Icolos: a workflow manager for structure-based post-processing of de novo generated small molecules

Bioinformatics, 2022, 1–2
https://doi.org/10.1093/bioinformatics/btac614
Advance Access Publication Date: 8 September 2022
Applications Note

Structural bioinformatics

Icolos: a workflow manager for structure-based post-processing of *de novo* generated small molecules

J. Harry Moore[1,*], Matthias R. Bauer[2], Jeff Guo[1], Atanas Patronov [1], Ola Engkvist[1,3] and Christian Margreitter [1,*]

[1]Molecular AI, Discovery Sciences, R&D, AstraZeneca, Gothenburg 431 83, Sweden, [2]Structure & Biophysics, Discovery Sciences, R&D, AstraZeneca, Cambridge CB2 8PA, UK and [3]Computer Science and Engineering, Chalmers University of Technology, Gothenburg 412 96, Sweden

*To whom correspondence should be addressed.

Associate Editor: Alfonso Valencia

Received on February 2, 2022; revised on August 6, 2022; editorial decision on September 2, 2022; accepted on September 7, 2022

Abstract

Summary: We present Icolos, a workflow manager written in Python as a tool for automating complex structure-based workflows for drug design. Icolos can be used as a standalone tool, for example in virtual screening campaigns, or can be used in conjunction with deep learning-based molecular generation facilitated for example by REINVENT, a previously published molecular *de novo* design package. In this publication, we focus on the internal structure and general capabilities of Icolos, using molecular docking experiments as an illustrative example.

Availability and implementation: The source code is freely available at https://github.com/MolecularAI/Icolos under the Apache 2.0 license. Tutorial notebooks containing minimal working examples can be found at https://github.com/MolecularAI/IcolosCommunity.

Contact: harry.moore@astrazeneca.com or christian.margreitter@gmail.com

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Structure-based computational methods provide significant predictive insight and see widespread use in drug discovery from hit discovery to lead optimization. However, manual execution of multistep workflows is inefficient, labor intensive and error prone, especially when stitching together multiple programs *ad hoc* using scripting languages such as bash.

Existing workflow managers, such as KNIME (Berthold *et al.*, 2009) and Pipeline Pilot (BIOVIA-Dassault Systèmes, 2022), are commonly used to automate such tasks providing a user-friendly interface, however, we required a platform which provides more flexibility, customization and control over each step, is better suited to running long, highly parallelized workflows (e.g. free energy and QM calculations) and would seamlessly integrate with our molecular *de novo* design tool, REINVENT, to construct complex scoring components through vendor-agnostic workflows, while being sufficiently flexible to support rapid prototyping. In particular, to handle highly parallelized and computationally expensive workflows, we required a platform with extensive job control capabilities, allowing workflows to be scaled from execution on a local workstation to a SLURM cluster or Cloud compute provider with minimal configuration change, whilst retaining minimal setup and deployment overhead. Lastly, the ability to document and share these workflows with colleagues in various departments was key, facilitating efficient knowledge transfer.

Here, we present Icolos, a modular, flexible and extensible workflow manager that provides a unified interface to a host of common commercial and open-source computational packages, encompassing docking, molecular dynamics, binding free energy and quantum mechanical calculations. Icolos has a built-in REINVENT integration and has been used in-house both to incorporate complex structural calculations into the agent's feedback loop, and as a standalone workflow manager for post-processing results (see Supplementary Material for example workflows). We achieve efficient scaling through parallelization of computationally demanding calculations and results in agreement with manually executed workflows, often at a fraction of the runtime of previous implementations relying on shell scripts or submission from a GUI.

2 Software implementation

Icolos workflows are constructed as a list of elementary 'steps' which defines the flow of information through the workflow. Over 40 individual steps are currently supported, covering a wide variety of commercial and open-source software, which can be combined in arbitrary order (see the Supplementary Material for a complete list). Templates for common workflows are available and can be readily extended or adapted. In principle, any program that provides a command-line executable or Python API could be incorporated as a workflow step.

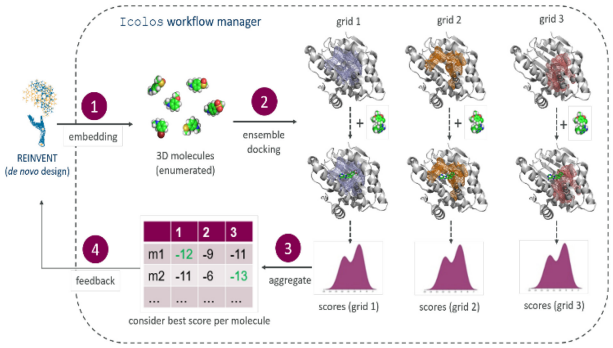

Fig. 1. Graphical summary of ensemble docking workflow implemented in Icolos

Workflow configurations are specified in a JSON file, with each step defined by a standardized set of fields, controlling the execution environment, parallelization scheme, error handling and settings to control both the underlying program and the step's execution. Typically, all underlying command-line options of the backends are accessible which allows a high degree of control. Since many workflows implement virtual screening capabilities on libraries of compounds, Icolos efficiently handles molecules in a multi-tier data structure comprised of compounds, enumerations (e.g. tautomeric states or stereo-isomers) and conformers. This is based on the RDKit Mol object, and keeps track of computed properties as the workflow progresses (Landrum *et al.*, 2022).

This provides the basis for flexible write-out capabilities in a variety of standard formats and allows for efficient parallelization at both the step and workflow level to leverage HPC resources.

In general, steps that perform computations on a set of compounds can be parallelized across multiple cores, and subjobs can be either run directly utilizing the master job's resources, or on a Slurm cluster through the integrated submission and monitoring interface. This enables execution of heterogeneous workflows requiring both CPU and GPU resources (for example a combined docking and molecular dynamics workflow), with efficient use of cluster resources. As an illustration, we introduce an ensemble docking workflow using Icolos, in which ligands are docked against multiple receptor grids, which can be constructed from either different crystal structures or representative structures obtained by clustering a molecular dynamics trajectory. In our experience, ensemble docking can lead to substantial ligand enrichment compared to a single grid, especially where there is significant receptor flexibility. The full workflow consists of the following steps and is summarized in Figure 1:

- Ligand embedding, enumerating possible protonation states, stereo-chemistry and tautomers;
- Docking against mutliple grids; and
- Filtering and reporting back the best score per compound across all grids.

For details and the full workflow configuration files, including the use of alternative ligand preparation and docking backends [such as AutoDock Vina (Eberhardt *et al.*, 2021)], we refer to the Supplementary Material, the examples distributed in the main Icolos repository and the IcolosCommunity repository.

3 Conclusion

We have developed Icolos, a general-purpose workflow manager for structure-based workflows. Icolos has been successfully deployed internally to develop, reproduce and distribute complex workflows in drug discovery projects, and handle complex scoring components for *de novo* molecular generation using REINVENT.

More complex use cases will be benchmarked and described in detail in subsequent publications. In particular, we plan to support AlphaFold (Jumper *et al.*, 2021) to generate structures as a starting point for existing MD workflows.

Acknowledgements

The authors would like to thank Maxime Tarrago, Jon Paul Janet, Martin Packer, Luca Carlino, Magdalena Weber, Linnea Johansson and the AstraZeneca Scientific Computing Platform team for their valuable input to the manuscript.

Financial Support: none declared.

Conflict of Interest: none declared.

References

Berthold,M.R. *et al.* (2009) *KNIME—The Konstanz Information Miner. ACM SIGKDD Explorations Newsletter*. ACM PUB27, New York, NY, USA, pp. 58–61.

BIOVIA-Dassault Systèmes (2022) *PipelinePilot*. Dassault Systèmes, San Diego. https://www.3ds.com/products-services/biovia/resource-center/citations-and-references.

Eberhardt,J. *et al.* (2021) AutoDock vina 1.2.0: new docking methods, expanded force field, and python bindings. *J. Chem. Inf. Model.*, **61**, 3891–3898.

Jumper,J. *et al.* (2021) Highly accurate protein structure prediction with AlphaFold. *Nature*, **596**, 583–589.

Landrum,G. *et al.* (2022) *rdkit/rdkit: 2022_03_2 (Q1 2022) Release*. https://rdkit-discuss.narkive.com/9QGX4Vxh/is-there-a-way-to-cite-rdkit-in-a-paper.