# Performance Analysis of GS2 Plasma Turbulence Code

*Sally Bridgwater, Nick Dingle, Numerical Algorithms Group, UK*

## Introduction

GS2 [1] is an open source gyrokinetic simulation code used to study turbulence in plasma, one application is for fusion experiments. It is a gyrokinetic flux tube initial value and eigenvalue solver and is written in Fortran and parallelised with MPI.

Performance analysis was performed under the **Performance Optimisation and Productivity Centre of Excellence (POP)** using a methodology to narrow down underlying causes of inefficiency. After an initial analysis changes were made by the developers based upon the recommendations. The refactored code was further analysed with two inputs variants, this comparison is presented below. Performance Analysis was performed using the BSC tools Extrae and Paraver [2].

## Methodology

POP efficiency metrics give an overview of how well the parallelisation of the application works and how efficiently the hardware is used [3].

The metrics are organised in a hierarchy and give a detailed overview of the performance of an application in a very condensed form. An ideal network is defined as instantaneous data transfer.

**Global Efficiency** GE = CE * PE

**Computational Efficiency** $CE = Tcomp_{ref}/Tcomp_p$

**Parallel Efficiency** $PE = LB * SE * TE$

**Instruction Efficiency (INS)**
Total useful instructions scaling with core count (*p*) relative to ref value (*ref*)
$INS = INS_p/INS_{ref}$

**Instructions per Cycle Efficiency (IPC)**
IPC scaling with core count (*p*) relative to ref value (*ref*)
$IPC = IPC_{ref}/IPC_p$

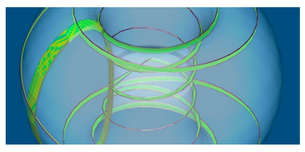
**Load Balance (LB)**
How evenly time in computation is distributed across ranks (*r*)
$LB = AVE_r(comp)/MAX_r(comp)$

**Serialisation Efficiency (SE)**
Time lost waiting for communication partners
$SE = MAX_r(comp)_{ideal}/runtime_{ideal}$

**Transfer Efficiency (TE)**
Inefficiency due to time in data transfer Network
$TE = runtime_{ideal}/runtime$

Metrics are percentages where 0% is low, 80-85% is the cut off for good performance and 100% is ideal performance.

## Analysis



A single GS2 timestep for this analysis included the following phases:

1) Nonlinear Advance (N)
2) Linear Advance (L)
3) Field Solver (F)
4) Second Linear Advance (L)

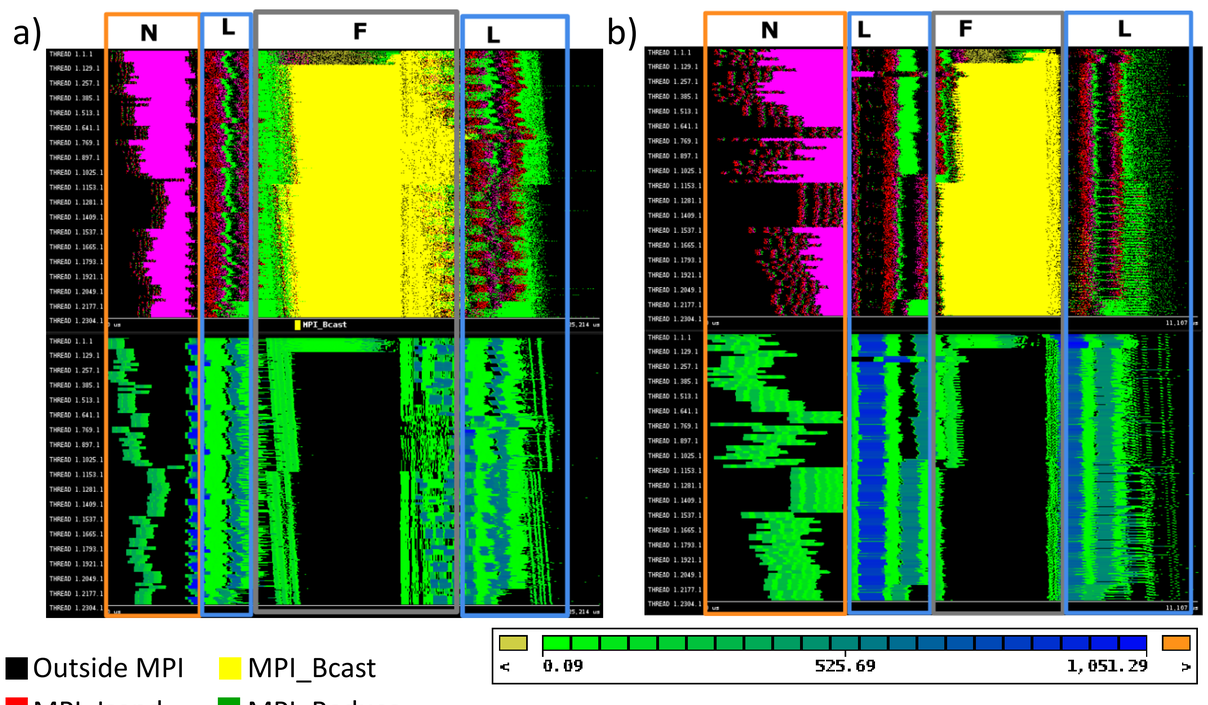
The analysis was performed on 2304 MPI ranks on the ARCHER UK Supercomputer.

The main input variables are :

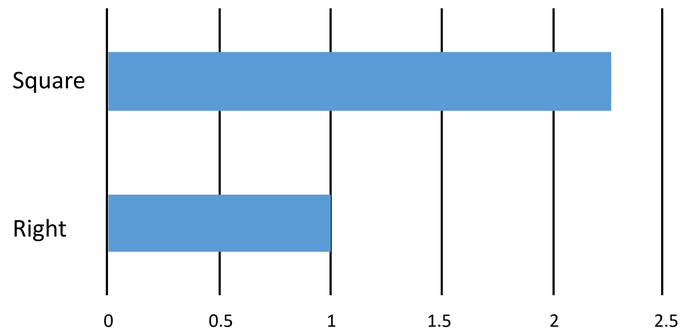
(ntheta, ngauss, negrid, nspec, nx, ny, nstep, field, layout)= (26, 5, 8, 2, 24, 24, 100, gf, yxles)

**Two versions of data distribution were analysed.** How five of the dimensions of the gyrokinetic distribution function are distributed across the MPI ranks was varied.

| Dimension | x | y | l | e | s |
|---|---|---|---|---|---|
| Right (default) | 3 | 2 | 24 | 8 | 2 |
| Square | 3 | 8 | 6 | 8 | 2 |




*Timelines of the two versions of GS2 for a) right and b) square split domains. Showing the MPI calls (top) and the duration of computation coloured by gradient (bottom) for one timestep with the four application phases of GS2 on 2304 MPI ranks.*




The Square split domain is over twice as fast as the default data distribution.

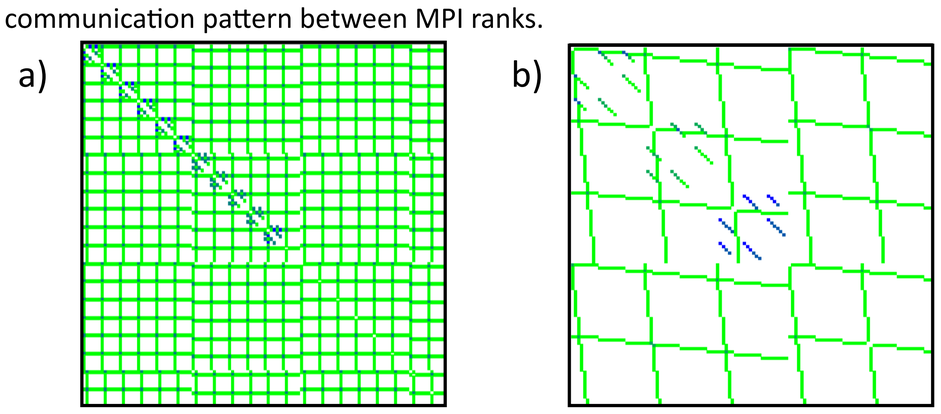
| | Right | Square |
|---|---|---|
| Runtime (ns) | 2.52E+07 | 1.11E+07 |

| | Right | Square |
|---|---|---|
| Number of cores | 2304 | 2304 |
| **Global Efficiency** | 26.0% | 50.7% |
| Computational Scaling | 75.2% | 100.0% |
| Useful Instructions Scaling | 85.3% | 100.0% |
| Useful IPC Scaling | 96.4% | 100.0% |
| Parallel Efficiency | 34.5% | 50.7% |
| Load Balance Efficiency | 78.8% | 80.2% |
| Serialisation Efficiency | 97.3% | 96.6% |
| Transfer Efficiency | 45.0% | 65.4% |

Issues highlighted from the metrics, in order of importance:

- Transfer Efficiency is low for both but significantly better for the square split. This is where most of the improvement is seen
- Load Balance is okay for both
- IPC and Instructions are worse for the right split i.e. more work is done slower on the right split
- Good Serialisation i.e. little time is spent waiting for communication partners to be available.

Since the Transfer Efficiency is the main inefficiency we investigated this further by looking into the communication pattern between MPI ranks.




*Communication matrix for 120 ranks for a) right and b) square split domains. Coloured by number of messages sent between partners, light green are fewer and dark blue more messages.*

Considerably more point-to-point messages are sent with the right split domain than the square split domain. The pattern is related to the input parameters for the domain decomposition.

## Conclusions

The square split domain was the most efficient split tested and around twice as fast as the current default option.

- Communication is the key bottleneck, specifically the amount of data transferred and the complexity of the communication patterns.
- Further investigation of the impact of data distribution with different inputs to determine an optimal configuration for runs is required
- This work clearly demonstrates there is a large scope for improvement to the communication in GS2

## Acknowledgements & References

We would like to thank STFC and CCFE for giving us permission to showcase the work they have undertaken with POP.

The POP project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 676553.

Plasma HEC Consortium EPSRC grant number EP/L000237/1.
Collaborative Computational Project in Plasma Physics - grant number EP/M022463/1.

[1] "Comparison of Initial Value and Eigenvalue Codes for Kinetic Toroidal Plasma Instabilities," M. Kotschenreuther, G. Rewoldt, and W.M. Tang, Comp. Phys. Comm. 88, 128 (1995).

[2] G. Llort, Servat, H., González, J., Giménez, J., and Labarta, J., "On the Usefulness of Object Tracking Techniques in Performance Analysis", in *Proceedings of SC13: International Conference for High Performance Computing, Networking, Storage and Analysis*, New York, NY, USA, 2013.

[3] "Efficiency Metrics in a POP performance audit", https://pop-coe.eu/sites/default/files/pop_files/metrics.pdf