MASSIVEFOLD

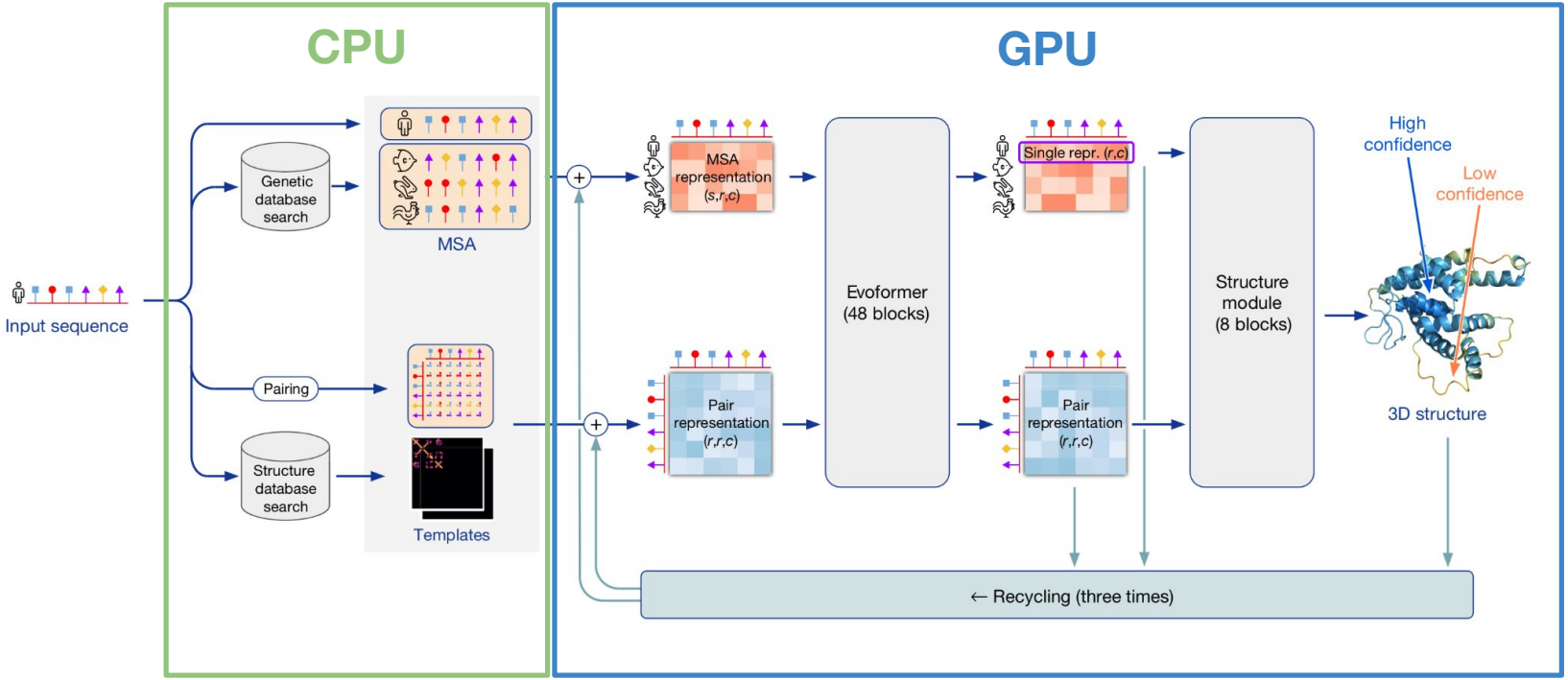
Parallelized AlphaFold2 and AlphaFold3

IFB training - AlphaFold and beyond - 10th to 12th December 2025

Guillaume Brysbaert
CNRS Research engineer - Bioinformatics
UGSF - Lille - France
guillaume.brysbaert@univ-lille.fr

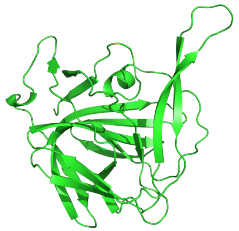
Nessim Raouraoua
Phd student - Bioinformatics
UGSF - Lille - France
nessim.raouraoua@univ-lille.fr

Process



Jumper J et al, Nature, 2023, https://doi.org/10.1038/s41586-021-03819-2

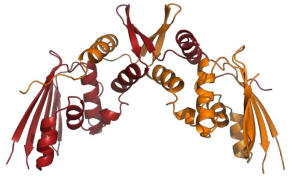
Monomer = 1 sequence



- Published in July 2021

- 5 neural network models * 1 predicted structure = 5 predictions

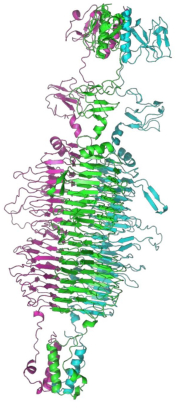
Multimer = x sequences



- Published end 2021 / beginning 2022 (in bioRxiv)

- Since the first release, 3 versions => 3 NN versions * 5 NN models = 15 NN models

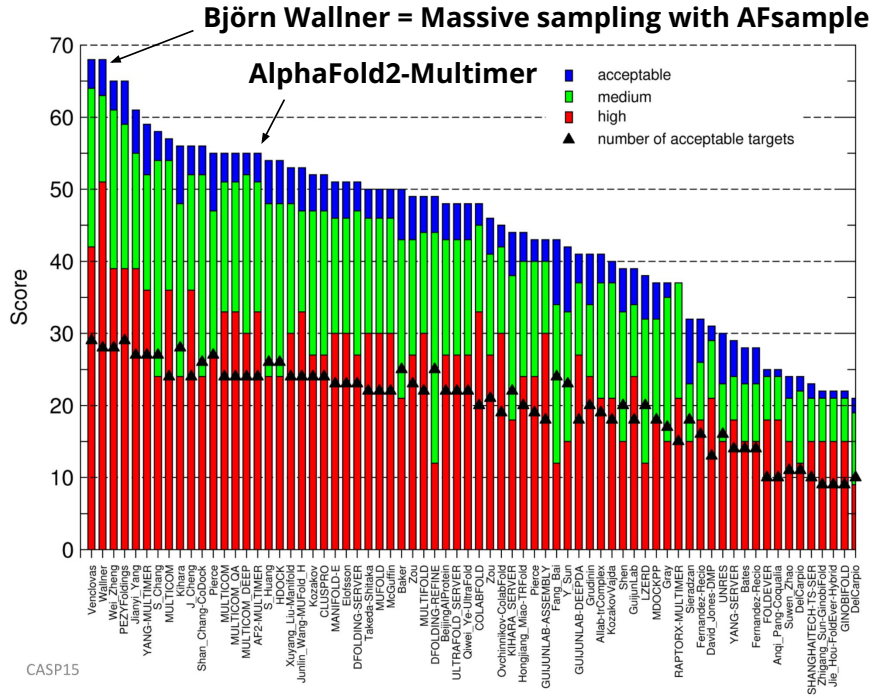
- 5 predicted structures per NN models



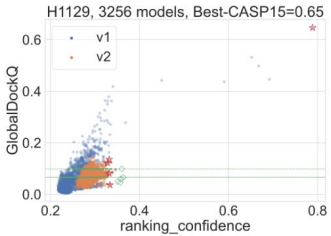
=> v2.1 (2021-10-27) => 5 NN models * 5 predicted structures = 25 predictions
=> v2.2 (2022-03-02) => 5 NN models * 5 predicted structures = 25 predictions
=> **v2.3** (2022-12-06) => 5 NN models * 5 predicted structures = 25 predictions

CASP15 - Dec 2022

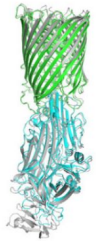
Protein assemblies



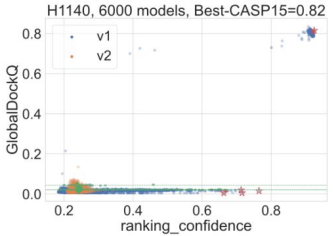
CASP15 - Dec 2022



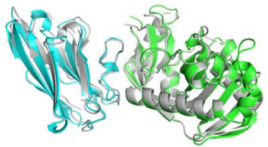
(A) H1129



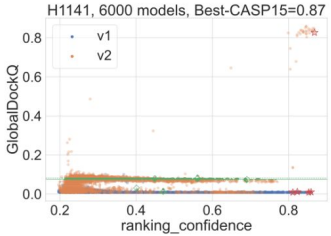
(B) H1129



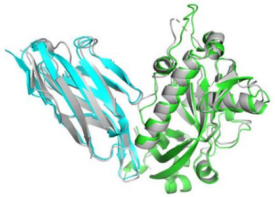
(C) H1140



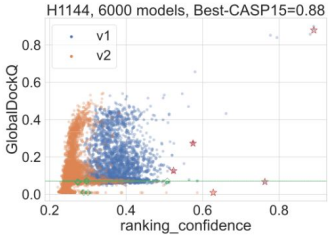
(D) H1140



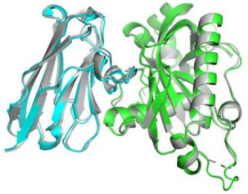
(E) H1141



(F) H1141



(G) H1144



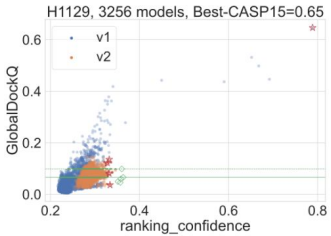
(H) H1144

Massive sampling:

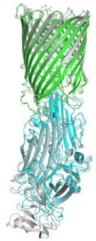
Björn Wallner, Proteins, 2023

- thousands of predictions
- diversity parameters: neural network version, random seed, templates, recycles, dropout

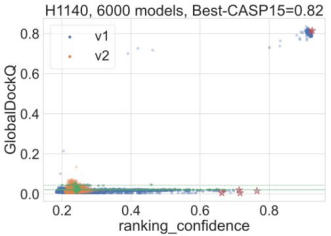
CASP15 - Dec 2022



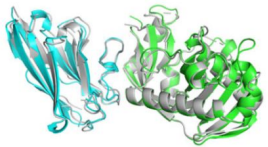
(A) H1129



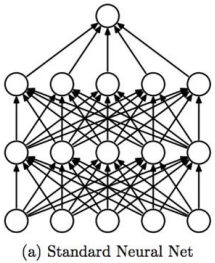
(B) H1129



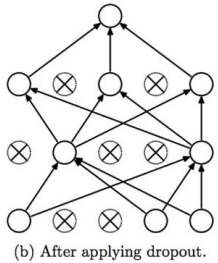
(C) H1140



(D) H1140

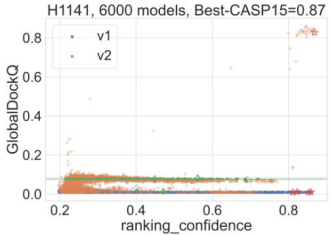


(a) Standard Neural Net

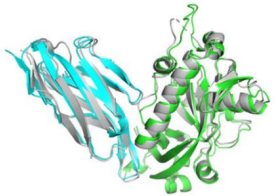


(b) After applying dropout.

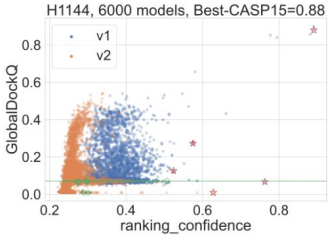
Srivastava, Nitish, et al. "Dropout: a simple way to prevent neural networks from overfitting", JMLR 2014



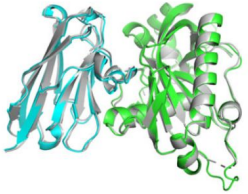
(E) H1141



(F) H1141



(G) H1144



(H) H1144

Björn Wallner, Proteins, 2023

Massive sampling:

- thousands of predictions
- diversity parameters: neural network version, random seed, templates, recycles, dropout

Limitations:

- cost in GPU hours
- management of such a large computation

MassiveFold

Started in March 2023 (GPU Hackathon at IDRIS with NVIDIA)

Nessim Raouraoua
Marc Lensink
Guillaume Brysbaert

Claudio Mirabello
Björn Wallner

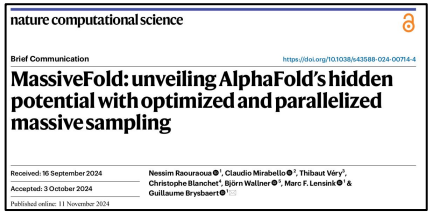
MUDIS4LS
Christophe Blanchet

IDRIS
Supercomputing cluster Jean Zay
Thibaut Véry
Bertrand Cabot

Goals:

- Update **AFsample** => **AFmassive**, to use on the french national cluster
- Optimization of the computing through **parallelization**

Developed in the context of the PIA3 Equipex+ MUDIS4LS (WP4) led by the IFB



* **AFmassive** is an extended version of **AlphaFold2**

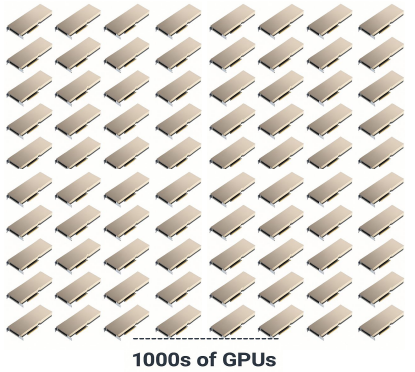
Code available on https://github.com/GBLille/MassiveFold

v1.5 installed on:

MASSIVEFOLD

UGSF server

Univ Lille cluster



CNRS Jean Zay cluster (IDRIS)

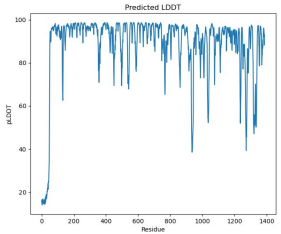
MassiveFold is easy to use

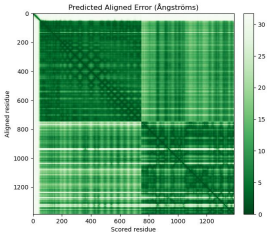
- Uniformisation of the usage for AF2/AF3/CF

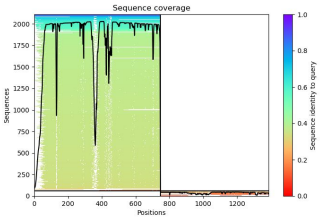
```
./run_massivefold.sh -s input/H1267.fasta -r afm_default -f AFmassive_params.json
./run_massivefold.sh -s input/H1267.fasta -r cf_default  -f ColabFold_params.json
./run_massivefold.sh -s input/H1267.fasta -r af3_default -f AlphaFold3_params.json
```

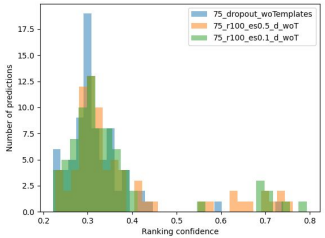
sequence **run name** **parameter file**

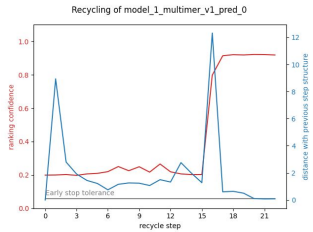
- All outputs are standardized
- Same plots for all tools:

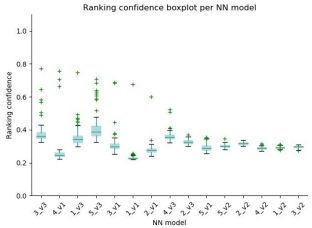


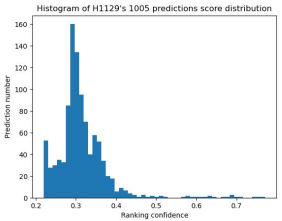


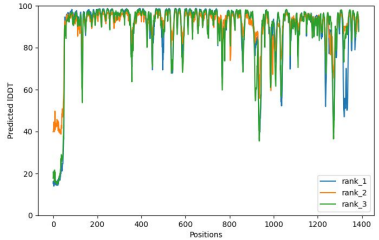


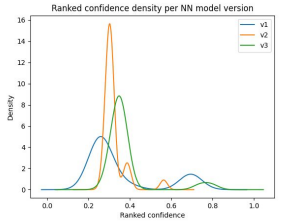




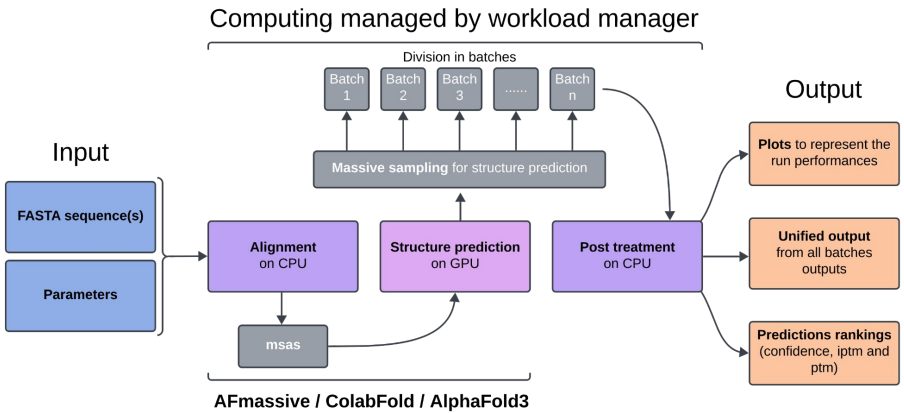








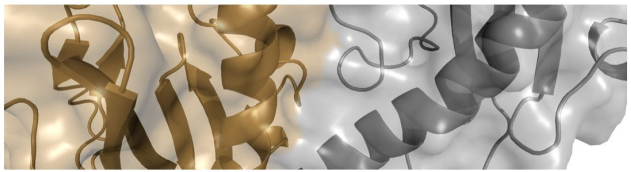
```
MassiveFold
├── massivefold/
└── massivefold_runs/      => where to run MassiveFold
    ├── AFmassive_params.json
    ├── ColabFold_params.json
    ├── AlphaFold3_params.json
    ├── headers/
    ├── input/                  => fasta files for monomers and multimers
    ├── log/                    => logs files (to debug in case of crashes)
    ├── output/
    |   └── H1140/
    |       ├── af3_default/
    |       ├── cf_default/
    |       └── afm_default/
    └── run_massivefold.sh      => executable (-h for help)
```



AFmassive v1.1.5

- Updated AFsample (Björn Wallner)
- AlphaFold2 with exposed parameters, in particular:
  dropout (with custom rates), templates, recycles, early stop tolerance

README Contributing Apache-2.0 license



DOI 10.5281/zenodo.13869932

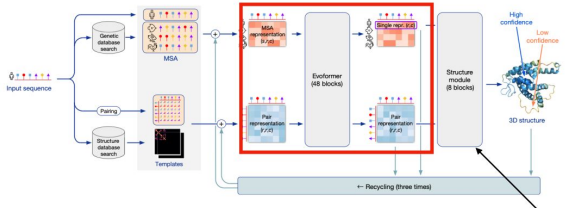
AFmassive

Table of contents

AFmassive is a modified AlphaFold version that integrates diversity parameters for massive sampling, as a derivative version of Björn Wallner's AFsample.
It was extended from DeepMind's AlphaFold v2.3.2.

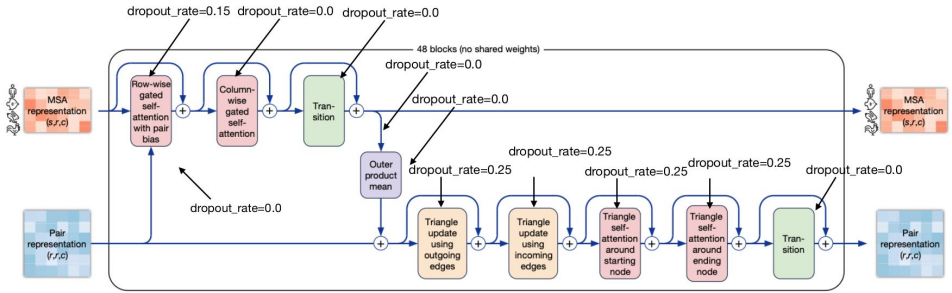
AFmassive was designed to be optimally used with MassiveFold for parallel computing but can also be used outside MassiveFold.
The parameters added to the original DeepMind's AlphaFold are described below.

https://github.com/GBLille/AFmassive



Björn Wallner, CASP15, 2022

dropout_rate=0.1



ColabFold - 2021

- Based on AlphaFold2 but faster (can also run ESMFold and other tools in beta)
- Alignments made with MMseqs2 on the ColabFold database
- Can be used through a Colab notebook or a version can be installed locally
- Splits alignments, structure prediction and relaxation

ColabFold - v1.5.5

For details of what was changed in v1.5, see change log!

Making Protein folding accessible to all via Google Colab!

Notebooks	monomers	complexes	mmseqs2	jackhmmer	templates
AlphaFold2_mmseqs2	Yes	Yes	Yes	No	Yes
AlphaFold2_batch	Yes	Yes	Yes	No	Yes
AlphaFold2 (from Deepmind)	Yes	Yes	No	Yes	No
relax_amber (relax input structure)					
ESMFold	Yes	Maybe	No	No	No
BETA (in development) notebooks					
RoseTTAFold2	Yes	Yes	Yes	No	WIP
Boltz	Yes	Yes	Yes	No	No
BioEmu	Yes	No	Yes	No	No
OmegaFold	Yes	Maybe	No	No	No
AlphaFold2_advanced_v2 (new experimental notebook)	Yes	Yes	Yes	No	Yes

nature methods
BRIEF COMMUNICATION
https://doi.org/10.1038/s41592-022-01488-1
Check for updates

OPEN

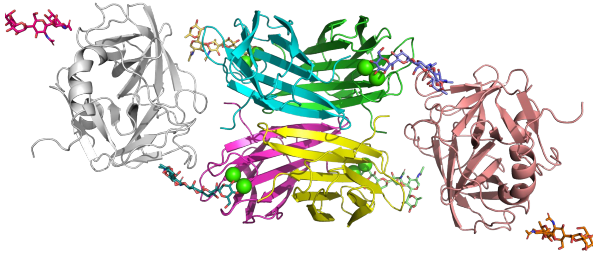
ColabFold: making protein folding accessible to all

Milot Mirdita[1,10] ✉, Konstantin Schütze[2], Yoshitaka Moriwaki[3,4], Lim Heo[5], Sergey Ovchinnikov[6,7,10] ✉ and Martin Steinegger[2,8,9,10] ✉

AlphaFold3 in MassiveFold

With AlphaFold3 locally installed, MassiveFold have features not allowed on the webserver

- possibility to use any ligand as **SMILES**, **CCD codes** or **IUPAC** code (glycans)



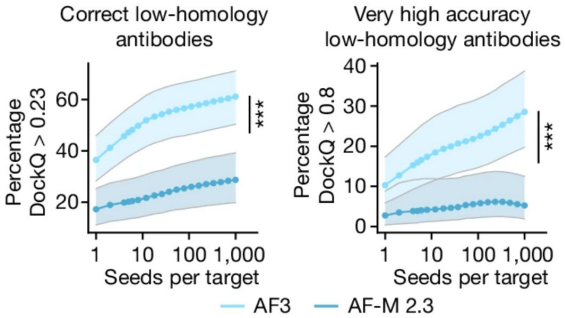
- unlimited ligand screening

- numerous PTMs, in particular glycosylation without size limit with IUPAC code, *e.g.*:

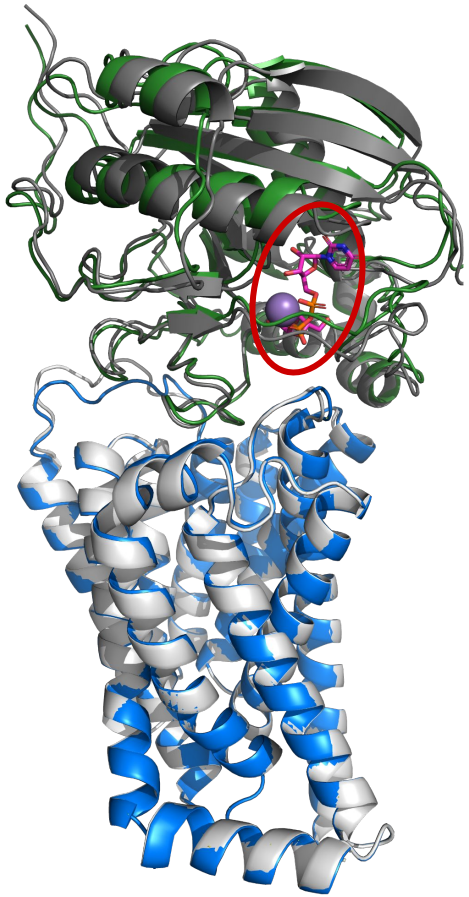
  Gal(1-4)GlcNAc(1-2)Man(1-3)[Gal(1-4)GlcNAc(1-2)[Gal(1-4)GlcNAc(1-6)]Man(1-6)]Man(1-4)GlcNAc(1-4)[Fuc(1-6)]GlcNAc

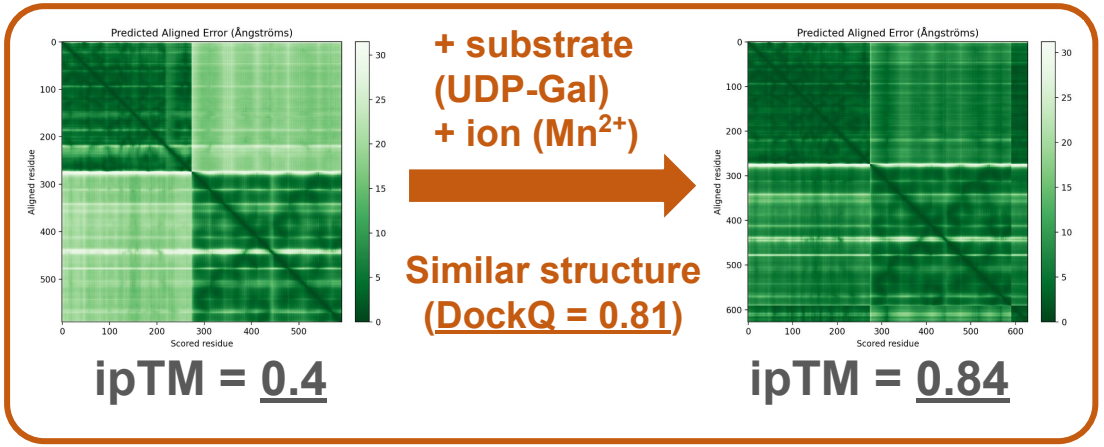
- diversity parameters: unpaired/paired msas, templates, number of samples per seed...

- massive sampling



Caution with AlphaFold3



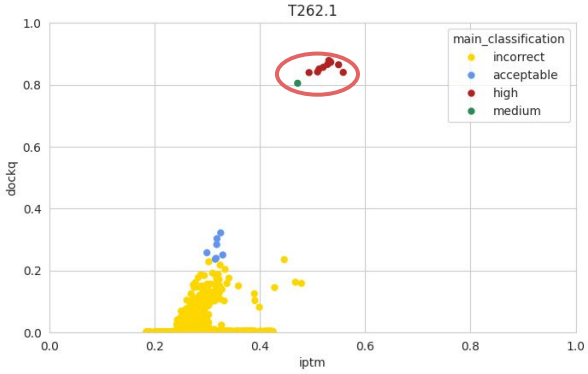


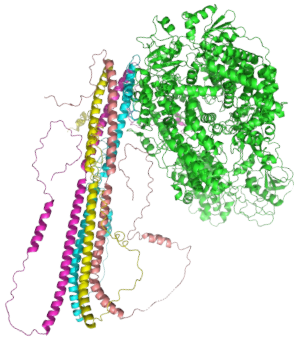
Biological context is key for AlphaFold3 confidence score

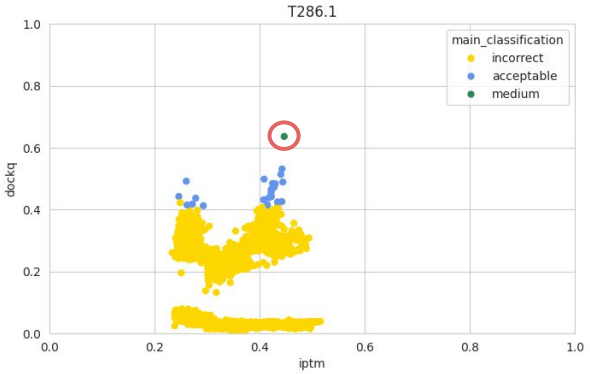
15

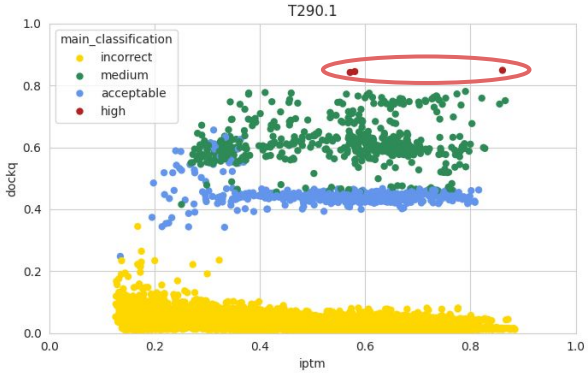
Massive sampling efficiency increase

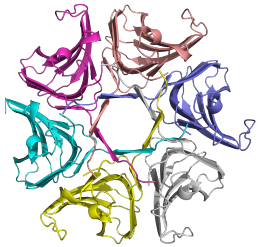
- Results of CASP16 (Dec 2024) experiment
- ‘Hard’ targets have few good models

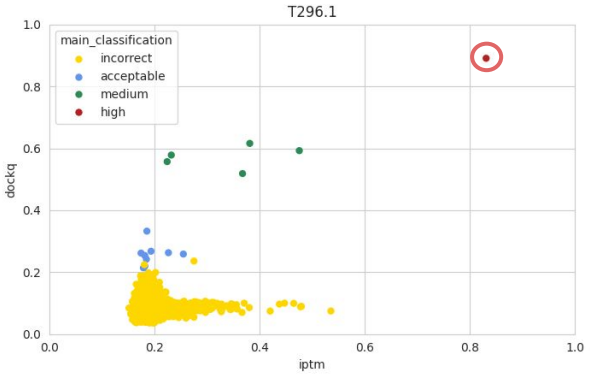


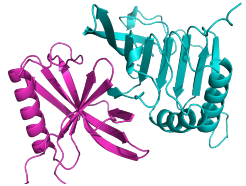












CASP15/CAPRI - Massive sampling vs AlphaFold3

Wallner's massive sampling vs AlphaFold3 on selected targets

Target	Top-1 DockQ Massive sampling	Top-1 DockQ AlphaFold3
H1129	**0.647**	0.059
H1140	0.819	**0.849**
H1141	**0.823**	0.030
H1144	**0.884**	0.317
T1173	**0.907**	0.043
T1187	**0.892**	0.102
H1167	0.064	**0.163**
H1168	0.549	**0.632**

Raouraoua Nessim et al. MassiveFold: unveiling AlphaFold's hidden potential with optimized and parallelized massive sampling. 2024. *Nature Computational Science*, https://www.nature.com/articles/s43588-024-00714-4 (Supplementary information)

Take-home-messages

MASSIVEFOLD

- **AlphaFold** has revolutionized structural bioinformatics but the computation is sequential
- **MassiveFold** optimizes the computation for massive sampling by parallelizing the inference
- **MassiveFold** can run AFmassive, ColabFold and AlphaFold3
- With **AlphaFold3 addition**, new features are available in **MassiveFold**
  - Only in MassiveFold!
- **MassiveFold** is ready for massive sampling

Acknowledgments