ASC Student Supercomputer Challenge 2017

Preliminary Contest Notification

In the preliminary round, the team is to submit all required documents as below. All files must be written in English.

a) Proposal. The file format must be either .doc or .pdf. The file name must contain both the name of the university and the name of the team advisor, e.g. “AAAUniversity_BBB.doc”**.**

b) All other documents must be compressed into one file, e.g. “AAAUniversity_BBB.zip”. It must include the following documents. Detailed information is specified in Appendix A.

- Output files of HPL
- Log files of MASNUM_WAVE
- Optimized source code and log files of the traffic prediction program

The login information will be emailed to each team by January 13. All proposals must be submitted to info@asc-events.org by March 6, 8:00AM (UTC/GMT +8:00). No submission will be accepted thereafter. The proposal can be submitted only once. A confirmation will be emailed to the team advisor.

The KNL remote login platform and Sunway TaihuLight platform will be open to all participants from January 13. The KNL remote login platform will be shut down from January 26 to February 3 during China's Spring Festival. For technical support, please contact techsupport@asc-events.org.

Appendix A. Proposal Requirements

I. A brief background description 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 detailed description of the key achievements on supercomputing research (no more than 2 items), attached with proof materials (published papers, award certificates, etc.)

II. Introduction of the team (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. Your team slogan

III. Technical proposal requirements (90 points)

1. HPC system design (15 points)

a) Design a system to achieve the best computing performance under 3,000-watt power consumption.

b) Specify your 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 NF5280M4 server. The components listed in the table below will be provided by Inspur to the teams that enter the final (The configuration may be changed). Other components (except the server itself) are acceptable, but should be prepared by the teams at their own costs. For example, you can change the number of NF5280M4 servers and accelerators, the type of the hard disk and memory, and even the type of the Ethernet in your proposed configuration.

Item	Name	Configuration

Server	Inspur NF5280M4	CPU: Intel Xeon E5-2660v4 x 2，2.0Ghz，14 cores Memory: 16G x8，DDR4，2133Mhz Hard disk: 1T SATA x 1 *Power consumption estimation:* *E5-2660v4 TDP 105W, 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 test (15 points)

The team will run a single node HPL test on a remote KNL cluster provided by Zhengzhou University. The configuration of KNL cluster can be found in Appendix B.

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

The HPL software can be downloaded at http://www.netlib.org/benchmark/hpl/.

Successful verification and optimization of HPL on KNL platforms are recommended. However, teams that cannot access the platform and have to use their own hardware platforms are encouraged to submit their analysis and results.

3. Test and optimization of the MASNUM_WAVE program on Sunway TaihuLight (30 points)

The application is the surface wave model named MASNUM Wave Numerical Model (MASNUM_WAVE). The teams are required to conduct performance optimization of the

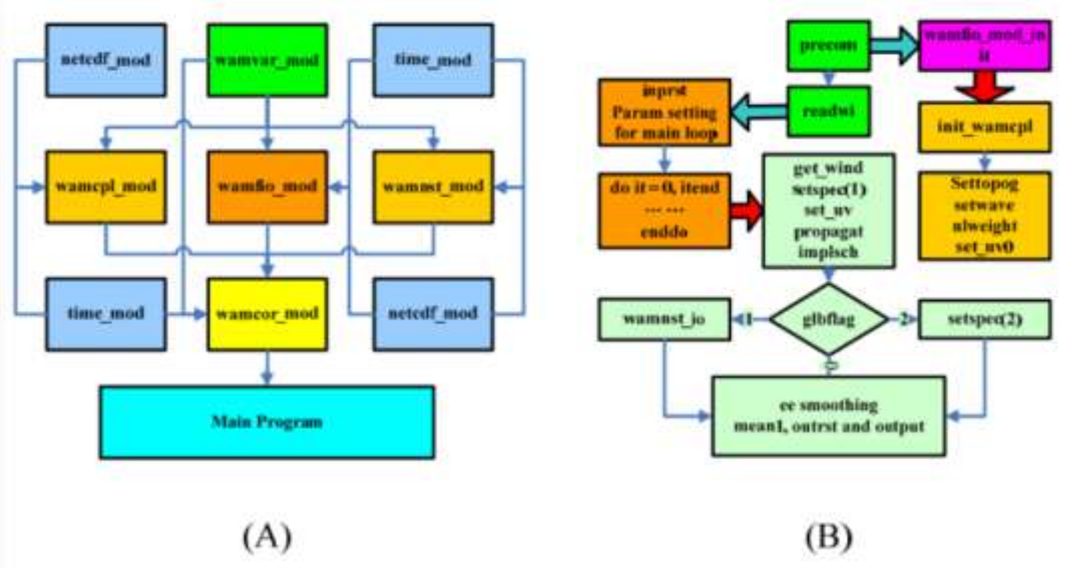
MASNUM_WAVE program on Sunway TaihuLight to verify the correctness and to achieve good performance and efficiency.

Application background:

As one of the most frequent marine disasters, wave has a great impact on marine activities such as marine engineering, military operations and marine science research. The model has been compared with WAM model in typical wind fields and gave concerted results in general sea state and good improvement in high sea state. With the increasing demands on higher forecasting precision and climate research, higher and higher resolution becomes a main stream in wave model development.

The Sunway TaihuLight supercomputer is ranked No. 1 in the TOP500 list. The complete system has a theoretical peak performance of 125.4 Pflop/s and a sustained performance of 93Pflop/s with 10,649,600 cores and 1.31 PB of primary memory. The 2016 Gordon Bell prize finalist "A Highly Effective Global Surface Wave Numerical Simulation with Ultra-High Resolution" was based on the masnum model and Sunway TaihuLight.

This model is arranged by 7 modules, including 5 main modules and 2 separate modules for useful tools. The dependence of these modules is given in Fig. A. The flowchart of the main procedures is given in Fig. B.



(A) (B)

a) Introduction of the application

(1) The entire package includes:

doc	User guide
exp	Examples exp1: west-northern pacific example (99-150E, 0-50N) exp2: global ocean example (0-360E, 72S-72N)
inputdata	Inputdata for wave model forcing: the forcing data such as 10-m wind fields topo: the topography data
source	Source code of wave model bin: makefile

	wave_coe: source code mpi_mod: source code associated with MPI parallelization
validation	Validation data exp1: example 1 validation data (pac_ncep_wave_20090228_standard.nc) exp2: example 2 validation data (global_ncep_wave_20090630_standard.nc)

(2) Compiling and running

- Enter the directory masnum_wave/source/bin
- Modify the necessary parameters in the "makefile" file
- then "make"
- If there's no error messages and "masnum.wam.mpi" is generated, it indicates the compiling successes.

Setting up the example 1 (exp1)

The example is the experiment to simulate the western Pacific (99-150E, 0-50N) ocean surface wave from January 1 and February 28, 2009. The resolution is 1/6 degree and the length of time step is 7.5 minutes.

- Enter directory "masnum_wave/exp/exp1"
- Modify the necessary parameters in the "exp1_run.csh" file
- Modify the execute command according to the job submitted system. In this example, the command is "bsub -I -n 1 -q q_sw_expr -share_size 6000 -host_stack 1024 -b -m 1 -o out.qrunout ./masnum.wam.mpi"

Setting up the example 2 (exp2)

The example is the experiment to simulate the global (0-360E, 72S-72N) ocean surface wave from January 1 and June 30, 2009. The resolution is 1 degree and the length of time step is 15 minutes.

- Enter directory "masnum_wave/exp/exp2"
- Modify the necessary parameters in the "exp2_run.csh" file
- Modify the execute command according to the job submitted system. In this example, the command is "bsub -I -n 1 -q q_sw_expr -share_size 6000 -host_stack 1024 -b -m 1 -o out.qrunout ./masnum.wam.mpi"

The meaning of the parameters in the exp*_run.csh

Directory	Description
needmake	Flag for make the executable file, YES or NO YES: make the executable file NO: do not make the executable file
Nproc	The number of processors for MPI should be set >=2
Masnum_home	The path for wave model home directory

(3) Cross validation

- After finish the example run, compare the variable “hs” in the “pac_ncep_wav_20090228.nc” with it in the “pac_ncep_wav_20090228_standard.nc”.
- Enter the directory masnum_wave/validation/exp*, then
- Modify the necessary parameters in the makefile
- Execute the command: “make”
- If everything is ok and the executable file “compare_exp*” is generated, it indicates the compiling successes.
- Execute the command:“compare_exp*”
- If “Compare Success” is printed on screen, it indicates the validation is pass.

b) Requirements for Preliminary Contest

i. The proposal document shall include the description of the testing software environment (operating systems, compliers, math libraries, MPI software and application software, etc. with version information), the testing methods, performance optimization methods, performance estimation, problem and solution analysis, etc. In-depth analysis into MASNUM_WAVE’s algorithm and source code is highly encouraged.

ii. The MASNUM_WAVE software can be downloaded from: /usr/sw-mpp/apps/src/masnum_wave. As MASNUM_WAVE provides different workloads that demonstrate different performance features, in this contest, the teams should focus on the two MASNUM_WAVE workloads included in the software package.

iii. Teams without the required hardware platforms are also encouraged to submit their thoughts and analysis.

c) Specific notes

i. The path of “netcdf” on the Sunway TaihuLight Platform: /usr/sw-mpp/apps/lib/netcdf

ii. “Sunway TaihuLight Quick Start Guide” and “VPN Instructions” are provided in the mail.

iii. Program download path: /usr/sw-mpp/apps/src/masnum_wave.

4. Deep learning contest: traffic prediction (30 points)

In this contest, the team is to devise an algorithm for predicting conditions in rush hour traffic, the hardware platform is not specified.

Application background:

Nowadays, real-time high-fidelity spatio-temporal data on transportation networks of overwhelming major cities have become available. People can obtain datasets from Loop Detectors, Mobile Phones, Traffic cameras, Self-Driving car. This gold mine of data can be utilized to learn about traffic behavior at different times and locations. Relieving the traffic congestion problem, potentially resulting in major saving in time and fuel. But it is still a challenge for traffic prediction for modeling the spatio-temporal relations.

Data are sampled from mobile phone APP. A traffic network graph at time point below, every node was denoted a link in this small area.

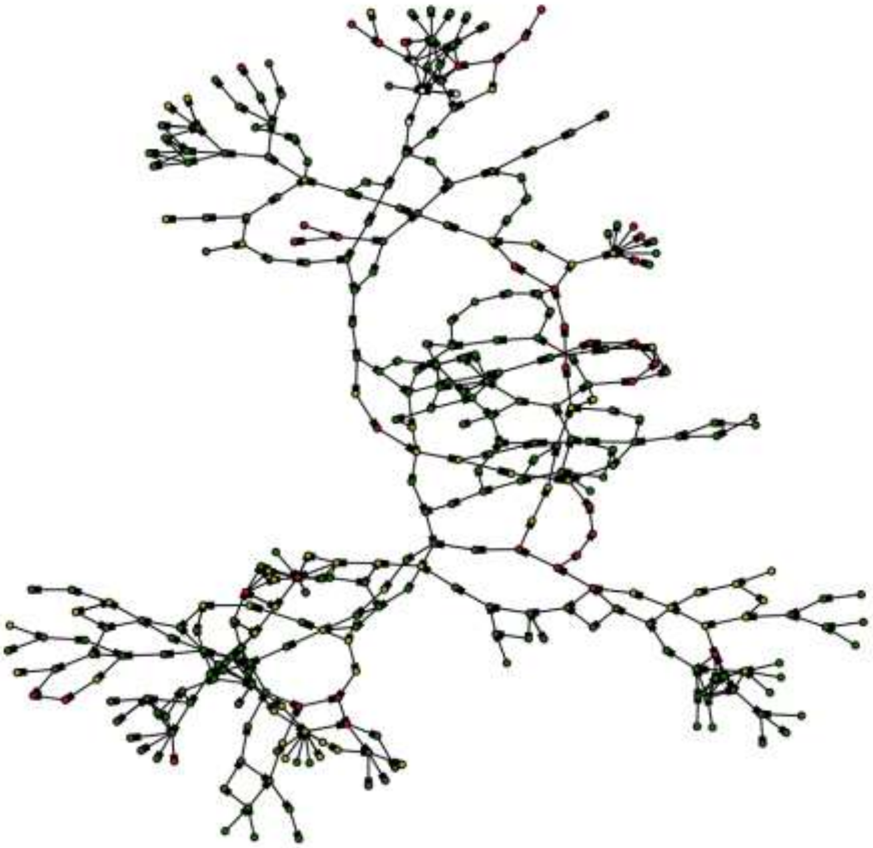


Figure 1. A traffic network graph at time point

Using the dataset which are aggregated about 50 days from 00:00 a.m. on March 1st to 08:00 on April 20th for training. Your task is to predict each link's speed for every 5 minutes from 08:00 to 10:00 on April 20th.

Data

- speeds.csv

This file contains the whole Link ID and time series of simulated congestion measurements from 201603010000 to 201604200800(%Y%m%d%H%M). Every measurement expressed by four states. The value 1 means fluency, 2 is slow, 3 is congested, 4 is extreme congested. The first column is Link ID and the rest are traffic condition which aggregated 5-min interval each from 00:00 a.m. on March 1st to 08:00 a.m. on April 20th.

- graph.csv

This file contains spatio information that forced by traffic network topology. Each row represents a ***directed*** junction which has two parts. The first column is source of link id and the second is destination of link id. That means has a junction between source and destination.

Please see https://github.com/PaddlePaddle/Paddle/blob/develop/demo/traffic_prediction/data/get_ddat.sh to get the data

Evaluation

- In preliminary contest, we only accept once submission.
- Your model outputs are evaluated by the Root Mean Squared Error (RMSE) of predictions.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(X_{actual,i} - X_{model,i})^2}{n}}$$

- $X_{actual,i}$ is actual value of moment i, while $X_{model,i}$ is your prediction moment of i
- 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

Solutions

- Outputs file

  Predictions must be submitted as a csv file that is basically same as speeds.csv file just different time seires that should be from **201604200805** to **201604201000**

- Project file

  Despite outputs file, you have to submit your project and it at least contains these parts:

  a) Trainer config file

  - You are expected to submit the file which contains the configured file of neural network architecture.

  b) Data provider

- This file generate provides data to PaddlePaddle core program

c) Data source path

- The files just contain the local data path.

d) run scripts

- This script is used to train and predict and generate directly outputs

e) Proposal document

- This document at least contains two part which include Summary and Detailed Process

f) README

- The file provides instructions about other files in your directory

Benchmark DEMO

There will be a DEMO based on PaddlePaddle. PaddlePaddle is a deep learning platform open-sourced by Baidu. With PaddlePaddle, you can easily train a classic neural network within a couple lines of configuration, or you can build sophisticated models that provide state-of-the-art performance on difficult learning tasks like sentiment analysis, machine translation, image caption and so on. The DEMO can be accessed from https://github.com/PaddlePaddle/Paddle/tree/develop/demo/traffic_prediction

1. Firstly, You must build and install (http://www.paddlepaddle.org/doc/build/) PaddlePaddle.

2. After installed PaddlePaddle, enter the demo (https://github.com/PaddlePaddle/Paddle/tree/develop/demo/traffic_prediction) directory and run following instructions:

`sh train.sh`

After the end of training, then run:

`sh predict.sh`

when prediction has been done, you'll get the output file named *result.csv*

Benchmark DEMO tips

1. The demo is very simple stuff of multi-task learning, if you want to get lower error. You need to expand your imagination and take advantage of all the data.
2. PaddlePaddle provides both English documents (http://www.paddlepaddle.org/doc/) and Chinese documents (http://www.paddlepaddle.org/doc_cn/)
3. If you have any question, you can also go to github and submit your issue besides sending emails to techsupport@asc-events.org.

Appendix B. The KNL Remote Testing Platform

I. Configuration

The platform consists of 11 nodes: 1 login node, 5 compiling nodes and 5 computing nodes. The detailed configuration is as follows:

- 5 compiling nodes for compiling application and submitting single-node jobs to the computing nodes. These nodes can be accessed through ssh from login node.

Item	Name	Configuration
Server	Inspur NF6248 x 5	CPU：Intel Xeon Phi 7210, 1.3Ghz, 64 cores
		Memory: 16G x4, DDR4, 2133Mhz
		Hard disk: 120G SSD x 1

- A cluster with 5 computing nodes.

Item	Name	Configuration
Server	Inspur NF6248 x 5	CPU：Intel Xeon Phi 7210, 1.3Ghz, 64 cores
		Memory: 16G x4, DDR4, 2133Mhz
		Hard disk: 120G SSD x 1

II. Instructions for using the KNL remote testing platform

1. Login account:

The login IP address and the account and password will be emailed to each team. Users can change the password after the first login. The login account will be closed automatically after the final contest. One account allows at most 2 users to log in simultaneously.

2. Storage space:

Each team will have 20GB storage space by default. Users shall make their own data backup and clear the disk space timely. In case the user needs to use a larger disk space, please contact techsupport@asc-events.org.

3. Usage guidelines

As mentioned above, the compiling nodes will be used for compiling. Single-node test should be performed on the computing nodes by submitting jobs. Detailed description can be found in the “Instruction Manual of the Remote Log-in Platform”.