AQuaRef: QM-based atomic model refinement becomes reality

Pavel Afonine

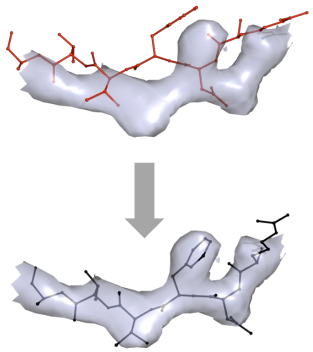
Lawrence Berkeley National Laboratory (LBNL)

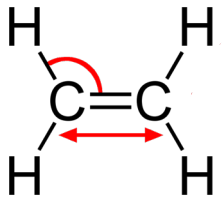
September 26th, 2024
BNL

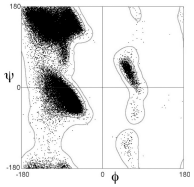
Atomic model refinement

T = T_{DATA} + $w * T_{RESTRAINTS}$

Optimize consensus between model-to-data fit and prior information







Restraints

$$T = T_{DATA} + w * T_{RESTRAINTS}$$

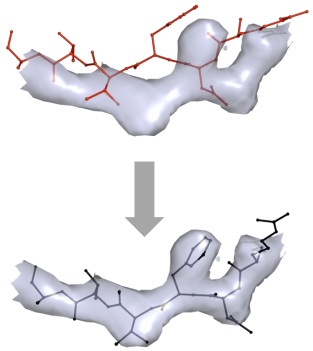
$$T_{RESTRAINTS} = T_{BOND} + T_{ANGLE} + T_{DIHEDRAL} + T_{PLANE} + T_{REPULSION} + T_{CHIRALITY}$$

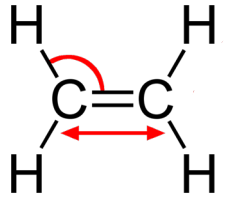
- Too simplistic:
  - No attraction terms (electrostatics, etc)
  - Not using information about protein structure (secondary structure, rotamers)
  - Limited to tabulated entities in the libraries (e.g., Monomer Library, GeoStd)

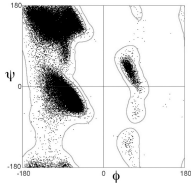
A better solution

T = **T_{DATA}** + **w * $T_{RESTRAINTS}$**

Optimize consensus between model-to-data fit and... common sense





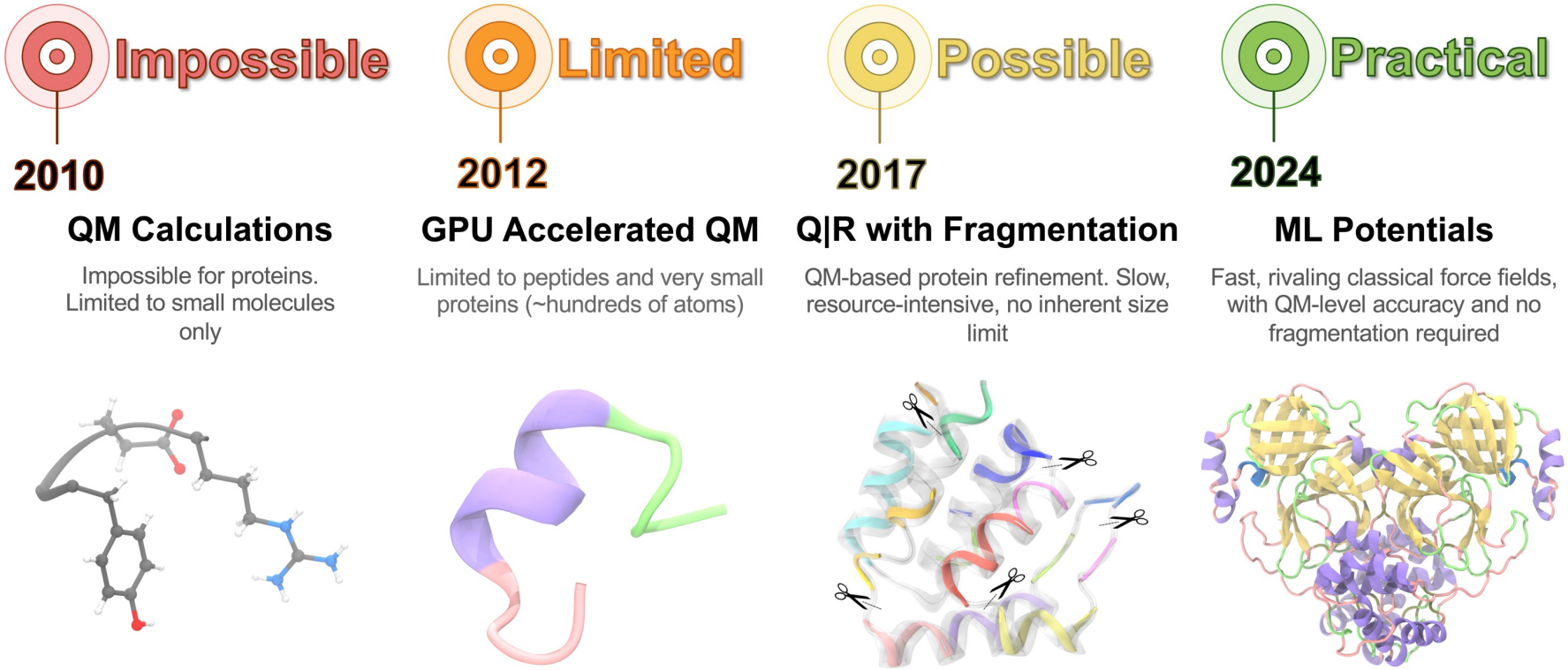


Bonds, angles, planes, torsions, chirality, non-bonded repulsion

Replace with energies/gradients from QM calculations

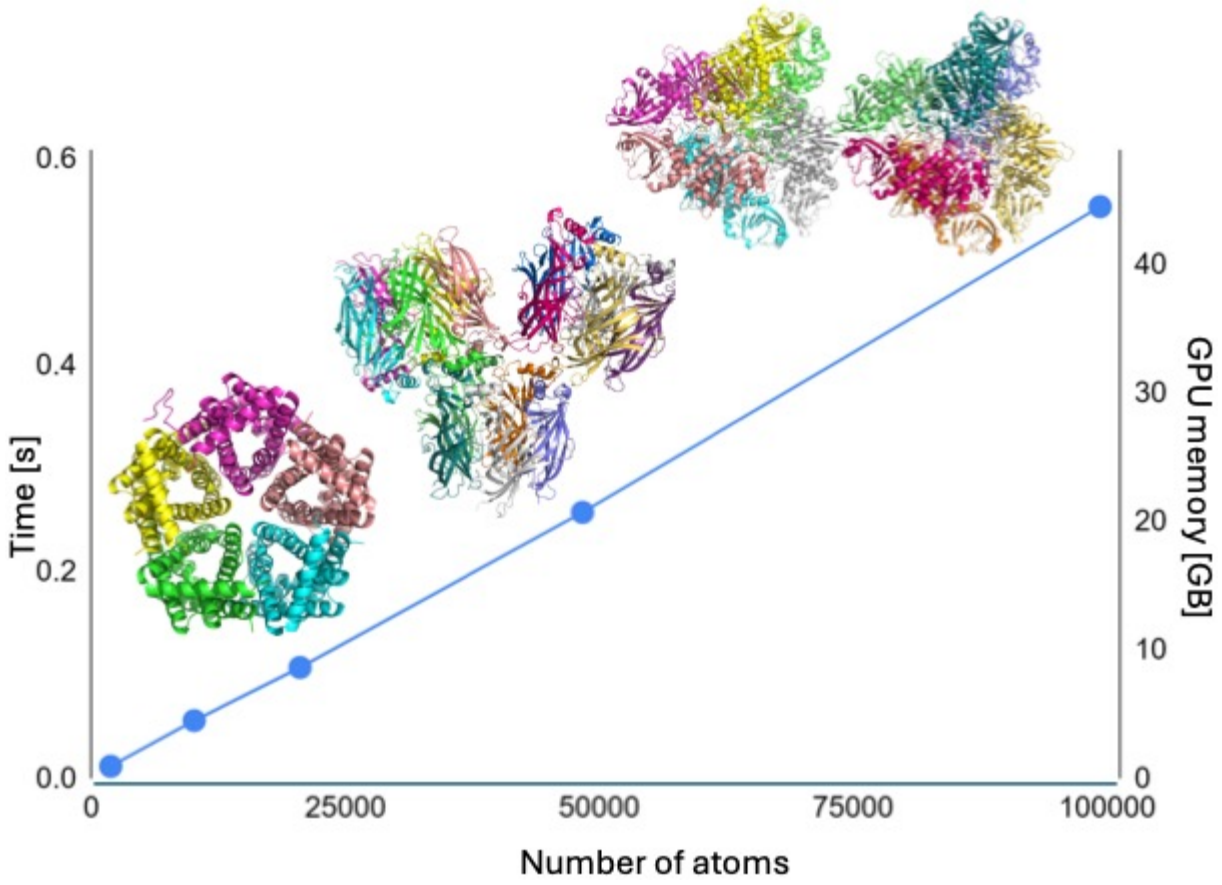
AQuaRef: AI-enabled Quantum Refinement

QM calculations for proteins structures*: history of progress



* **Whole protein** structures, **not** QM/MM and other ONIOM like methods!

Time & Memory Scaling: single energy calculation



Q|R: Quantum Refinement project

Blend of expertise and background

Crystallography methods + Software development + QM expertise

www.qrefine.com

Q|R Quantum Refinement

http://qrefine.com qrefine@googlegroups.com

Q|R @ GitHub

Pending.AI (Australia)

Mark Waller

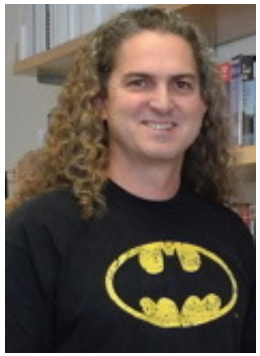
Holger Kruse

Uni of Wrocław

Malgorzata Biczysko

LBNL (USA)

Pavel Afonine



Nigel Moriarty

The Team: crystallography + chemistry + AI expertise

Carnegie Mellon University


Olexandr Isayev


Roman Zubatyuk


Hatice Gökcan

Uni. of Florida


Adrian Roitberg

Blend of expertise and background

Crystallography methods + **Software development** + **QM expertise** + **ML / AI expertise**

Pending AI


Marl Waller


Holger Kruse

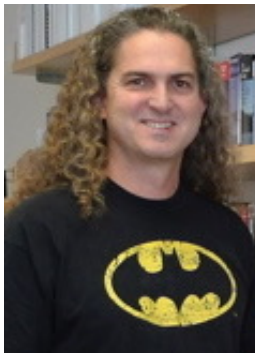
Uni of Wrocław


Malgo Biczysko

LBNL


Billy Poon


Pavel Afonine


Nigel Moriarty

- Limitations
  - Proteins only. Including other types of molecules is work in progress
  - No alternative conformations
- Timing
  - Same as standard refinement or 2-3 times slower
- Hardware requirement:
  - GPU laptop or a workstation
- Availability
  - AQuaRef is now available in Phenix nightly builds and Q|R

Availability

phenix-online.org/download/phenix/nightly/?version=dev-5430

Installers for Phenix version dev-5430 (development version)

For a summary of possible errors and/or bugs, please see the build and testing logs:

ci, mac-intel-osx-x86_64, intel-linux-2.6-x86_64-centos6

Version documentation
Summary of changes

Linux

Kernel 2.6 (64-bit; CentOS 6) [**download command-line installer**]

CentOS 6 is the earliest linux distribution supported. This installer will work with newer linux distributions.

macOS

OS X 10.9+ (64-bit Intel) [**download point-and-click installer**]

OS X 10.9+ (64-bit Intel) [**download command-line installer**]

The graphical installer will install to /Applications; the command-line installer may be relocated with the --prefix argument.
On macOS 12 and later, you will need to run "xattr -c " to avoid errors about untrusted files
For Phenix 1.20.1 and before, the Phenix GUI will crash on macOS Sonoma (14). Please download this fix to fix your Phenix installation. Extract the tarball and run "sudo ./fix_wxpython.sh /Applications/phenix-1.20.1-4487". You will need administrative access to modify files in /Applications.

Windows (partially supported)

For full functionality, you should run the Python 2.7 linux installer on Windows using WSL. Instructions for setting up Phenix with WSL can be found here.

A native Windows build is only available for Python 3, but you will encounter crashes. See below for the link to the Windows installer.

The next official release of Phenix will be in Python 3 and Windows will have a working installer again.

Python 3.9 (Alpha) - CUDA 11 and CUDA 12 support only for quantum refinement on Linux, general Phenix GUI is still in testing and does not have full functionality

Linux (CUDA 12) [**download command-line installer**]

Linux (CUDA 11) [**download command-line installer**]

Python 3.9 (Alpha) - For testing and development, do not expect full functionality in the Phenix GUI

Linux [**download command-line installer for development builds and modifications (contains "modules" directory)**]

Linux [**download command-line installer**]

macOS (Intel) [**download graphical installer**]

macOS (Intel) [**download command-line installer**]

macOS (Apple Silicon) [**download graphical installer**]

macOS (Apple Silicon) [**download command-line installer**]

Windows [**download graphical installer**]