ISSUE 41 APRIL 2024

# INSIDE THIS ISSUE
(3 Pages)


| Topic | Page No. |
|---|---|
| **Research Highlight** Simulating the convective pattern formations observed in a Dusty Plasma Experiment | 1 |
| **HPC Article** Part 2 : Unlocking HPC Potential: A Practical Tutorial on Deploying Docker Hub Containers seamlessly with Singularity | 2 |
| **ANTYA Utilization: MARCH 2024** | 3 |
| **ANTYA HPC Users' Statistics — MARCH 2024** | 3 |
| **Other Recent Work on HPC (Available in IPR Library)** | 3 |



# GAṆANAM (गणनम्)

HIGH PERFORMANCE *COMPUTING* NEWSLETTER

INSTITUTE FOR PLASMA RESEARCH, INDIA


## Simulating the convective pattern formations observed in a Dusty Plasma Experiment


**Ankit Dhaka (Research Scholar, Dusty Plasma Section, FPED, IPR)**

***Email: ankit.dhaka@ipr.res.in***

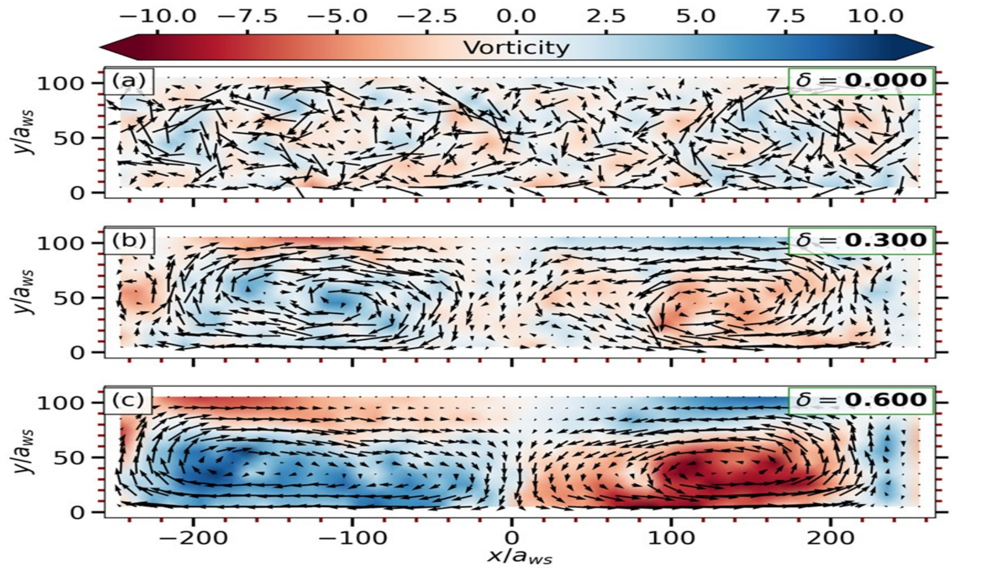




**Fig. 2: The velocity vector field plots from the simulations with different temperature gradients. The arrows indicate the direction of mean velocity in a cell and fill colour denotes the magnitude of vorticity**

Dusty plasma is a system consisting of micron-sized particles in the presence of conventional electron-ion plasma. These particles become negatively charged due to a high flux of electrons as compared to ions. As the bare charge of the dust particles is shielded by the plasma species, they interact with a shielded Coulombic interaction known as the Yukawa or Debye-Hückel potential for the case of un-magnetized plasma. These charged particles levitate in the strong electric field region near the sheath of the plasma and form clouds. The coupling strength between the dust particles is defined by a dimensionless parameter known as the Coulomb Coupling parameter which is defined as the ratio of electrostatic potential energy at mean separation to the average kinetic energy. The coupling strength of the dusty plasma, for cases of the same screening length, can be controlled by judiciously changing the charge and temperature of the dust particles, which depend on the background plasma parameters. A higher coupling strength corresponds to longer spatial correlations of dust particles. Therefore, the phase of a dusty plasma can be changed from fluid to solid by controlling the extent of spatial correlations (shorter to longer). In a recent experiment conducted in the newly commissioned device at IPR, it was observed that two counter-rotating vortex structures were spontaneously formed. Further analysis of experimental data revealed that a vertical temperature gradient is formed in the dust cloud. It was found that the vortex structures formed only when a temperature gradient in the dust cloud was established in the vertical direction. Furthermore, it was also noted that the strength of these patterns increases with an increase in the temperature gradient. This strongly suggests that these vortex structures may arise from the convection phenomenon in the dusty plasma[1]. In order to supplement the experimental observations, a set of molecular dynamics simulations are performed on ANTYA.

The molecular dynamics simulations solve the coupled equations of motion of the particles in one or many force fields using the velocity verlet algorithm. As the solver of the simulations advances the velocities and positions in time, an entire 6N+1 dimensional phase space of the dynamical can be numerically obtained. Therefore, MD simulations have been found useful in obtaining kinetic information about dynamical systems [2],[3]. In the present study, MD simulations are performed to support the experimental observation of convective pattern formations using Large-scale Atomic/Molecular Massively Parallel Simulator (LAMMPS) code [4] with hardware acceleration on ANTYA. Two P100 cards from the GPU compute nodes of ANTYA were utilised to run the simulations for N=16000 particles These simulations are carried out in a rectangular domain of dimension and $L_x$ and $L_y$. The inter-particle force field is set to be of the Yukawa interaction, which is represented by the Debye-Huckel potential. The dust grains are modeled as point-like charged particles with mass $M_D$. The boundary conditions are selected to be reflective along the y direction to mimic the effect of strong electric field of sheath. A confining field is applied along ***x*** direction to prevent the particles from diffusing outward. The number of particles N, temperature T_D, and charge Q_D dictate the Coulomb Coupling parameter (Γ) of the dusty plasma. These simulation parameters are chosen in such a manner that the Coulomb coupling parameter matches its value estimated in the experiments.

> ***Molecular dynamics simulations are performed to study the effect of temperature gradient on convective patterns in bounded dusty plasma as observed in a recently conducted laboratory experiment at IPR.***

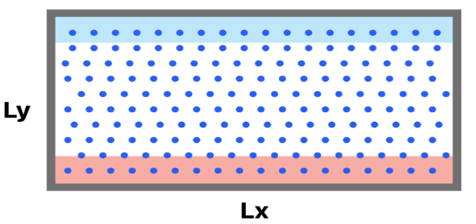



**Fig. 1 : A schematic of the simulation domain. The blue dots represent point-like particles**

In order to simulate the effect of the temperature gradient, two regions, namely hot and cold, are created in the bottom and top part of the simulation box, respectively. In Fig. 1, the hot region is represented by red colour, whereas the cold region is by blue colour. The temperature gradient is induced in the system by maintaining the particles of the red region at a higher temperature ($T_H$) and particles of the blue region at a lower temperature ($T_L$) as compared to the average temperature of the bulk. The process of maintaining a given temperature in a region is done by introducing a drag/source term in the equations of motion of the particles. The simulations are conducted in the manner described below. First, the particles in the simulations domain are equilibrated to achieve a spatially uniform temperature. Once the system achieves a desired temperature (corresponding to the experimentally observed Coupling parameter), the particles in the red and blue regions are taken to the temperature $T_H$ and $T_L$, respectively. This is followed by starting a production run of the simulations, which records the velocities and positions of the particles at regular intervals. The fluid quantities are calculated from the MD trajectories by performing a grid average of $N_x \times N_y$ cells over the simulation domain. Here, $N_x$ and $N_y$ denote the number of cells in the ***x*** and ***y*** directions, respectively

The vorticity field plots from the simulations are plotted in Fig. 2 for three different values of temperature gradient in the system. The parameter **δ** shows the extent of temperature gradient which is defined as $\delta = (T_H - T_L)/(T_D)$ where $(T_D)$ denotes the average temperature of the system . The arrows in the figure show the direction of velocity of the fluid element, whereas the fill colour denotes the magnitude of the local vorticity. In the case where the temperature gradient is absent, the velocities of fluid elements are randomly oriented as expected. Once the magnitude of the temperature gradient is increased, a small cluster of vorticities starts to appear, which clearly indicates the circulation in the velocity field. The circulation emerge in the form of two counter-rotating vortex structures as shown in Fig 2. A further increase in the magnitude of temperature gradient beyond this point results an increase in the vorticity of structures. These results of the simulations are inline with the observations from the experiments over range of discharge conditions.

**References:**


1. *H. Charan and R. Ganesh, "Observation of the Rayleigh-Bénard convection cells in strongly coupled Yukawa liquids," Physics of Plasmas 22 (2015), 10.1063/1.4927754*
2. *Dhaka, A., Subhash, P.V., Bandyopadhyay, P. et al. Auto-correlations of microscopic density fluctuations for Yukawa fluids in the generalized hydrodynamics framework with viscoelastic effects. Sci Rep* ***12****, 21883 (2022)*
3. *P. Kaur and R. Ganesh, "Negative entropy-production rate in Rayleigh-Bénard convection in two-dimensional Yukawa liquids," Physical Review E 100, 053201 (2019)*
4. *A. P. Thompson, H. M. Aktulga, R. Berger, D. S. Bolintineanu, W. M. Brown, P. S. Crozier, P. J. in 't Veld, A. Kohlmeyer, S. G. Moore, T. D. Nguyen, R. Shan, M. J. Stevens, J. Tranchida, C. Trott, S. J. Plimpton, Comp Phys Comm, 271 (2022) 10817.*

## Part 2 : Unlocking HPC Potential: A Practical Tutorial on Deploying Docker Hub Containers seamlessly with Singularity

In the second part of this series, this article delve into the practical steps of leveraging ready-made containers from Docker Hub (can be accessed from here) with Singularity in HPC settings. With Singularity and Docker containers pulled, this article focuses on commands and examples that streamline the process, ensuring efficient deployment of applications. .

***# Load singularity module available in ANTYA***
*[user@login1 ~]$ module  load singularity/3.5.3/3.5.3*

### Step 1: Downloading container from Docker Hub. For Instance, a ready made Tensorflow container has been pulled from Docker Hub

***# Pull Tensorflow container from Docker Hub in following format docker://container:tag***
*[user@login1 ~]$ singularity pull docker://tensorflow/tensorflow:2.8.0-gpu*
***# A SIF file will be created from docker container in users' pwd which user can verify and use to work with image file in future without need to pull container again.***

***NOTE: Sometimes newer tags may not work in current environment. Therefore try to pull older  tags which are still compatible with current environment.***

```
(base) [testuser2@login2 ~]$ singularity pull docker://tensorflow/tensorflow:2.8.0-gpu
INFO:    Converting OCI blobs to SIF format
WARNING: 'nodev' mount option set on /tmp, it could be a source of failure during build process
INFO:    Starting build...
Getting image source signatures
Copying blob 7b1a6ab2e44d done
Copying blob 28a427df77d0 done
Copying blob 7df9c933f388 done
Copying blob 5efcf2478dc2 done
Copying blob 3196a0117ed3 done
Copying blob 558a8016d36f done
Copying blob 4187faa68339 done
Copying blob 5cb96aedb476 done
Copying blob d827240235b3 done
Copying blob 75600f20c268 done
Copying blob 097a8b939f57 done
Copying blob 22ec0787b1d5 done
Copying blob ad4c0760f8ac done
Copying blob f26dd99dc5e3 done
Copying config 84b49f80a6 done
Writing manifest to image destination
Storing signatures
2024/02/27 14:02:31  info unpack layer: sha256:7b1a6ab2e44dbac178598dabe7cff59bd67233dba0b27e4fbd1f9d4b3c877a54
2024/02/27 14:02:32  info unpack layer: sha256:28a427df77d0b2d2c3290a9fa08f16f3812698d637c3657a740eb8640f901467
2024/02/27 14:02:33  info unpack layer: sha256:7df9c933f388e394ee82476829a80303216b989c2572548d4e2795e26ecb3f92
2024/02/27 14:02:33  info unpack layer: sha256:5efcf2478dc291438a65f6ca2fef61f1190b506c37aeb15e277affcb37295780
2024/02/27 14:02:33  info unpack layer: sha256:3196a0117ed30ca93cde89530ecd0060f9058eadaa2826b878ded77d97c08ddb
2024/02/27 14:02:33  info unpack layer: sha256:558a8016d36fad3ad0d79359400a62bf8df911d2eb134ef9d60407770e7ee308
2024/02/27 14:02:36  warn rootless{usr/lib/x86_64-linux-gnu/gstreamer1.0/gstreamer-1.0/gst-ptp-helper} ignoring
2024/02/27 14:03:25  info unpack layer: sha256:4187faa68339e66067963b3d740d1cb78be74aa0a2419bce8950bfd9d8777735
2024/02/27 14:03:31  info unpack layer: sha256:5cb96aedb4769107a2f9feff8f01a9d801ea1c92be19bc483bc93a26c7053b2d
2024/02/27 14:03:31  info unpack layer: sha256:d827240235b3b8e19d192a1b9ff21a6655b5006fa4a18f5c5f554cbda8ff1c42
2024/02/27 14:03:32  info unpack layer: sha256:75600f20c268e5e497e9e362b2d379f23ad458d3a0687f33b79de8ca511e249b
2024/02/27 14:03:32  info unpack layer: sha256:097a8b939f57a0af3ca30a2990975f6939dda69fb74c70f9fbf065b49dda7176
2024/02/27 14:03:32  info unpack layer: sha256:22ec0787b1d50bfad015777328f281e9c03e4a8ddb7199966a578924f5186d68
2024/02/27 14:03:52  info unpack layer: sha256:ad4c0760f8acbb58f15ca19c798b38b6c514c300ed735dc825291e94ea05b665
2024/02/27 14:03:52  info unpack layer: sha256:f26dd99dc5e33c409d436453ac9083bd27e3169c2a6281e13d0c79ec8b03e56e
INFO:    Creating SIF file...
INFO:    Build complete: tensorflow_2.8.0-gpu.sif
(base) [testuser2@login2 ~]$ ls
index.html  index.html.1  index.html.2  jupyter_test.sh  tensorflow_2.8.0-gpu.sif  test
(base) [testuser2@login2 ~]$
```

### Step 2: Use the Image file to run the script (Directly)

***# To directly run the executable from container***
*[user@gn04 ~]$ singularity run --nv tensorflow_2.8.0-gpu.sif python test.py*

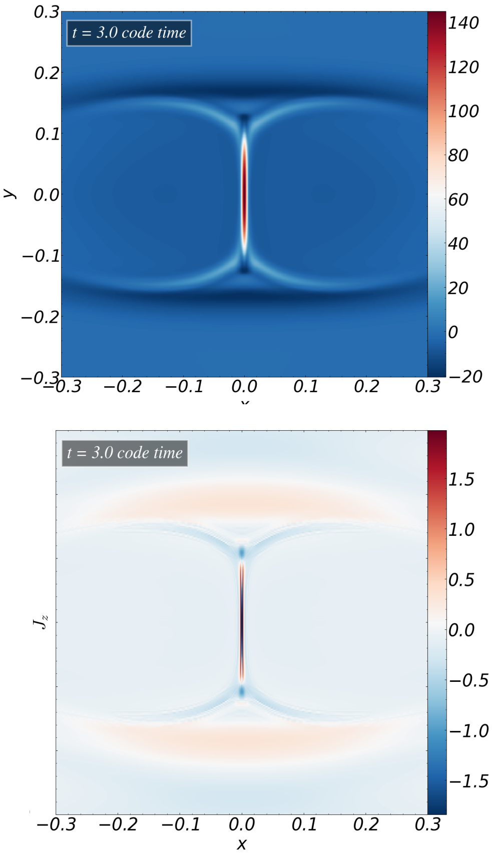
```
(base) [testuser2@gn04 tensorflow_container]$ singularity run --nv tensorflow_2.8.0-gpu.sif python testScript.py
2024-03-01 12:01:49.122365: I tensorflow/core/platform/cpu_feature_guard.cc:151] This TensorFlow binary is optimized with oneAPI Deep Neural Network Library (oneDNN) to use the following CPU instructions in pe
rformance-critical operations:  AVX2 AVX512F FMA
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
2024-03-01 12:01:53.529972: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1525] Created device /job:localhost/replica:0/task:0/device:GPU:0 with 15401 MB memory:  -> device: 0, name: Tesla P100-PCIE-16GB,
 pci bus id: 0000:18:00.0, compute capability: 6.0
2024-03-01 12:01:53.536660: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1525] Created device /job:localhost/replica:0/task:0/device:GPU:1 with 15401 MB memory:  -> device: 1, name: Tesla P100-PCIE-16GB,
 pci bus id: 0000:af:00.0, compute capability: 6.0
Epoch 1/100
2024-03-01 12:01:58.090414: I tensorflow/stream_executor/cuda/cuda_dnn.cc:368] Loaded cuDNN version 8100
938/938 [==============================] - 22s 8ms/step - loss: 0.1788 - accuracy: 0.9433 - val_loss: 0.0744 - val_accuracy: 0.9773
Epoch 2/100
938/938 [==============================] - 8s 8ms/step - loss: 0.0515 - accuracy: 0.9851 - val_loss: 0.0387 - val_accuracy: 0.9879
Epoch 3/100
938/938 [==============================] - 7s 8ms/step - loss: 0.0380 - accuracy: 0.9887 - val_loss: 0.0333 - val_accuracy: 0.9910
Epoch 4/100
938/938 [==============================] - 7s 8ms/step - loss: 0.0293 - accuracy: 0.9912 - val_loss: 0.0440 - val_accuracy: 0.9889
Epoch 5/100
938/938 [==============================] - 7s 8ms/step - loss: 0.0252 - accuracy: 0.9925 - val_loss: 0.0418 - val_accuracy: 0.9900
Epoch 6/100
938/938 [==============================] - 7s 8ms/step - loss: 0.0208 - accuracy: 0.9943 - val_loss: 0.0420 - val_accuracy: 0.9895
Epoch 7/100
938/938 [==============================] - 7s 8ms/step - loss: 0.0188 - accuracy: 0.9948 - val_loss: 0.0293 - val_accuracy: 0.9926
Epoch 8/100
938/938 [==============================] - 7s 8ms/step - loss: 0.0170 - accuracy: 0.9955 - val_loss: 0.0363 - val_accuracy: 0.9924
Epoch 9/100
160/938 [====>.........................] - ETA: 5s - loss: 0.0128 - accuracy: 0.9962
```

### Step 3: To open shell associated in the container environment

***# To open shell and work in interactive mode***
*[user@gn04 ~]$ singularity shell --nv tensorflow_2.8.0-gpu.sif*
*Singularity> python*
*>>> import tensorflow as tf*
*>>> print(tf.__version__)*
2.8.0
*>>> print(tf.config.list_physical_devices('GPU'))*
[PhysicalDevice(name='/physical_device:GPU:0', device_type='GPU'), PhysicalDevice(name='/physical_device:GPU:1', device_type='GPU')]

***NOTE: Users are advised to submit job using batch job script submission. Kindly use the command mentioned while using interactive mode in job script for execution.***

## HPC PICTURE OF THE MONTH




**Pic Credit: Dr. Jagannath Mahapatra**

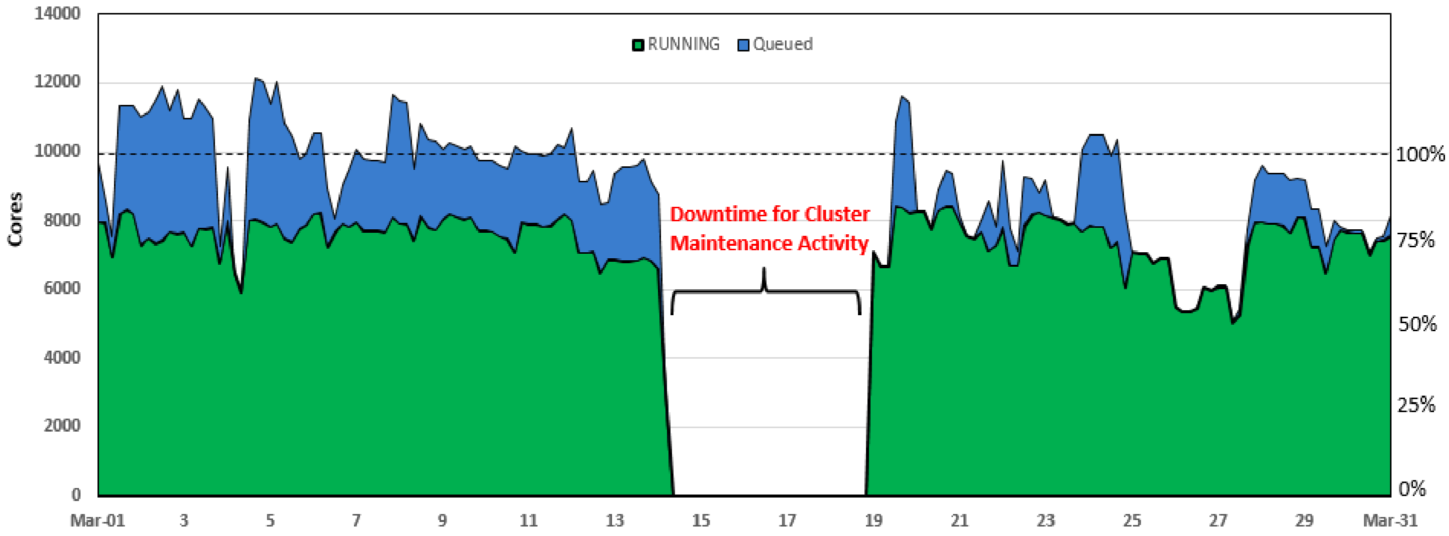
**A zoom-in view of a reconnecting current sheet (site of magnetic reconnection, formed when two magnetic flux tubes coalesce) showing colormap of out-of-plane current density Jz (left panel) and □.u (divergence of velocity, a measure of plasma compressibility, right panel) at the time of maximum reconnection rate. Compressibility effects are found to be the strongest in the vicinity of the reconnection site. The simulation are carried out using MPI-AMRVAC solver using a mesh grid of 4096×2048 in Antya cluster.**

**NOTE: Multinode Support for Singularity Containers**

**Generally, some ready made open source containers may not support multi node execution. There are some external provisions which can be used to span containers on multi node. Hence, users are advised to submit singularity container jobs on singl e node. The provisions will be discussed in future articles.**

# ANTYA Utilization: MARCH 2024

**ANTYA Daily Observed Workload**




## Other Recent Work on HPC (Available in IPR Library)

| | |
|---|---|
| **Aerodynamic Drag Reduction Studies in The Presence of Dielectric Barrier Discharge Plasma** | **Jugal Chowdhury** |
| **Quasi-longitudinal whistler propagation in presence of finite ion response** | **Gayatri Barsagade** |
| **Visible Camera Image Interpretation of Tokamak Plasma Using Deep Learning Method** | **Vishnukumar Rahingaram Chaudhari** |

## ANTYA HPC USERS' STATISTICS— MARCH 2024

**Total Successful Jobs~ 2160**

♦ **Top Users** (Cumulative Resources)

- **CPU Cores** — **Amit Singh**
- **GPU Cards** — **Swapnali khamaru**
- **Walltime** — **Jugal Chowdhury**
- **Jobs** — **Jugal Chowdhury**

## Acknowledgement

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

**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 GAṈANAM, please write to us.***

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