Fast Simulations and Bayesian Hyperparameter Optimization

L. Anderlini, A. Rizzi

Istituto Nazionale di Fisica Nucleare, Sezioni di Firenze[1] e Pisa[2]

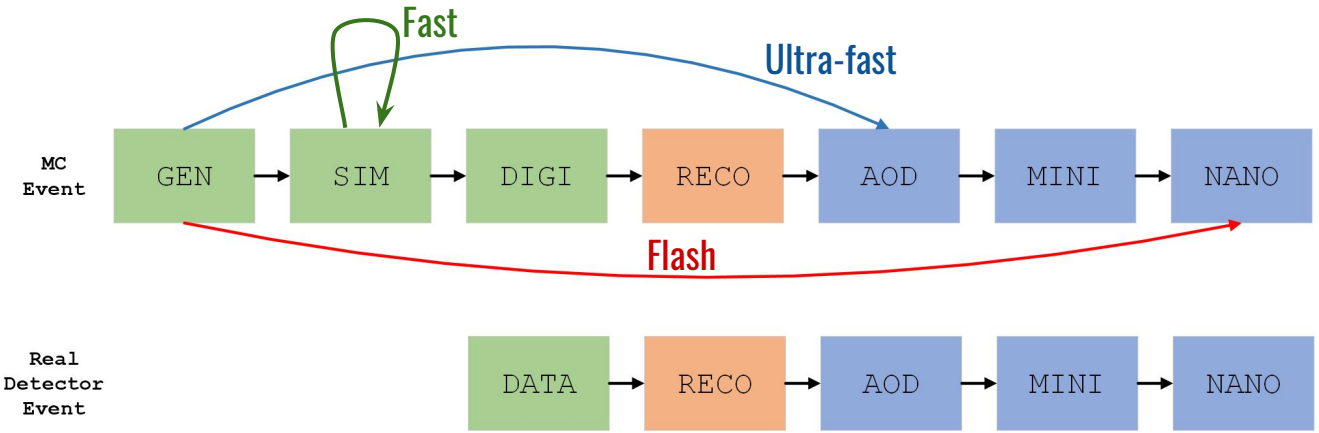
Istituto Nazionale di Fisica Nucleare
SEZIONE DI FIRENZE

Istituto Nazionale di Fisica Nucleare
SEZIONE DI PISA

Motivation: *unsustainable CPU costs for simulation*

The increase in luminosity expected for the upcoming and future Runs of the LHC at CERN will require **simulated samples with unprecedented statistics** to model data.

Today, detailed simulation accounts for the vast majority of the CPU costs for the LHC Collaborations. Several alternatives are under investigation to **speed up the production** of simulated events, the most promising rely on **Deep Generative Neural Networks**.



State of the art

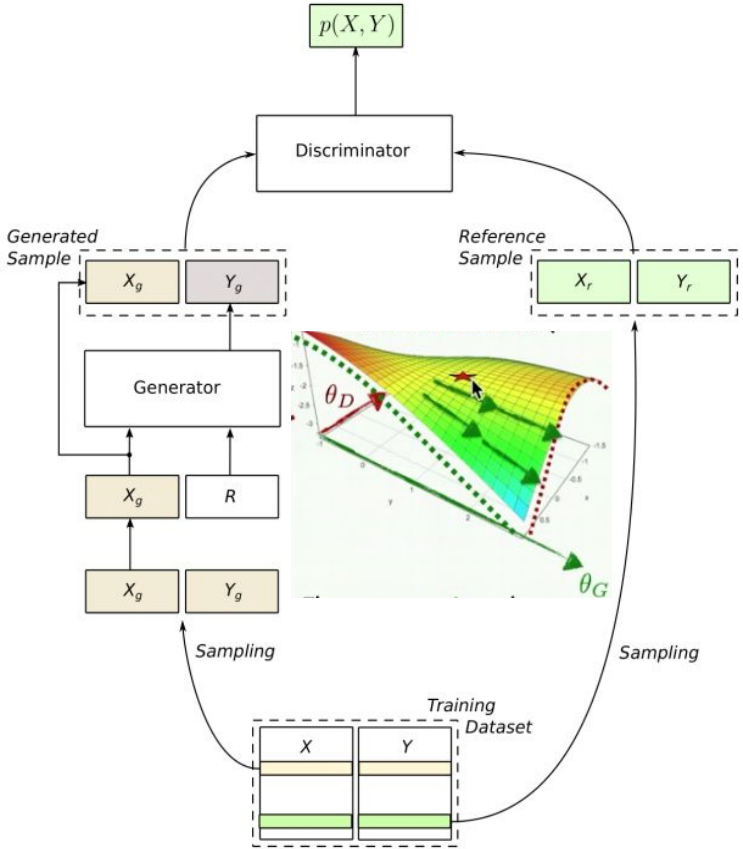
A survey on Italian activities ML-based fast sim (2022-10-13)

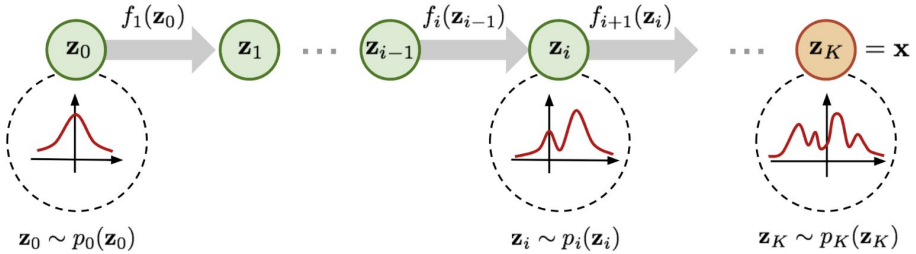
CMS: Vaselli Master Thesis GitHub Repository

LHCb: Barbetti Master Thesis Lamarr (LHCb) proceedings

ATLAS: AtlFast3 paper

Generative Deep Neural Networks





Generative Adversarial Networks (GANs) and **Normalizing Flows** (NFs) are emerging as go-to solutions for building parametrizations for fast simulations.

GANs require a very large number of epochs to reach the optimum (in a saddle point)

NFs are computationally very expensive to train.

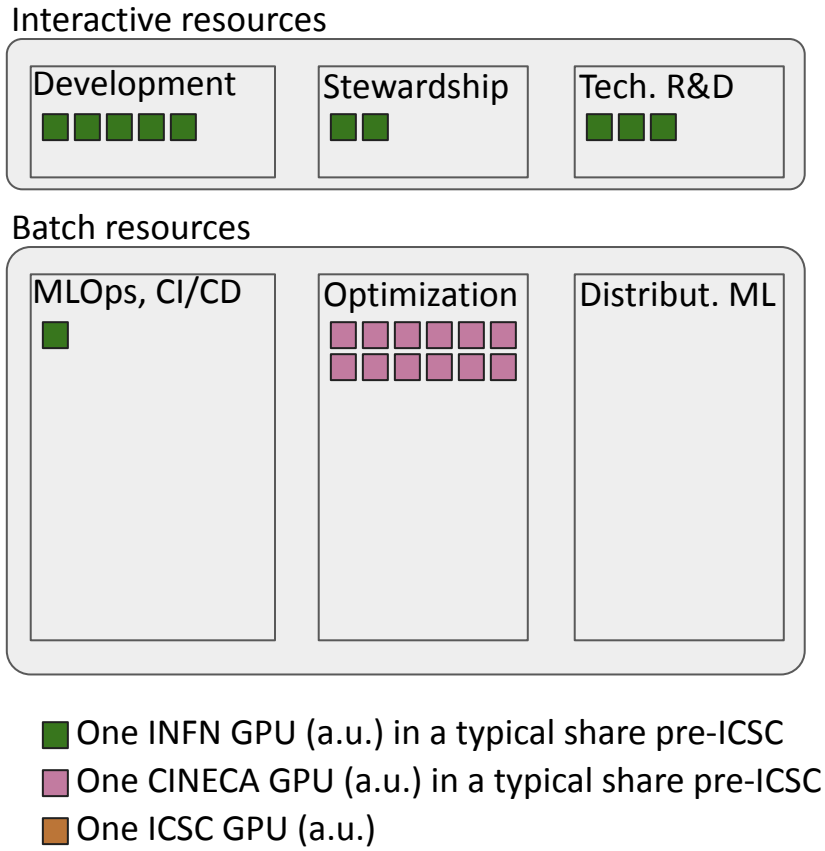
Current activity (pre-ICSC resources)

Mostly interactive access. GPUs in the Cloud (ML_INFN) and on premises are employed:

- **development** and training of the models
- pair programming and **tutoring**
- technological **R&D** (docker, minio…)

Not enough resources for MLOps, some CI/CD on private resources for dependencies.

Access to **M100** is a relief for hyperparameter optimization, but very hard to use interactively. No effort on distributed training due to lack of resources.



Ideal situation with ICSC resources

Ease access to resources for interactive development.

Important synergies with the Analysis Facility

Enable MLOps workflows in the cloud (possibly opportunistically).

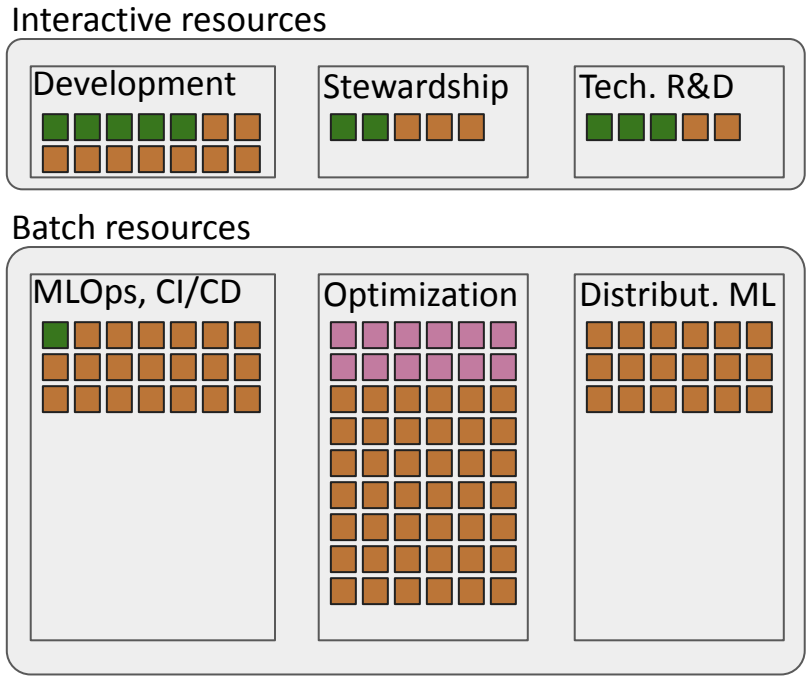
Discussions are ongoing…

Enable larger hyperparameter optimization campaigns.

Focus of this talk

Motivate studies on Distributed Training.

Might require additional people



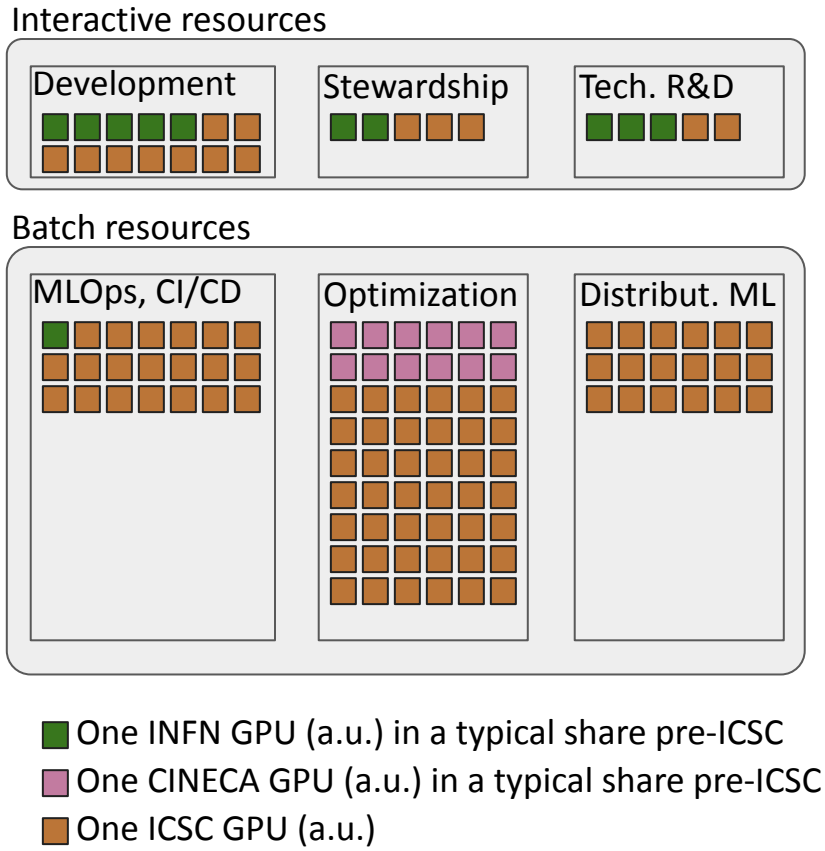
GPU resources: *matching demand and offer*

For optimal progress, good match of people and GPU resources is needed.

Today: relevant activity in Italy, but limited access to GPUs slows down developments

Tomorrow (rough estimates):

- an increase O(×10) in available GPUs would satisfy the demand,
- scaling further will require engineering effort (MLOps, distributed computing, code review, …) beyond current crew.



A word on the software infrastructure

Plugging more GPUs is not sufficient to make them available to the community.

Important effort on the Analysis Facilities for combining **interactive and batch access** to the CPUs in the cloud and the experience in ML_INFN **serving GPUs through INFN Cloud** set the foundations for serving FastSim (and other DL?) workloads in ICSC.

Disclaimer. In this talk we will focus on ***Hopaas,*** a particular element of this infrastructure designed to coordinate hyperparameter optimization studies through multiple sites.

Hopaas will not be particularly useful without the rest of the sw infrastructure.

[arXiv:2301.05522]

Hopaas: *Hyperparameter Optimization as a Service*

Hyperparameter exploration is a problem of optimization on a costly, noisy function.
Evaluating the function: *training the model with a given configuration*

GPU resources are made available by multiple providers (CERN, INFN, CINECA, private resources, AWS, GCP…).

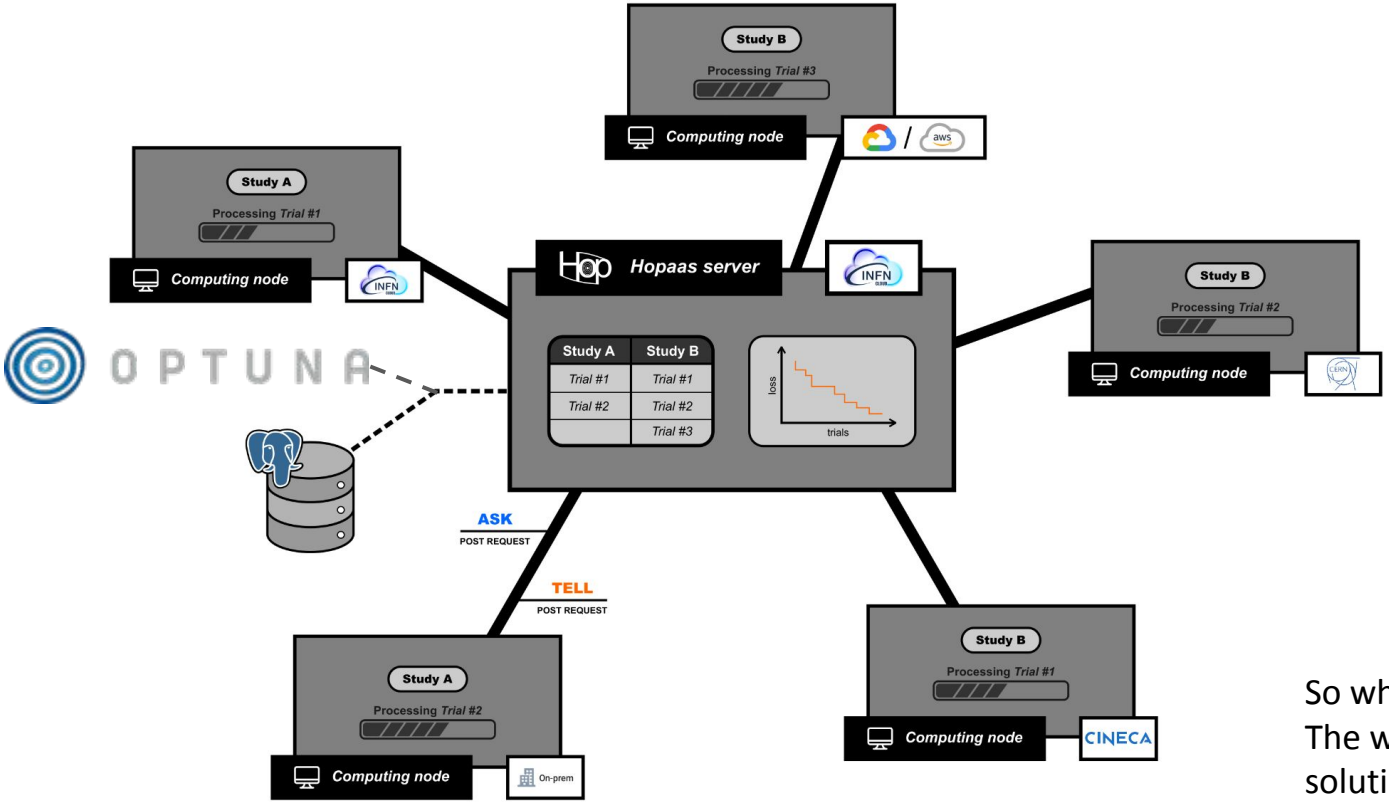
Studies may rely on multiple providers:

- in parallel (performing multiple trainings with different HP sets)
- changing site (concluding the study changing provider)

We aim at a central service coordinating HP optimization with state-of-the-art Bayesian techniques through multiple computing instances provisioned by multiple providers.

[arXiv:2301.05522]

A minimal service: *bayesian optimization + monitoring*



MLOps and ML tracking

Commercial and FOSS solutions to ML tracking exist and might be explored to complete or even replace Hopaas.

Some pointers:

- MLflow (FOSS)
- neptune.ai (OSS)
- Weights & Biases (commercial)
- optuna-distributed (FOSS)

So why *Hopaas?*
The workflows supported by FOSS solution were incompatible with M100 setup.

Rest APIs: *a well-established Web Standard*

To cope with firewall and network restrictions set by different providers, Hopaas was designed to rely on token-authenticated **HTTPS-encrypted Rest APIs**.

Each computing node (spawned for example with slurm or HTCondor) declares to Hopaas the study it will perform and **`ask`**s for a set of hyperparameters to evaluate.

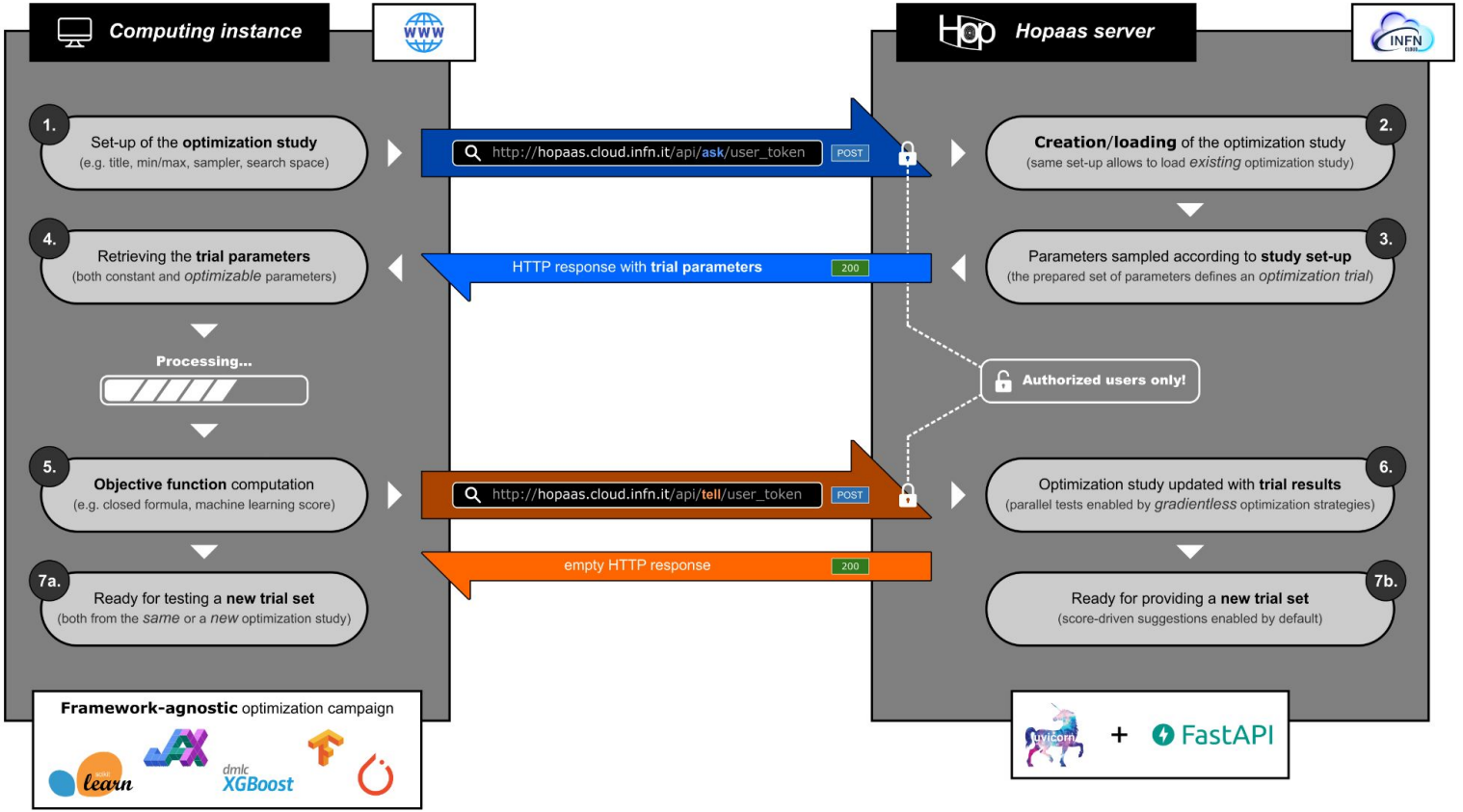
At the end, it **`tell`**s hopaas the outcome of the trial.

Preliminary evaluation of the loss function can optionally be submitted to Hopaas, which may suggest to interrupt the study to save computing power (**`should_prune`**).

👉API reference documentation

[arXiv:2301.05522]

API-based Ask&Tell interface



[arXiv:2301.05522]

State-of-the-art gradient-less Optimization with Optuna

Gradient-less optimization of noisy function has been an active research field for the last decade. Optuna is emerging as the to-go framework, providing off-shelf multiple **Bayesian**, **Genetic**, and **Evolutionary Optimization** algorithms.

Hopaas relies on Optuna as a dependency of the back-end application.

The configuration of the Optuna algorithms is defined through the **`ask`** API.

Optuna natively relies on a separate **PostgreSQL** backend for storage.
Multiple instances of the back-end application can run in parallel scaling the service in case of high demand*.

*) Current docker-compose implementation does not include autoscaling upon higher demand.

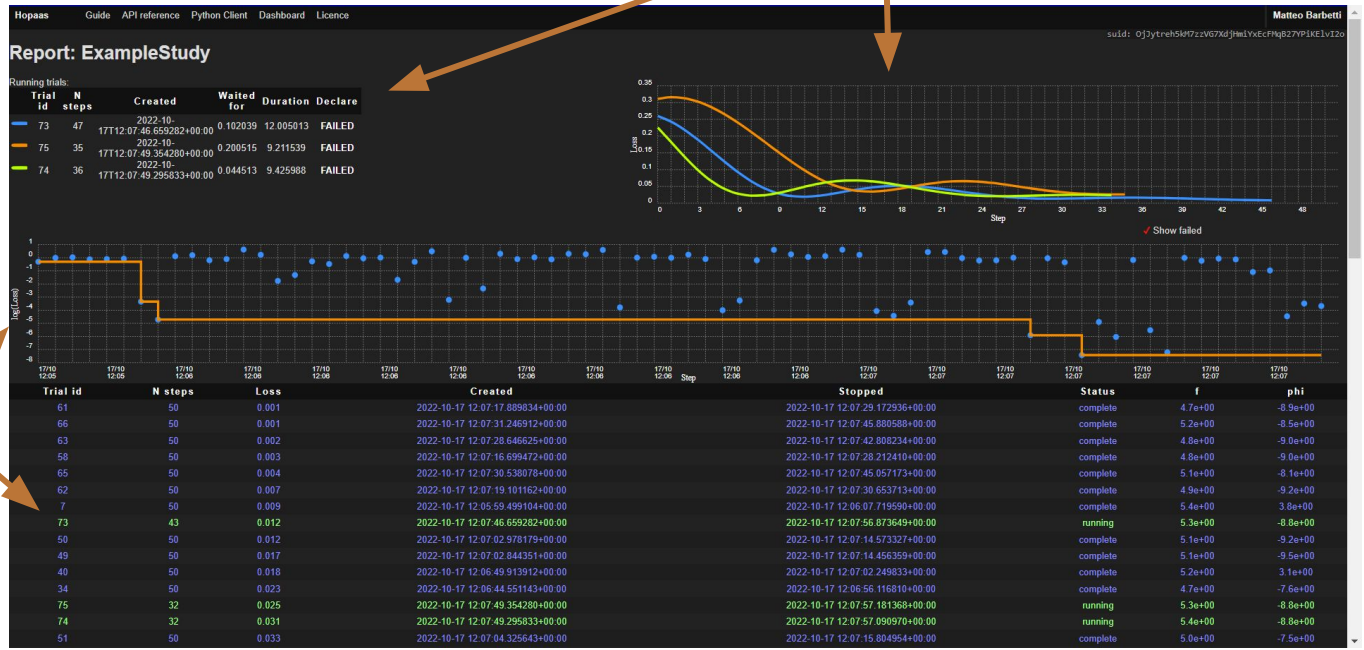
[arXiv:2301.05522]

Front-end web application and monitoring

The status of the ongoing monitoring can be tracked through a minimal web interface developed in vanilla JavaScript and Chartist.js.

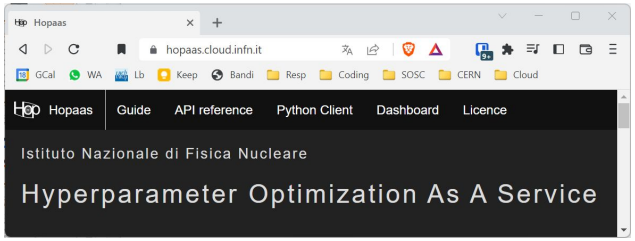
Monitoring of the ongoing trials

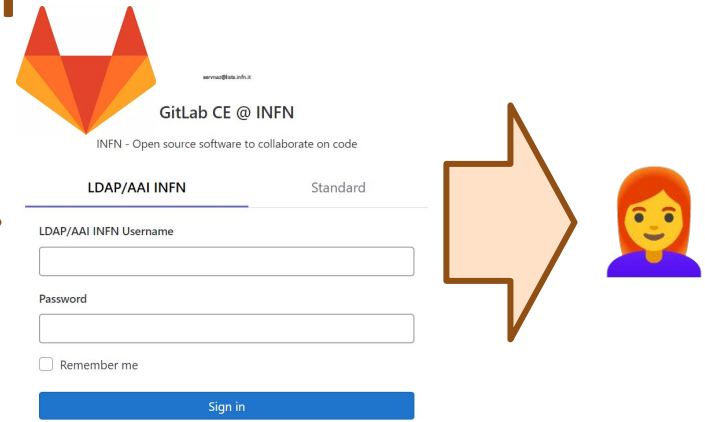
History of the trials of this study.

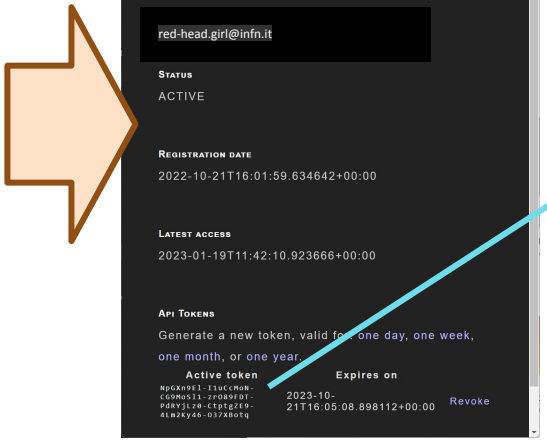


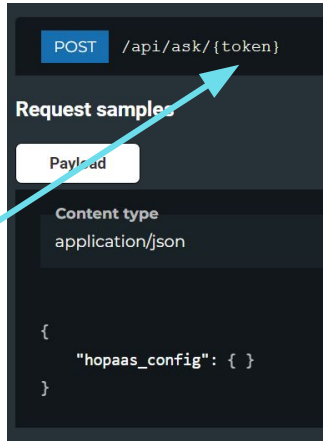
[arXiv:2301.05522]

Token-based user authentication









The study generated by this study will be associated to 

 will find study and trials monitoring in her dashboard

[arXiv:2301.05522]

Optimizing the LHCb fast simulation on Marconi 100

An alpha-test deployment of Hopaas is available on INFN Cloud (hopaas.cloud.infn.it).

It has been used to coordinate intense hyperparameter tuning campaigns on the **Particle Identification models for the LHCb ultra-fast simulation** framework (Lamarr).

Up to 30 GPUs provisioned by CINECA M100 were managed using *slurm* and *Hopaas* to enhance the exploration of the Hyperparameter space of models developed on INFN Cloud instances (ML_INFN).

With this study, **we identified the set of parameters the most relevant to improve models.**

Sharing the status of the study between interactive instances (Jupyter) and batch jobs (M100) was a crucial feature enabling for testing, debugging and other investigations.

Dutiful remarks and conclusion

- **Developing web applications is not our job.**
- We made an effort to deploy a secure back-end well integrated with INFN Cloud, but the service did not undergo any scrutiny by computing security experts.
- Hopaas front-end is simplistic. No admin panels, no resource sharing, no custom documentation of studies and trials, no interface with storage/registries…

Hopaas was developed to solve a very common problem in a very specific context, and it worked exceptionally well for our application.

We would love to keep using (and developing) this tool for distributed gradient-less optimizations through resources from multiple-provider in the ICSC era.

If becoming an asset for ICSC, **professional developers** should be assigned to the project and migration to FOSS solutions at least for model tracking and monitoring re-evaluated.