Gyro-Kinetic DataBase project

*Guillaume Fuhr[1], Yann Camenen[1], Karel L. Van De Plassche[2,4],
Feda Almuhisen[3], Clarisse Bourdelle[3], Jonathan Citrin[4], Aaron Ho[4],
Matisse Lanzarone[1], Anass Najlaoui[1]*

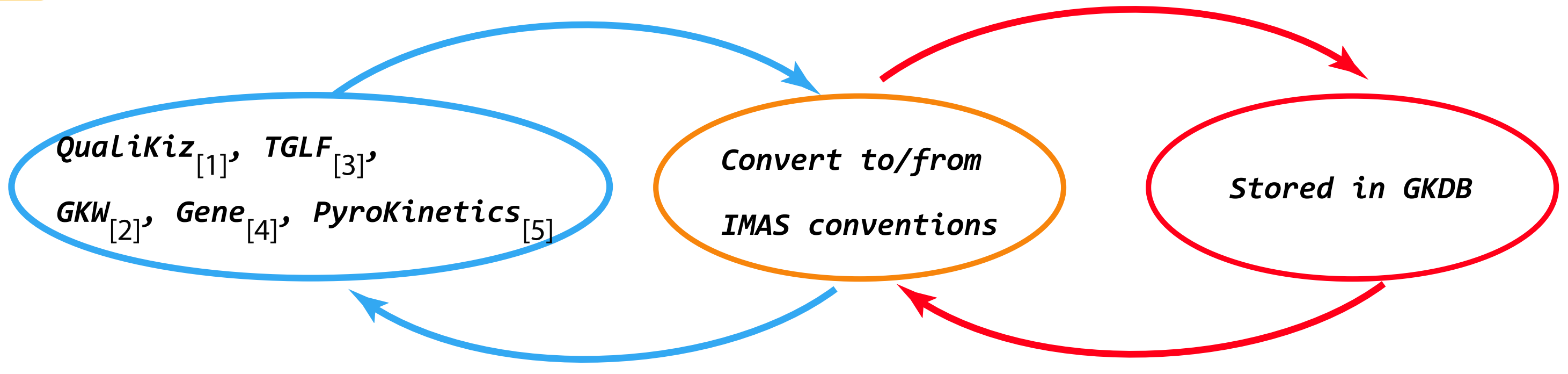
[1]Aix-Marseille Univ., CNRS, PIIM UMR7345, Marseille, France
[2]Ignition Computing, Keizersgracht 16D, 5611GD Eindhoven, Netherlands
[3]CEA, IRFM, F-13108 Saint-Paul-lez-Durance, France
[4]DIFFER- Dutch Institute for Fundamental Energy Research, De Zaale 20, 5612 AJ Eindhoven, Netherlands

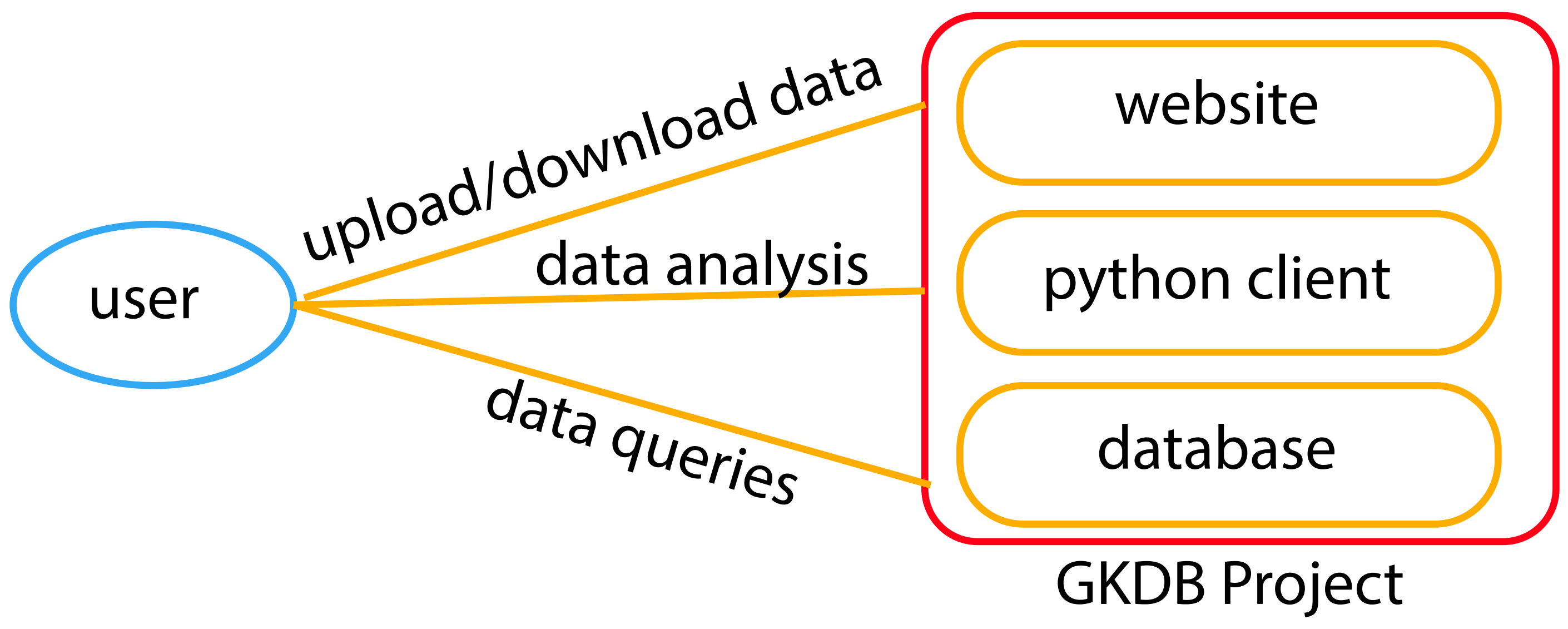
E-mail : guillaume.fuhr@univ-amu.fr

Introduction

The GKDB Project : a repository of simulation data, a platform for code benchmarking, and a springboard for the development of fast and accurate turbulent transport models.
- Using IMAS Conventions for interoperability -> ***Any local gyrokinetic code can use it or participate***.

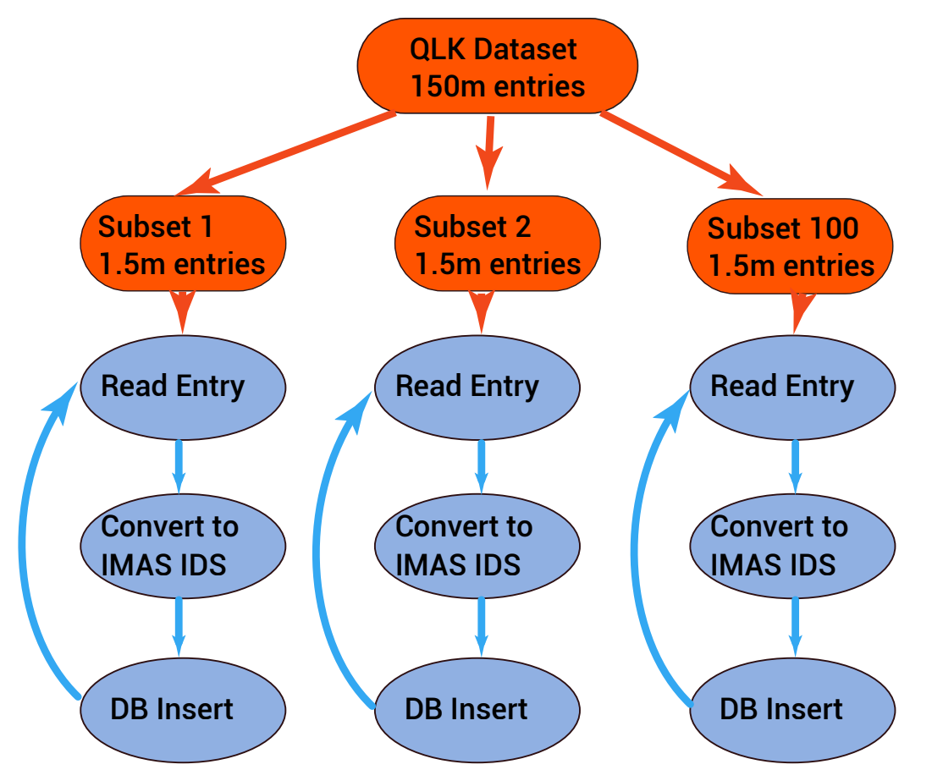
GKDB Project, a fully Open Source Project to manage local gyrokinetics simulations:
- Interface in Python under development : python package followed by webinterface access
- Python package follows Imas Data Structure conventions : *idspy_dictionaries*
  - only dependencies : Python>=3.10 and Numpy
- Project hosted and documented on GitLab (https://gitlab.com/gkdb/)
- Docker-Compose Image File available for testing purpose
- GKDB Data will be released under Open Source Licence (expected CC BY 4.0 for Database data and ODBL for Database structure)
- Associated Python source code are released under BSD3 Licence



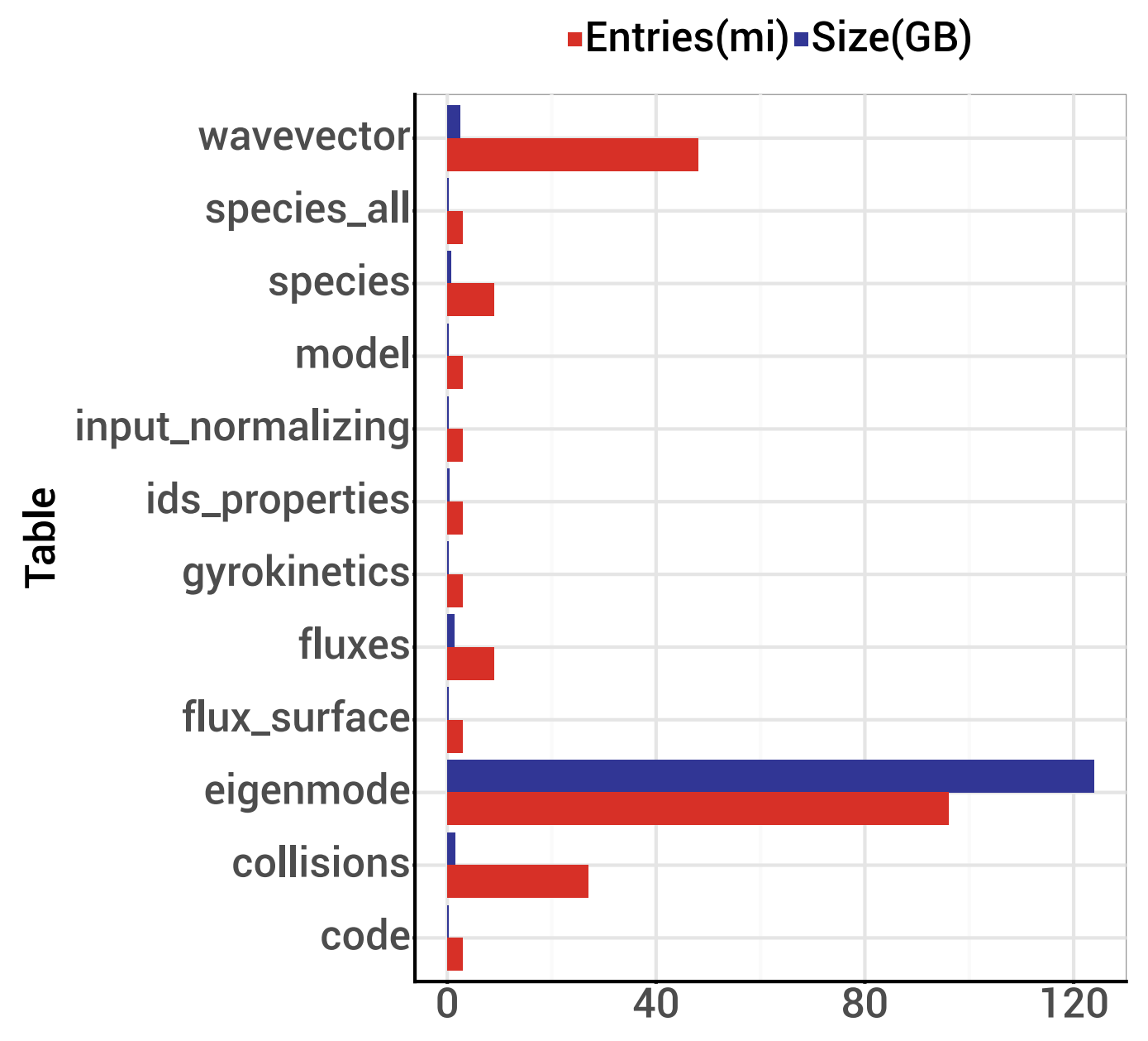


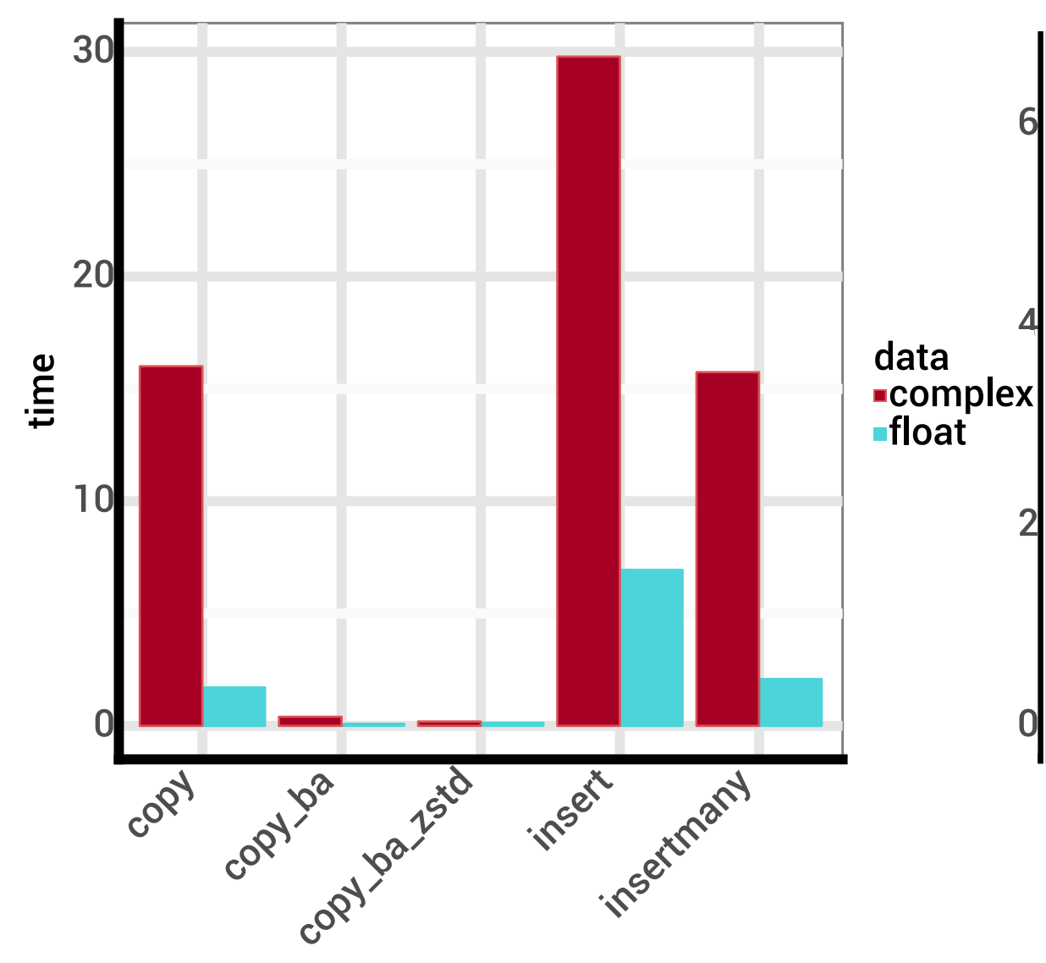
Tests and Performances

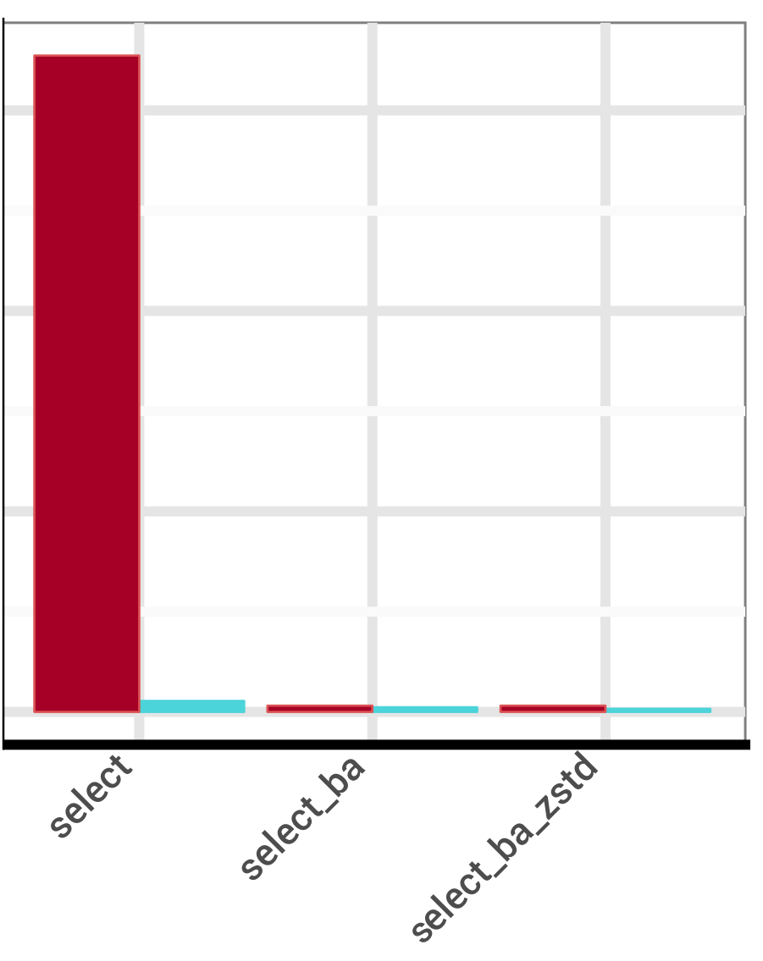
Stress Test :
- Using 20% of a 7D QualiKiz Hypercube simulations (16 millions simulations).
  - Input Datas : 460MB
  - «Brute Force method» for the test dataset : more than a year
  - Current Performances : 6h on a laptop (divide&conquer strategy, SQL Copy...)
  - DB size after eigenfunctions reconstruction : 140 GB
  - Expected DB size for the full dataset : 800 GB
  - Expected insertion time for the full dataset : 1.2 days



Current Pipeline : 5h for 3 millions IDS







Measured times including data mapping to :
- insert 100x arrays of 10 000 elements
- select 50 arrays of the same size

ByteArray + Zstd comp. :
- 150x faster for inserts
- 100x faster for queries

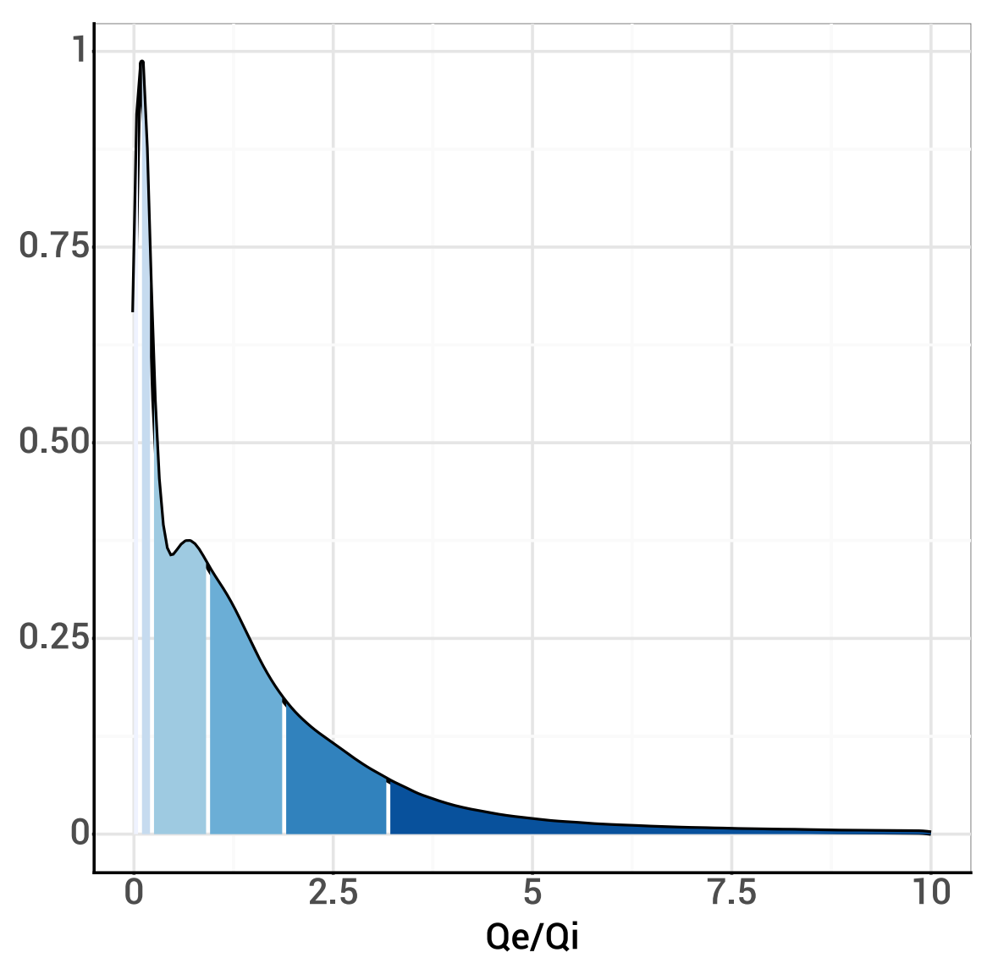
Allow us to use complex numbers without any performances cost.

Some Usage Examples

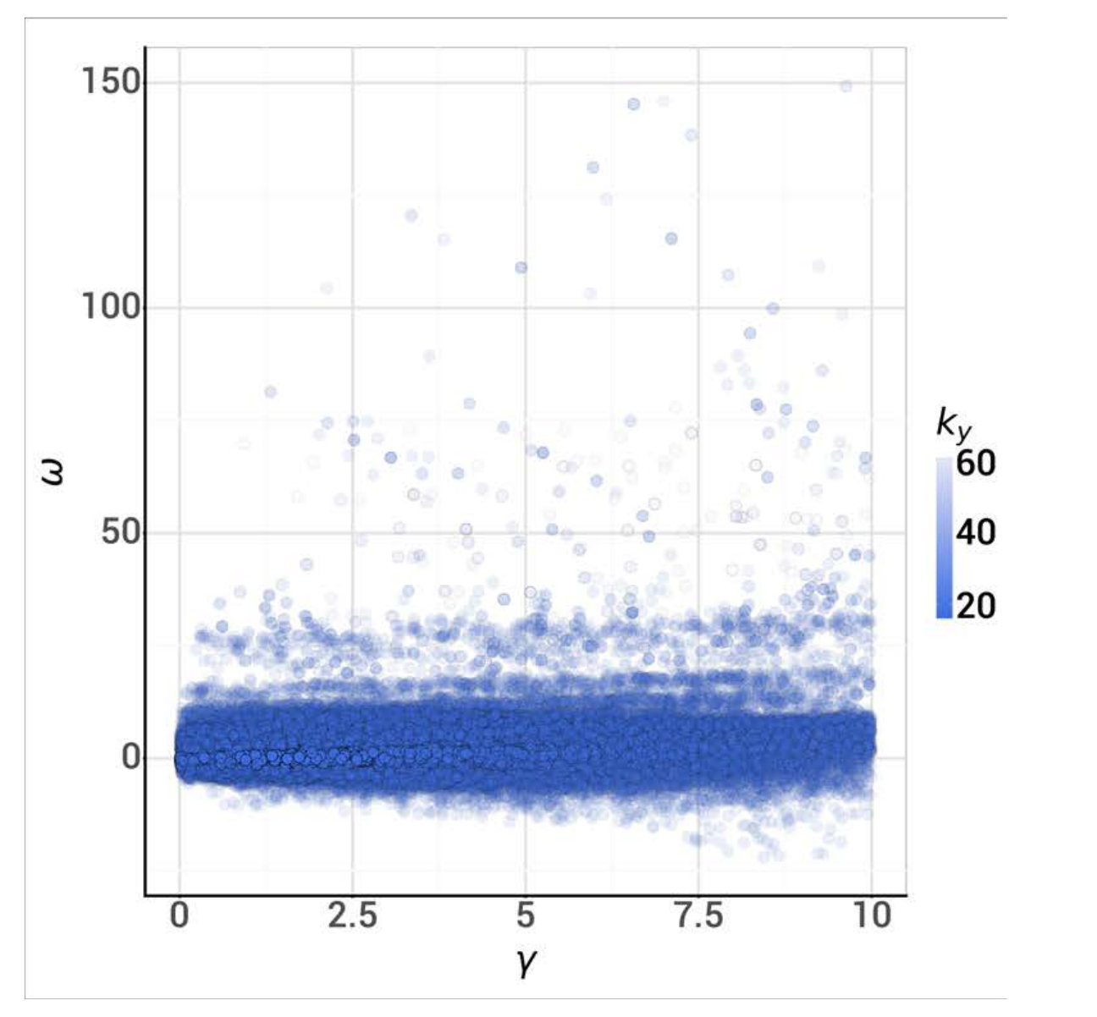
- Python Interface made as simple as possible.
- Usage of AsyncIO to increase performances with GKDB queries
- Some possible queries :

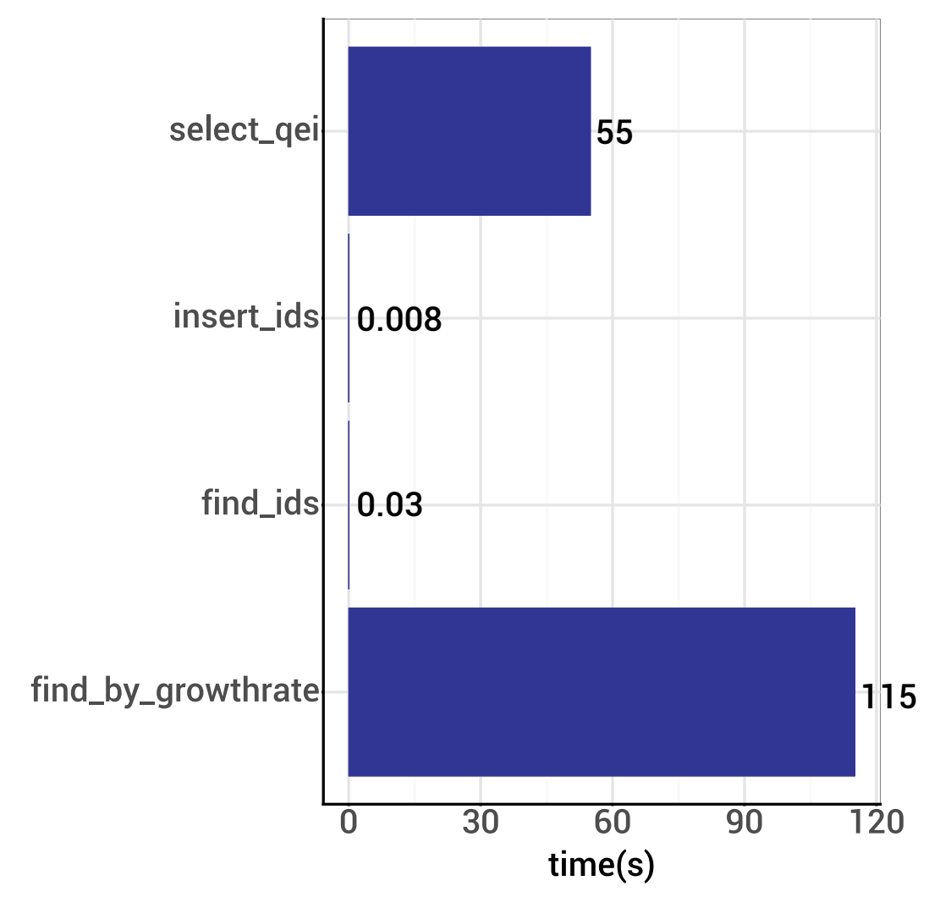
```
db = AsyncGkdbClient(GKDB_CFG...)
await db.connect()
query_gamma = await db.find_by_growthrate(1., 10.)
query_qei = await db.find_by_fluxes_ratio(0., 10.)
ids = await db.find_by_uuid('018a48ed......')
```

Density distribution of Q_e/Q_i

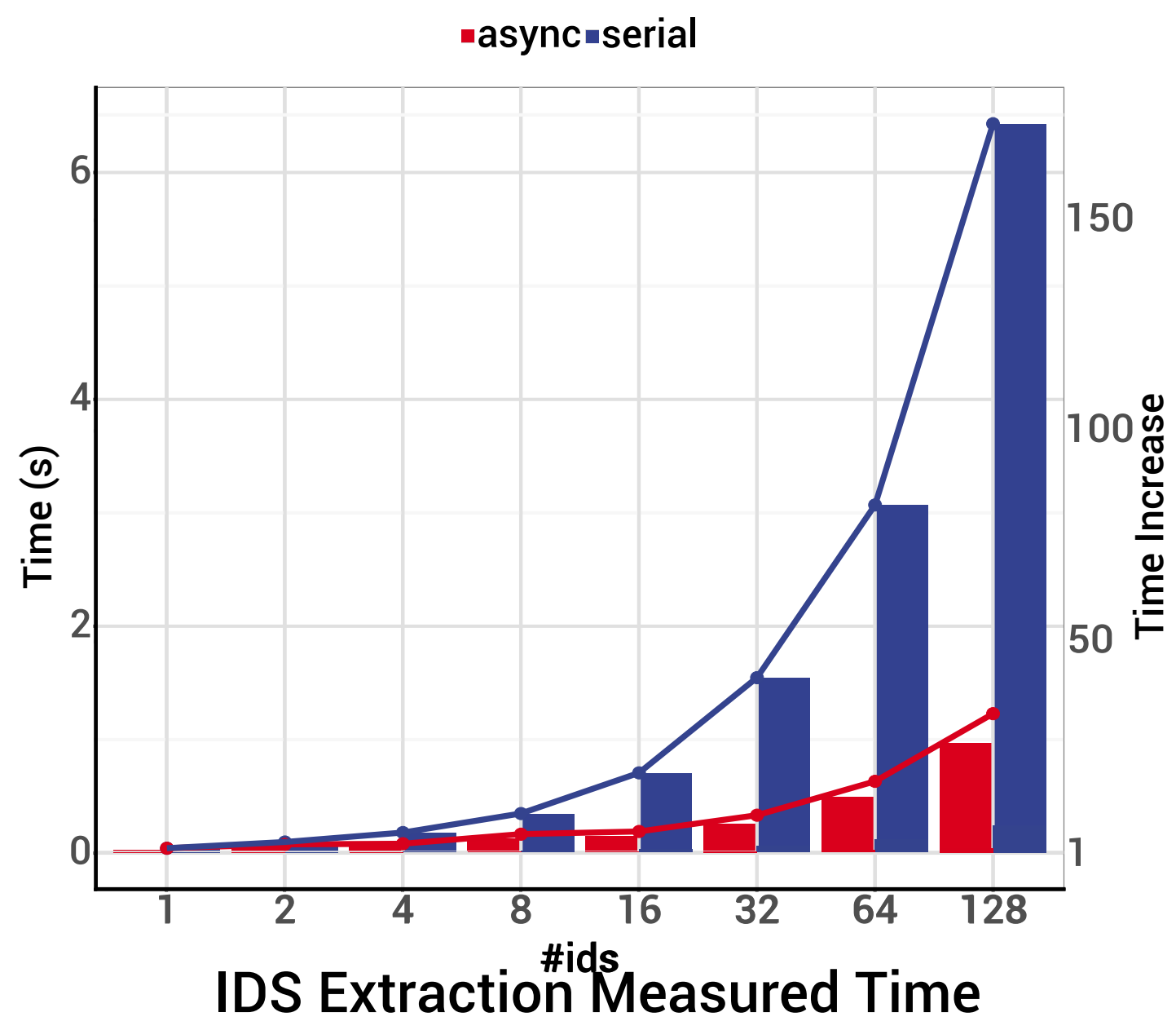


Mode growth rate vs Mode Frequency





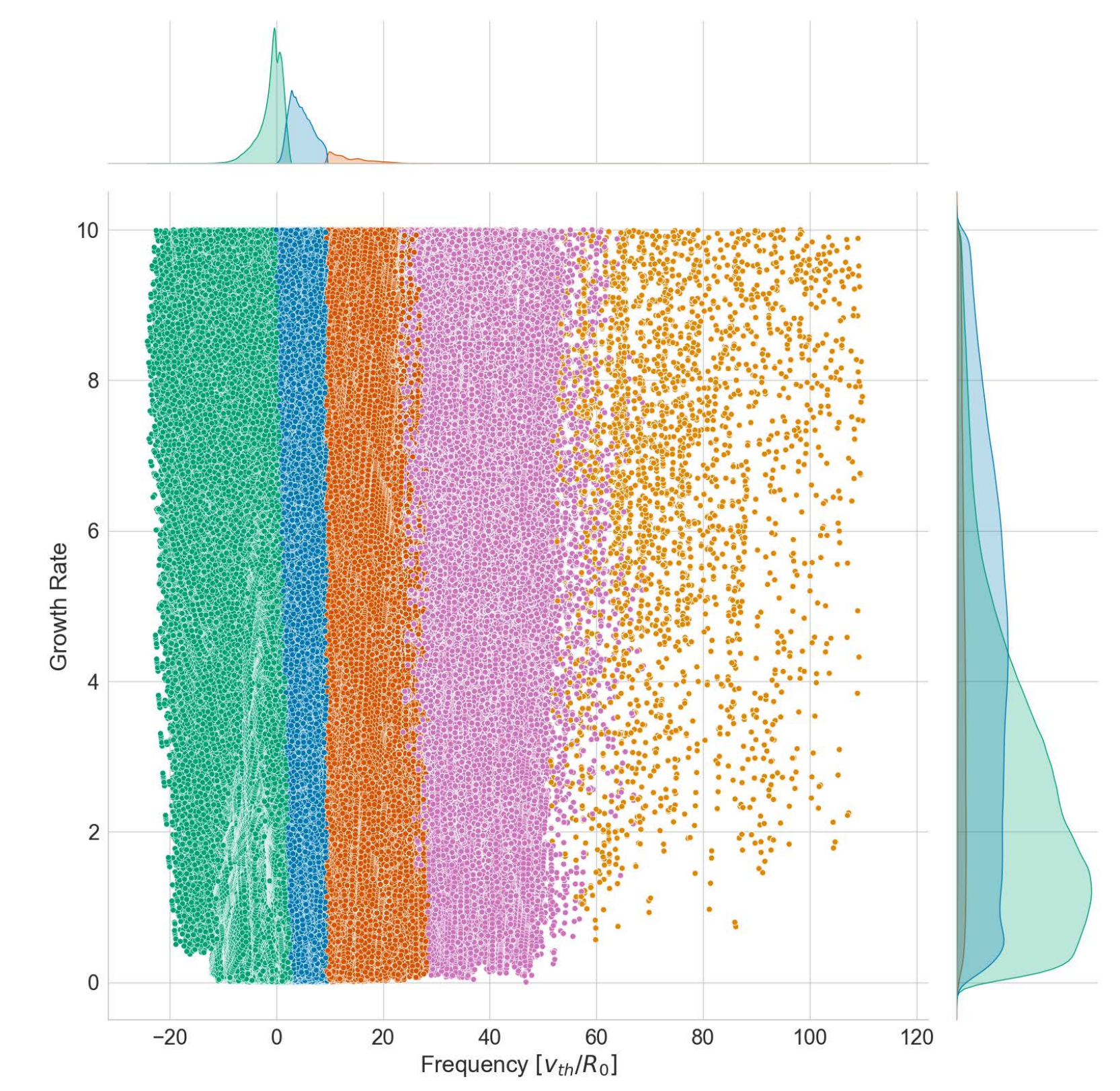
Query Times



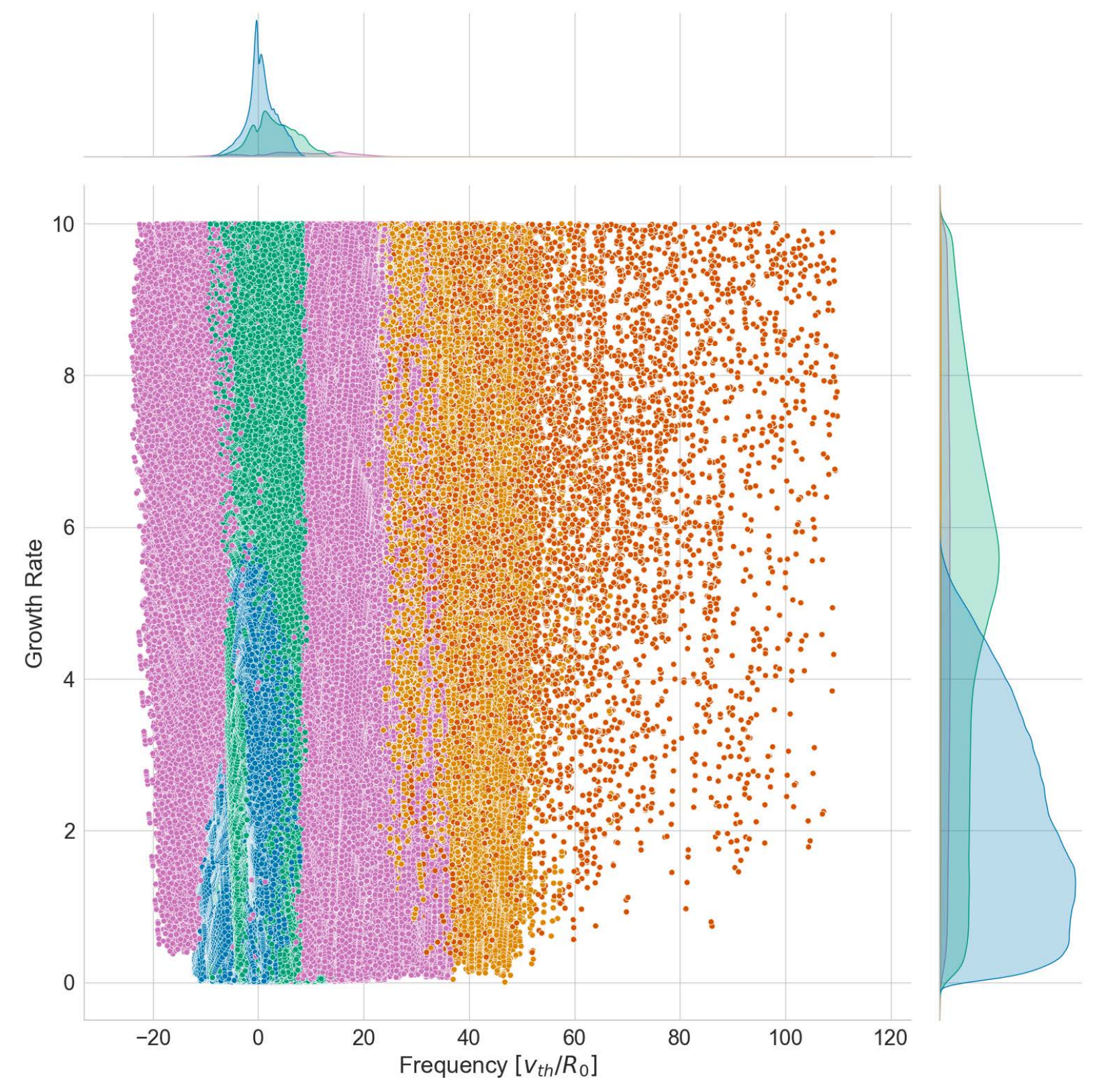
IDS Extraction Measured Time

Classification of linear instabilities in QualiKiz

- Instabilities are generally identified based on a range of expected values of frequencies for a given mode.
- What do we obtain using clustering techniques on simulation data?
- Presented analysis is based on an extraction from the GKDB for all growth rates and frequencies for eigenmodes based on Qualikiz simulations.
- Dataset contains growth rate, frequencies, kx, ky... for 20 millions of eigenmodes



K-Means Clustering



Gaussian Mixtures Clustering

Conclusion and Perspectives

A preliminary version of the GKDB project is presented here.
The GKDB is designed to be a repository of local gyrokinetics simulation data.
The aim of the project is to share local GK simulations and help for code comparisons, benchmarkings,...
The global amount of data in the DB will be usefull for studies like code analysis but also for training of IA models requiring huge quantity of data.

How to reach us or find out more about the GKDB project :
- *Project is hosted and documented on GitLab : https://gitlab.com/gkdb*
- *On slack : https://faster-groupe.slack.com/*

Milestone :
- Public release of the Web Interface
- Public release of the GKDB and the Python client

References :
[1] J. Citrin et al., Plasma Phys. Control. Fusion 59, 124005 (2017) - www.qualikiz.com
[2] A.G. Peeters et al., Comput. Phys. Comm. 180, 2650 (2009) - https://bitbucket.org/gkw/gkw/wiki/Home
[3] G.M. Staebler et al., Phys. Plasmas 14, 055909 (2007)
[4] F. Jenko et al., Phys. Plasmas 7, 1904 (2000) - https://genecode.org/
[5] https://github.com/pyro-kinetics/pyrokinetics

This research was funded, in whole or in part, by l'Agence Nationale de la Recherche (ANR), project ANR-22-CE30-0023