ASC Student Supercomputer Challenge (2019)

Preliminary Contest Notifications

Dear ASC Teams,

Thank you very much for participating in the ASC Students Supercomputer Challenge 2019 (ASC19). This document will provide detailed information about the preliminary round of the contest.

About the Preliminary Round

In the preliminary round, each team is required to submit a set of documents that include a proposal, optimized source code files and output files (detailed requirements specified in Appendix A). Furthermore, the proposal should be written in English, and will be reviewed by the ASC evaluation committee.

Submission Guidelines

All teams should submit their assignment make their submissions to info@asc-events.org before 8:00 AM, March 3th, 2019(UTC/GMT +8:00). A confirmation letter for your submission will be sent to you by email. The submission should include the following items:

a) The proposal file (in .docx or .pdf format), named with the university or college name and the contact person name (e.g. AAAUniversity_BBB.docx).

b) The additional files should be compressed into one file (e.g. AAAUniversity_BBB.zip, other compression formats are also OK). The compressed file should at least include 4 folders (detailed requirements specified in Appendix A):

- Output files of HPL
- Output files of HPCG
- Required files of Single Image Super Resolution Challenge
- Required files of CESM tests

For any further inquiries, please contact the ASC committee via:

Technical Support: techSupport@asc-events.org
General Information: info@asc-events.org
Press: media@asc-events.org

Wish you the best of luck at ASC19.

Sincerely,
ASC19 Committee

Appendix A: Proposal Requirements

I. Brief introduction of the university's or the department's supercomputing activities (5 points)

1. Supercomputing-related hardware and software platforms
2. Supercomputing-related courses, trainings, and interest groups
3. Supercomputing-related research and applications
4. A brief description of the key achievements on supercomputing research (no more than 2 items)

II. Team introduction (5 points)

1. Brief description of the building process of your team
2. Brief introduction of each team member (including group photos of the team)
3. Team slogan.

III. Technical proposal requirements (90 points)

1. Design of HPC system (15 points)

a) Within the 3,000-watt power budget, the system should be designed to achieve the best computing performance.

b) Specify the system's software and hardware configuration and interconnection. Describe the power consumption, evaluate the performance, and analyze the advantages and disadvantages of your proposed architecture.

c) Your system should be based on the Inspur NF5280M5 server. The components and the specific power consumption listed in the table below are for your reference when design the system.

Item	Name	Configuration
Server	Inspur NF5280M5	CPU: Intel Xeon Gold 6132 x 2, 2.6GHz, 14 cores Memory: 16G x 12, DDR4, 2666Mhz Hard disk: 1T SATA x 1 *Power consumption estimation:* *6132 TDP 140W, memory 7.5W, hard disk 10W*
HCA card	FDR	InfiniBand Mellanox ConnectX®-3 HCA card, single port QSFP, FDR IB *Power consumption estimation: 9W*
Switch	GbE Switch	10/100/1000Mb/s, 24 ports Ethernet switch *Power consumption estimation: 30W*
	FDR-IB switch	SwitchX™ FDR InfiniBand switch, 36 QSFP port *Power consumption estimation: 130W*

Cable	Gigabit CAT6 cables	CAT6 copper cable, blue, 3m
	InfiniBand cable	InfiniBand FDR optical fiber cable, QSFP port, cooperating with the InfiniBand switch for use

2. HPL and HPCG (15 points)

The proposal should include descriptions of the software environment (operating system, complier, math library, MPI software, software version, etc.), the testing method, performance optimization methods, performance estimation, problem and solution analysis, etc. In-depth analysis on HPL and HPCG algorithm and the source code is would be a plus.

Download the HPL software at http://www.netlib.org/benchmark/hpl/.
Download the HPCG software at https://github.com/hpcg-benchmark/hpcg

It is recommended to run verification and optimization of HPL and HPCG on x86 Xeon CPU or Tesla GPU platforms. However, teams that have to use other hardware platforms are welcomed to submit their analysis and results.

3. Single Image Super Resolution Challenge (30 points)

Task

Single image super-resolution (SR), a very attractive research topic over the last two decades, is a highly challenging task that estimating a high-resolution (HR) image from its low-resolution (LR) counterpart. It has found practical applications in many real-world problems in different fields, from satellite and aerial imaging to medical image processing, to facial image analysis, text image analysis, sign and number plates reading, and biometrics recognition [1].

Nowadays, following the renaissance of deep learning, there are promising researches on using deep convolutional neural networks (CNN) to perform super-resolution [2]. One of the ultimate goals in super-resolution is to produce outputs with high visual quality, as perceived by human observers. However, current state-of-the-art (SOTA) optimization-based super-resolution methods are largely focused on minimizing the mean squared reconstruction error and have high peak signal-to-noise ratios (PSNR), but they are often lacking high-frequency details and are perceptually unsatisfying in the sense that they fail to match the fidelity expected at the higher resolution [3]. In order to overcome this weakness, generative adversarial network (GAN) is introduced to SR[3] to encourage the network favor solutions that look more like natural images.

In this competition, the participant should design an algorithm using SOTA strategies like deep learning to do the 4x SR upscaling for images which were down-sampled with a bicubic kernel. For instance, the resolution of a 400x600 image after 4x upscaling is 1600x2400. The evaluation will be done in a perceptual-quality aware manner. The perceptual index (PI) defined in pirm2018 [4] will be used to calculate the quality of the reconstructed high-resolution images. Lower PI means higher quality of the reconstructed image. *Ma* and *NIQE* are two no-reference image quality measures [5-6].

$$Perceptual\ index = \frac{1}{2}\left((10 - Ma) + NIQE\right)$$

Dataset and code

- Please use the following links to download the dataset and code.
  Microsoft OneDrive: https://1drv.ms/f/s!AnPwwYkVpK6HhRohl-gY_9YK3VWP
  or
  Baidu SkyDrive: https://pan.baidu.com/s/19qBbUcMUiX2kc3ORXK9vpA Password: o571
  (Data on the two websites are the same. You can choose either of them to download the dataset and source code.)

- In the preliminary competition, 80 LR images (png format) is supplied in *ASC19_03/images* directory for testing, these images were prepared by the committee and never appears on the public websites. The corresponding HR images is reserved by the committee for score computing. An example of LR/HR image is illustrated in Figure 1.

- The committee wouldn't supply any training/validation dataset, the participants can prepare their own dataset for network training and validation.

- Scripts used for calculating PI score and Root-Mean-Square Error (RMSE) are supplied in *ASC19_03/scripts/score* directory. Following the *README* to setup the scripts for validation.

- Script used for 4x down-sampling is supplied in *ASC19_03/scripts/down-sampling* directory.

- A specific training/validation dataset and testing dataset will be supplied for the challenge in the finals.

Figure 1: (upper), original HR image; (lower) 4x down-sample LR image.

Result Submission

A: Each team should submit the 80 reconstructed high-resolution images for scoring tests.

B: For both preliminary and final phases, each team should also submit a folder that contains source code and model that could reproduce the test results.

The folder structure should like:

Folder Name	Contents
Single Image Super Resolution Challenge	Root directory
HR_images	reconstructed high-resolution images
script	PyTorch source code here
model	PyTorch model here

Note:

1. Each team should submit only ONE test results file and corresponding source code of model.
2. The committee may run the submitted code to ensure the legitimacy and effectiveness of the submitted test results.
3. The committee will ignore any submission that doesn't adhere to the rules. Please do check your result file with the scoring script before submission.

Metric

The score is calculated based on the formula below:

$$\text{score} = S_{PI} + S_{prop}$$

Define the minimum valid PI among all participants as PI_{min}, S_{PI} is calculated based on the formula below:

$$S_{PI} = \frac{20}{(PI/PI_{\min})^4}$$

The participants should note that the *RMSE* should locate in the range $8 < RMSE < 18$, otherwise the S_{PI} score will be zero.

S_{prop} is the score given by the committee based on the proposal of the participant. The maximum of S_{prop} is 10. The participants are encouraged to describe their neural network design and the performance of their network in detail.

Training Framework and baseline code

The participants must use PyTorch framework (https://pytorch.org/) for this task. The submission using or depending on any other deep learning framework will be forfeited.

The committee do not supply any baseline code for this task, the participants should design their deep learning network based on public resources. The participants should also consider the training performance of their neural network. Using distributed training strategies like data-parallelism and model-parallelism to accelerate the training process is encouraged.

Reference

[1]. K. Nasrollahi and T. B. Moeslund. Super-resolution: A comprehensive survey. In Machine Vision and Applications, volume 25, pages 1423–1468. 2014.
[2]. Johnson, Justin; Alahi, Alexandre; Fei-Fei, Li (2016-03-26). "Perceptual Losses for Real-Time Style Transfer and Super-Resolution". arXiv:1603.08155
[3]. Ledig, C., Theis,L., Huszar,F., Caballero,J., Cunningham,A.,Acosta,A., Aitken, A.P., Tejani, A., Totz, J., Wang, Z., et al.: Photo-realistic single image super- resolution using a generative adversarial network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4681–4690 (2017)
[4]. https://www.pirm2018.org/PIRM-SR.html
[5]. A. Mittal, R. Soundararajan and A. C. Bovik, "Making a “Completely Blind” Image Quality Analyzer," in IEEE Signal Processing Letters, vol. 20, no. 3, pp. 209-212, March 2013.
[6]. Ma C , Yang C Y , Yang X , et al. Learning a No-Reference Quality Metric for Single-Image Super-Resolution[J]. Computer Vision & Image Understanding, 2016.

4. The CESM Test (30 points)

Application background

Among the models that are widely used in the climate modeling community, Community Earth System Model (CESM) has become one of the most popular climate models in the world and is widely used for various studies of climate variability, climate prediction, and climate change.

CESM is a fully-coupled, community, global climate model that provides state-of-the-art computer simulations of the Earth's past, present, and future climate states. The CESM is one of about a dozen climate models worldwide that can be used to simulate the many components of Earth's climate system, including the oceans, atmosphere, sea ice, and land cover. Using the CESM, researchers can now simulate the interaction of marine ecosystems with greenhouse gases; the climatic influence of ozone, dust, and other atmospheric chemicals; the cycling of carbon through the atmosphere, oceans, and land surfaces; and the influence of greenhouse gases on the upper atmosphere.

The mathematical principle and algorithm for each components of CESM have been fully described in References [1] and [2]. We recommend using the stable version of cesm1.2.2 and its source code is available in Reference [3]. More information about installation and usage of CESM can be found from Reference [4].

References:

[1]. CAM4 description:
http://www.cesm.ucar.edu/models/ccsm4.0/cam/docs/description/cam4_desc.pdf
[2]. CPL document:
http://www.cesm.ucar.edu/models/cesm1.2/cpl7/
[3]. CESM source code: https://svn-ccsm-models.cgd.ucar.edu/cesm1/release_tags/cesm1_2_2/
[4]. CESM usersguide:
http://www.cesm.ucar.edu/models/cesm1.2/cesm/doc/usersguide/book1.html

Introduction to the test command

In this challenge, all participants are encouraged to complete the computation, obtain the right rights and make efforts to reduce the computational needs, both in time and resources. The proposal document should include descriptions of the software environment (operating system, complier, math library, MPI software, netcdf library version and CESM version, etc.), the testing method, performance optimization methods, performance estimation, problem and solution analysis, etc. In-depth analysis into CESM's algorithm and source code is highly encouraged. The detailed tasks and requirements of this challenge are listed below.

Compile and install CESM and run the program against the given data according to the instructions from EXP 1 to 2. Please submit files generated at the final iteration, screen output of each step and command-line of each step, which should be compressed into a tar.gz file (See the table below).

Folder Name	Compressed file name	Contents
CESM		Root directory
Machine file	Machine.tar.gz	env_mach_specific, mkbatch, config_compilers.xml

		config_machine.xml
EXP 1	Exp1.tar.gz	DOUT_S_ROOT/…/logs env_run.xml env_mach_pes Casestatus Macros Command line file(*.sh) Screen output(*.log) Evaluation data
EXP 2	Exp2.tar.gz	DOUT_S_ROOT/…/logs env_run.xml env_mach_pes Casestatus Macros Command line file(*.sh) Screen output(*.log) Evaluation data

1. The requested CESM files should be uploaded into where the ASC19 Committee specified (TBD, which will be informed via E-mail later), and you should copy the download links with MD5 codes of upload files into your proposal (with password if needed).

2. Submit a description and summary into the proposal with what kind of computational resources (configuration and architecture) you use to perform the computation, and how long it takes for each step (submission of a log file is encouraged). You may also describe how you compile the package and whether you perform some modifications of the code, how and why.

3. Describe the strategies you consider (or have performed) that can reduce the computational needs of this kind of computational challenges.

4. Note that each parameter with its value we provided is necessary for ensuring the consistency of results. Other parameters which may influence application performance, you can use them according your platform, we don't show them in this instruction.

5. Since the model output data is large, the evaluation data can choose five variables from the model output data and you should extract them to the specific file for verification. For example, 0001-01.nc should contain the five variables of January 0001, and 0001-02.nc should contains the values of the five variables in February 0002, the same as the others, up to 0010-12. The five evaluation variable can be the U10,TS, PS, Z3,SST.

In order to Compile and install CESM, you may refer to the following steps:

1) Download the CESM source code. You can see the CESM User's Guide, section "Downloading CESM" for complete information on correctly getting the code. Before download the source code, you may need a svn tool.
   We recommend that you could use the version of cesm1_2_2. If you couldn't download the pio and genf90 successfully, you could download them form the following websites:

Microsoft OneDrive: https://1drv.ms/f/s!AnPwwYkVpK6HhRohl-gY_9YK3VWP
or
Baidu SkyDrive: https://pan.baidu.com/s/19qBbUcMUiX2kc3ORXK9vpA Password: o571
(Data on the two websites are the same. You can choose anyone you want to download.)

2) You should port the CESM to your personal system with the Chapter 5 of the CESM User's Guide. You may create your personal file, like the env_mach_specific, the mkbatch, config_compilers.xml config_machine.xml

3) Then you could create the case, and then compiling and running the model.

We need you to create the following case, compile and install the following case successfully. The run length can be controlled with STOP_N and STOP_DATE in the env_run.xml. You can modify this to ensure the run length be ten years. Note that you can use "RESUBMIT" to submit tasks multiple times.

EXP1 : The compset is E1850CN and the resolution is 0.47x0.63_gx1v6.

- To create the case that compset is E1850CN and the resolution is 0.47x0.63_gx1v6.

./create_newcase –res 0.47x0.63_gx1v6 –compset E1850CN –case $CASE –machine $YOURMACHINENAME

- Entering your $CASE directory and then running the script

  ./cesm_setup and ./$CASE.build

- Before you run $CASE.run, you should precise the data name of the ocean model- "DOCN_SOM_FILENAME"

./xmlchange -file env_run.xml -id DOCN_SOM_FILENAME -val pop_frc.gx1v6.091112.nc

- then you will run the script ./$CASE.submit

After the CESM is successfully running, you could find the log and hist file in you output file DOUT_S_ROOT which you are setting.

The Case name should be b.e12.E1850CN. 0.47x0.63_gx1v6.001

EXP2: The compset is B1850CN and the resolution is 0.47x0.63_gx1v6.

- To create the case that compset is B and the resolution is 0.47x0.63_gx1v6.

./create_newcase –res 0.47x0.63_gx1v6 –compset B –case $CASE –machine $YOURMACHINENAME

- Entering your $CASE directory and then run the script

  ./cesm_setup and ./$CASE.build

- then you will run the script ./$CASE.submit

After the CESM was successfully running, you could find the log and hist file in you output file DOUT_S_ROOT which you are setting.

The Case name should be b.e12.B. 0.47x0.63_gx1v6.001

The input data of the case you could download from the website https://svn-ccsm-inputdata.cgd.ucar.edu/trunk/inputdata/ manually or use the svn to download the input data automatically when you run the $CASE.build script after creating the case. The username and password for the website is guestuser and friendly separately.

Note that if you run the CESM model successfully, you should use the netcdf libraries, we recommend that you could use the netcdf version (4.3.3.1 or later), which the netcdf fortran and the netcdf C is installed separately. You could download from the website https://www.unidata.ucar.edu/downloads/netcdf/index.jsp. And install the Software reference the website https://www.unidata.ucar.edu/software/netcdf/docs/getting_and_building_netcdf.html.

c) The Evaluation

We may use the diagnostic package of the CESM and the RMSE to evaluate the model result.

The diagnostic package of the CESM can obtain the climatology and the annual variability of each variable.

In our evaluation, we use the 500hPa geopotential height as a representative. And then calculated the RMSE between your model result and the observation data.

$$\text{RMSE} = \sqrt{\frac{\sum_{t=1}^{n}\left(x_{1,t} - x_{2,t}\right)^2}{n}}$$

- $x_{1,t}$ is the time series of your model result, while $x_{2,t}$ is your prediction moment of the observation data.

- RMSE reflects the deviation of the predicted data from the true value, RMES smaller, while higher the prediction accuracy.

- refer to https://en.wikipedia.org/wiki/Root-mean-square_deviation for details

Note: We have professional staff to do this evaluation. This part is only to introduce the evaluation method, because the diagnostic package is related to several atmosphere software such as netcdf, ncl and nco, et al., which is more complex for the students who is not in the profession. This part is not necessary and you don't need to provide any results for this part. It’s for the reference only.

For any further questions, please contact techsupport@asc-events.org