

INSIDE THIS ISSUE

(3 Pages)

Topic	Page No.
Research Highlight	
Flood Modelling Simulation: A Feasibility Study Using Open Source Code	1
HPC Article Series	
Running MATLAB Programs on ANTya — Part-5: Running MATLAB on Multiple Cores with the Parallel Computing Toolbox	2
ANTYA Updates and News	
HPC Picture of the Month	2
Tip of the Month	
ANTYA Workload Status and Suggestions for Submitting Jobs to Get the Required Computational Resources	3
ANTYA HPC Users' Statistics — APRIL	3
Other Recent Work on HPC (Available in IPR Library)	3

GANANAM (गणनम्)

HIGH PERFORMANCE COMPUTING NEWSLETTER
INSTITUTE FOR PLASMA RESEARCH, INDIA



Flood Modelling Simulation: A Feasibility Study Using Open Source Code

Shailja Tiwari (SO-E, MultiDisciplinary Research Division, IPR)
Email: shailja@ipr.res.in

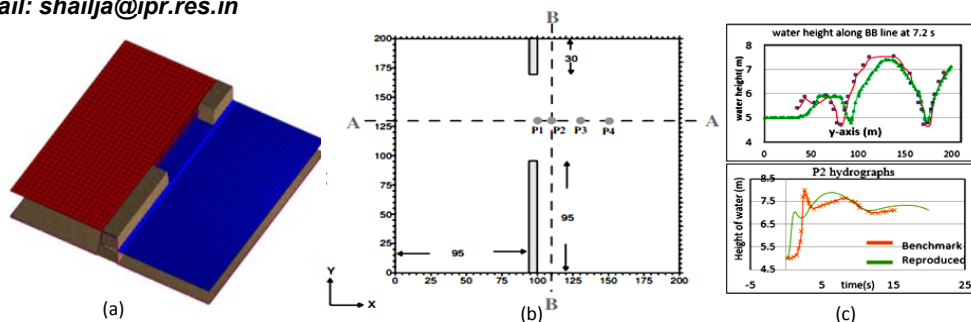


Figure 1: (a) Geometric schematization of dam break over flat bed having dam surface height 10 m (Red), obstacles with 75 m gap in between (Grey) and flat bed blue with water height 5 m (Blue) [4] (b) Points of measurement along AA' and BB' in flat bed [4]. (c) Hydrographs obtained in OpenFOAM w.r.t. benchmark data are within limits considering modeling variation of data used.

Floods are one of the most frequently occurring natural disasters causing the water to flow onto the area of land which is normally dry. There are many causes which can lead to flooding, natural (heavy rainfall, melting of snow, tsunami, cloud burst etc.) as well as contributed by man-made activities (Dam breaks, blockage of underground stormwater drains etc.). Severe rainfall and dam burst processes result in excess water flooding through cities and rural areas and are highly disruptive to daily life activities. Estimation of highly flood-affected zones using real-scale modelling is very important to mitigate flooding risks. It is expected that a good mathematical model [1] for flood estimation should give the spatial and temporal evolutions of the flood event [2], as well as the velocity field apart from identifying the severely affected areas. A database of different runs for several flooding scenarios can be created for a given city (and also repeated for different major cities), showing peak flood levels, and peak velocities at specified observation points. For a situation like a cloud-burst or a dam break, an estimate for water-height (t) and velocity (t) function data at different specified locations downstream from the dam or inside the city can be obtained timely which will be helpful in mitigation and pre-preparedness of these events.

"OpenFOAM is tested and validated against the dam break benchmark problem in HPC ANTya at IPR for checking the feasibility of flood modelling simulations as well the correctness of results"

1-D and 2-D models have been used to model dam-break flooding but there are limitations attached to capturing spatial extent and flow depth data which requires carrying out a 3-D numerical simulation based on Navier Stokes Equations. In this feasibility work, a multiphase approach to find numerical solutions to the Navier-Stokes equation for an incompressible fluid (water) in an open-source computational fluid dynamics platform named OpenFOAM[3] has been used. For validation, a case study is performed for a dam-

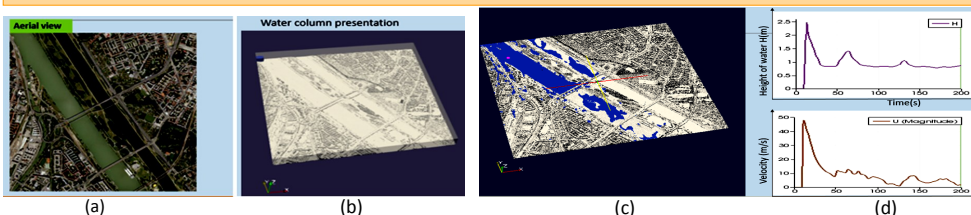


Figure 2: (a) Satellite view of Vienna city's small tile, (b) The simulation geometry used in OpenFOAM. (c) The cross section point in the water column image. (d) Hydrographs obtained at cross section point (300, 2100, 5.1).

break case based on [4] and results are reproduced within limits considering modelling variations of data used. The test consists of simulating the partial collapse of a dam in a spatial domain with a 200m×200m flat region, with a dam in the middle as shown in Figure 1(a). Figure 1(b) shows locations p1, p2, p3 and p4 where data is calculated in terms of water height variation w.r.t. time and space along line AA' and BB' in a 2-D cross-section. At the beginning of the simulation, the water surface level is set at 10 m for the upstream region and 5 m for the downstream one and the unsteady flow is generated by the instantaneous collapse of an asymmetrical 75 m long portion of the barrier. The hydrograph comparison with the benchmark data is within 10-15% variation in water height as shown in Figure 1(c). This study suggests that a multiphase solver in OpenFOAM seems a good choice to perform a 3-D simulation for dam-break in a city which can evaluate flood mapping zones to categorize high, medium and low-risk zones in the urban and rural regions.

To investigate that, the modelling technique similar to [5, 6] is employed which uses Digital Elevation Models (DEM) of city and rainfall data to provide a real-time estimate of water elevation levels at a very low cost compared to various commercial software e.g. TufLOW, HEC-RAS,

MIKE FLOOD etc. A feasibility study has been performed with DEM of 2.5 km² tile area of Vienna city's on a 1:1 scale (<https://www.wien.gv.at/ma41datenviwer/public/>) and tested with an imaginary dam break situation with 100 m water height column as shown in Figure 2(a). The geometry in simulation is divided into 200x200x100 mesh cells and simulation run time was recorded for 50 seconds which took ~2 hrs real-time with 40 cores on ANTya. Paraview is used for animation and hydrograph generation shown in Figure 2(b). Our results show it is possible to obtain real-time hydrographs data in OpenFOAM with DEM data of any terrain during flooding events.

As a next step, methodology to include rainfall and cloudburst events is going for validation of real-time flooding situation in an Indian city.

References:

1. Fennema, R. J. et al., Explicit methods for two dimensional transient free-surface flows, JHE, 116(8), 1013–1034, 1990.
2. Jorgenson, J., et al., Two-dimensional modeling of dam breach flooding, US-China workshop on advanced computational modelling in hydroscience & engineering, 19–21 September, USA, 2004.
3. <https://openfoam.org/>.
4. C. Biscarini, et al., CFD modelling approach for dam break flow studies, Hydrol. Earth Syst. Sci., 14, 705–718, 2010 www.hydrol-earth-syst-sci.net/14/705/2010/.
5. A. I. Kurbanaliev, et al., OpenFOAM Multiphase Solver interFoam for Large Scale Modeling, Proceedings of WCECS 2019, October 22–24, San Francisco, USA.
6. Komeil Samel, et al., Shahar Chay Dam-Break simulation using OpenFOAM, 6th Inter. Conf. on Civil, Architectural & Environmental Sciences, June 19, 2019, Stockholm-Sweden.

Running MATLAB Programs on ANTya

Part-5: Running MATLAB on Multiple Cores with the Parallel Computing Toolbox

This article covers how a user can launch MATLAB on multiple cores in parallel and run MATLAB program faster on ANTya. In earlier issue 2, MATLAB job submission using the batch scheduler has already been demonstrated. It has been observed that most of the users are using only single core for submitting their MATLAB jobs. For the codes having independent tasks, it is possible to use the more compute cores to run them faster using the MATLAB module installed on ANTya.

"MATLAB supports both implicit parallelism (an automatic built-in multithreading) and explicit parallelism with the Parallel Computing Toolbox (PCT)"

"For using PCT, the following three commands can be simply modified: for \rightarrow parfor, feval \rightarrow parfeval and sim \rightarrow parsim"

"Not much programming effort required to run existing MATLAB codes using PCT"

"PCT take advantage of computing cores on ANTya to solve complex problems"

How to Implement in ANTya?

An example MATLAB problem of "Getting eigenvalues of square matrices" has been taken for demonstration. The source code which has been modified to perform the runs on ANTya has been taken from [MATLAB website](#). The code has been executed in serial as well as in parallel on different numbers of cores. The demonstration shown here will only apply for a single compute node parallelization in ANTya.

Eigenvalues of Square Matrices

Serial Code

Parallel Code

```
# Serial code: eigen_serial.m
M=500; % #of rows and columns of each matrix
N=1000; % number of trials
tic; % Elapsed time using tic and toc
a = zeros(N,1); % initialize output vector
for i = 1:N
    a(i) = max(eig(rand(M)));
end
time = toc;
disp(['Processing time: ' num2str
(time)]);
```

```
# Parallel code: eigen_parallel.m
M=500; % #of rows and columns of each matrix
N=1000; % number of trials
tic; % Elapsed time using tic and toc
a = zeros(N,1); % initialize output vector
parfor i = 1:N % Changed for to parfor
    a(i) = max(eig(rand(M)));
end
time = toc;
disp(['Processing time: ' num2str
(time)]);
```

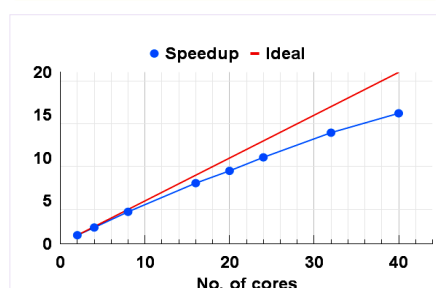
The parallel code has been launched by increasing the no. of cores (2 4 8 16 24 32 40) to see the speedup obtained which shows the parallel processing capability of MATLAB available in ANTya. Following scripts show the running of code on 32 cores. Similar way it can be launched on different no. of cores.

```
# Parallel code: eigen_parallel.m
parpool('local',32); % A local pool with 32 cores
M=500; % #of rows and columns of each matrix
N=1000; % number of trials
tic; % Elapsed time using tic and toc
a = zeros(N,1); % initialize output vector
parfor i = 1:N % Changed for to parfor
    a(i) = max(eig(rand(M)));
end
time = toc;
disp(['Processing time: ' num2str
(time)]);
```

```
# Job script: job_matlab.sh
#!/bin/bash
#PBS -q regularq
#PBS -N eigen_speedup
#PBS -l select=1:ncpus=32
#PBS -l walltime=00:30:00
#PBS -j oe

module load MATLAB/2016a
cd $PBS_O_WORKDIR

##change the input filename as needed
matlab -nodisplay -nosplash -r
"eigen_speedup, exit"
exit 0
```



No. of Cores	Processing Time (s)
2	78.0577
4	41.135
8	20.9388
16	11.0354
32	6.0239
40	5.1283

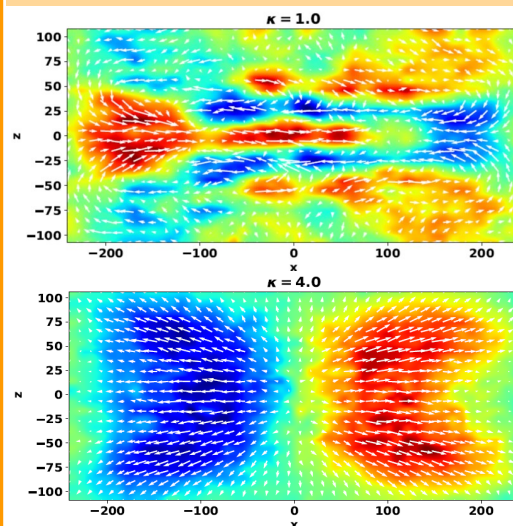
ANTYA UPDATES AND NEWS

1. New Packages/Applications Installed

- \Rightarrow **The no. of Applications stays the same for this issue.**
- \Rightarrow **ANTYA Users /home data monthly backed up cycle completed.**

HPC PICTURE OF THE MONTH

Quadrupolar Like Flow in 3D Yukawa Liquids



Pic Credit: [Suruj Jyoti Kalita](#)

The figure shows the Y averaged V_x velocity contours as a function of X and Z for 60% perturbation strength. A quadrupolar flow is visible for $\kappa(k) = 1.0$ with outward flow in the spanwise z direction and inward flow in the streamwise x direction, while a strong outward flow is seen for the $\kappa = 4.0$. Each rectangular region represents the entire Y averaged simulation domain of $L_x L_z$ 494 X 220. As κ value is changed, the flow structure is bifurcated from one pattern to another. [Ref: S Kalita, R Ganesh, Phys. Fluids 33, 095118 (2021)]

This figure is obtained using an in-house developed "MPMD-3D" code, which is available in both CPU and GPU parallel versions. For the above figure, 4 GPU P100 Tesla cards are used in HPC ANTya with 614304 particles and the wall time is ~ 19.5 hours.

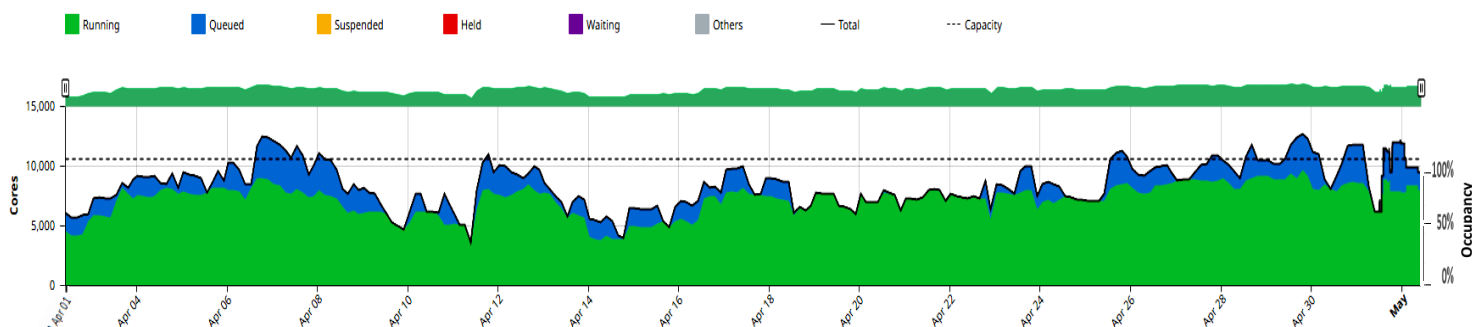
TIP OF THE MONTH

Consider transitioning your CPU codes and GPU codes compiled with Intel modules and pgi modules respectively to oneAPI and Nvidia HPC SDK Suite compilers respectively for getting the updated packages and support.

ANTYA Workload Status and Suggestions for Submitting Jobs to Get the Required Computational Resources

With the commissioning of HPC Cluster ANTYA in 2019 having a total compute performance of 1 PetaFlop (1PF) comprising of 80% CPU and 20% GPU performances, the usage of the system has been increasing very rapidly both for the CPU as well as for GPU resources. The number of HPC Users has jumped to more than 200 in the last 3 years. This has led to an increase in the number of applications running on ANTYA and subsequently we are observing longer wait times for the users to get the computational resources. We are observing more than 95% peak loads consistently for the 9600 CPU cores. It is expected that this workload will not go down too much in the coming months and therefore the users need to plan out their jobs which require large no. of computational cores or GPU cards well in advance. You may face situations where your jobs would remain in queue for more than 24 hours and your requests for additional compute resources during the peak time would not be prioritized.

ANTYA Workload Observed in April 2022



Suggestions for Job Submission

“Do not wait for the cluster load to decrease to submit jobs”

Submit your jobs in queue to get the priority among the queued jobs submitted after you

“Be aware of the ANTYA Job Queuing System (AJQS)”

AJQS details are available in issue10 of [GANANAM](#)

“Check under utilization of the compute resources for running jobs”

Check if your running jobs are using as many cores as you have requested in job scripts

“Make sure to have restart feature in your codes”

Restart feature will enable you to use free cores in queues with shorter walltime

Other Recent Work on HPC (Available in IPR Library)

CFD Analysis of Preheating Stage of the Plasma Processing Chamber and Comparison of Results with the Experimental Data

**HARDIK
GIRISHBHAI MISTRY**

Finite element simulation for validation of multi-dipole line cusp magnetic field configuration for MPD

**AMIT KUMAR
PATEL**

Acknowledgement

The HPC Team, Computer Division IPR, would like to thank all Contributors for the current issue of **GANANAM**.

ANTYA HPC USERS' STATISTICS—

APRIL

♦ Total Successful Jobs — **4299**

♦ Top Users (Cumulative Resources):

- CPU Cores **Amit Singh**
- GPU Cards **Suruj Kalita**
- Walltime **Gayathri**
- Jobs **Promit Moitra**

On Demand Online Tutorial Session on HPC Environment for New Users Available
Please send your request to hpcteam@ipr.res.in.

Join the HPC Users Community
hpcusers@ipr.res.in
If you wish to contribute an article in **GANANAM**, please write to us.

Contact us
HPC Team
Computer Division, IPR
Email: hpcteam@ipr.res.in

Disclaimer: “GANANAM” is IPR’s informal HPC Newsletter to disseminate technical HPC related work performed at IPR from time to time. Responsibility for the correctness of the Scientific Contents including the statements and cited resources lies solely with the Contributors.