkits.

Benchmarks & Applications

HPL / HPCG	NVIDIA HPL 2.1 HPCG 3.1 GNU GCC 4.8.5 CUDA 10.0 Intel MKL OpenMPI 3.1.4
VPIC	Intel Compiler 2018 update 4 Intel MPI 2018 update 4
SST	GNU GCC 4.8.5 Intel MPI 2018 update 4 Zoltan 3.83 ParMETIS 4.0.3 SCOTCH 5.1
Normal Modes	Intel MPI 2018 update 4 ParMETIS 4.0.3

Conclusion

Both of our hardware and our software are carefully picked for our applications. With dynamic clock adjustment, we get to fully-utilize the 3000-Watts in all applications. Our software configuration allows us to spend less time on setup and management, and instead focus on application optimization. With these effort, we believe we can have great performance this year.