



UNIVERSITY OF LEEDS

This is a repository copy of *BAlaS: fast, interactive and accessible computational alanine-scanning using BudeAlaScan*.

White Rose Research Online URL for this paper:  
<http://eprints.whiterose.ac.uk/156028/>

Version: Accepted Version

---

**Article:**

Wood, CW, Ibarra, AA, Bartlett, GJ et al. (3 more authors) (2020) BAlaS: fast, interactive and accessible computational alanine-scanning using BudeAlaScan. *Bioinformatics*. ISSN 1367-4803

<https://doi.org/10.1093/bioinformatics/btaa026>

---

© The Author(s) 2020. Published by Oxford University Press. All rights reserved. This is an author produced version of a journal article published in *Bioinformatics*. Uploaded in accordance with the publisher's self-archiving policy.

**Reuse**

Items deposited in White Rose Research Online are protected by copyright, with all rights reserved unless indicated otherwise. They may be downloaded and/or printed for private study, or other acts as permitted by national copyright laws. The publisher or other rights holders may allow further reproduction and re-use of the full text version. This is indicated by the licence information on the White Rose Research Online record for the item.

**Takedown**

If you consider content in White Rose Research Online to be in breach of UK law, please notify us by emailing [eprints@whiterose.ac.uk](mailto:eprints@whiterose.ac.uk) including the URL of the record and the reason for the withdrawal request.



[eprints@whiterose.ac.uk](mailto:eprints@whiterose.ac.uk)  
<https://eprints.whiterose.ac.uk/>

---

# BAlaS: fast, interactive and accessible computational alanine-scanning using BudeAlaScan

Christopher W. Wood<sup>1\*</sup>, Amaury A. Ibarra<sup>3</sup>, Gail J. Bartlett<sup>2</sup>, Andrew J. Wilson<sup>4,5</sup>, Derek N. Woolfson<sup>2,3,6</sup> and Richard B. Sessions<sup>3,6\*</sup>

<sup>1</sup>School of Biological Sciences, University of Edinburgh, Roger Land Building, EH9 3FF., <sup>2</sup>School of Chemistry, University of Bristol, Cantock's Close, BS8 1TS., <sup>3</sup>School of Biochemistry, University of Bristol, Medical Sciences Building, University Walk, BS8 1TD, Bristol, UK., <sup>4</sup>School of Chemistry, University of Leeds, Woodhouse Lane, Leeds LS2 9JT, UK., <sup>5</sup>Astbury Centre for Structural Molecular Biology, University of Leeds, Woodhouse Lane, Leeds LS2 9JT, UK., <sup>6</sup>BrisSynBio, University of Bristol, Life Sciences Building, Tyndall Avenue, Bristol BS8 1TQ, UK

\*To whom correspondence should be addressed.

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

## Abstract

**Motivation:** In experimental protein engineering, alanine-scanning mutagenesis involves the replacement of selected residues with alanine to determine the energetic contribution of each side chain to forming an interaction. For example, it is often used to study protein-protein interactions. However, such experiments can be time-consuming and costly, which has led to the development of programs for performing computational alanine-scanning mutagenesis (CASM) to guide experiments. While programs are available for this, there is a need for a real-time web application that is accessible to non-expert users.

**Results:** Here we present BAlaS, an interactive web application for performing CASM via BudeAlaScan and visualizing its results. BAlaS is interactive and intuitive to use. Results are displayed directly in the browser for the structure being interrogated enabling their rapid inspection. BAlaS has broad applications in areas such as drug discovery and protein-interface design.

**Availability:** BAlaS works on all modern browsers and is available through the following website: <https://balas.app>. The project is open source, distributed using an MIT license, and is available on GitHub (<https://github.com/wells-wood-research/balas>).

**Contact:** [chris.wood@ed.ac.uk](mailto:chris.wood@ed.ac.uk), [r.sessions@bristol.ac.uk](mailto:r.sessions@bristol.ac.uk)

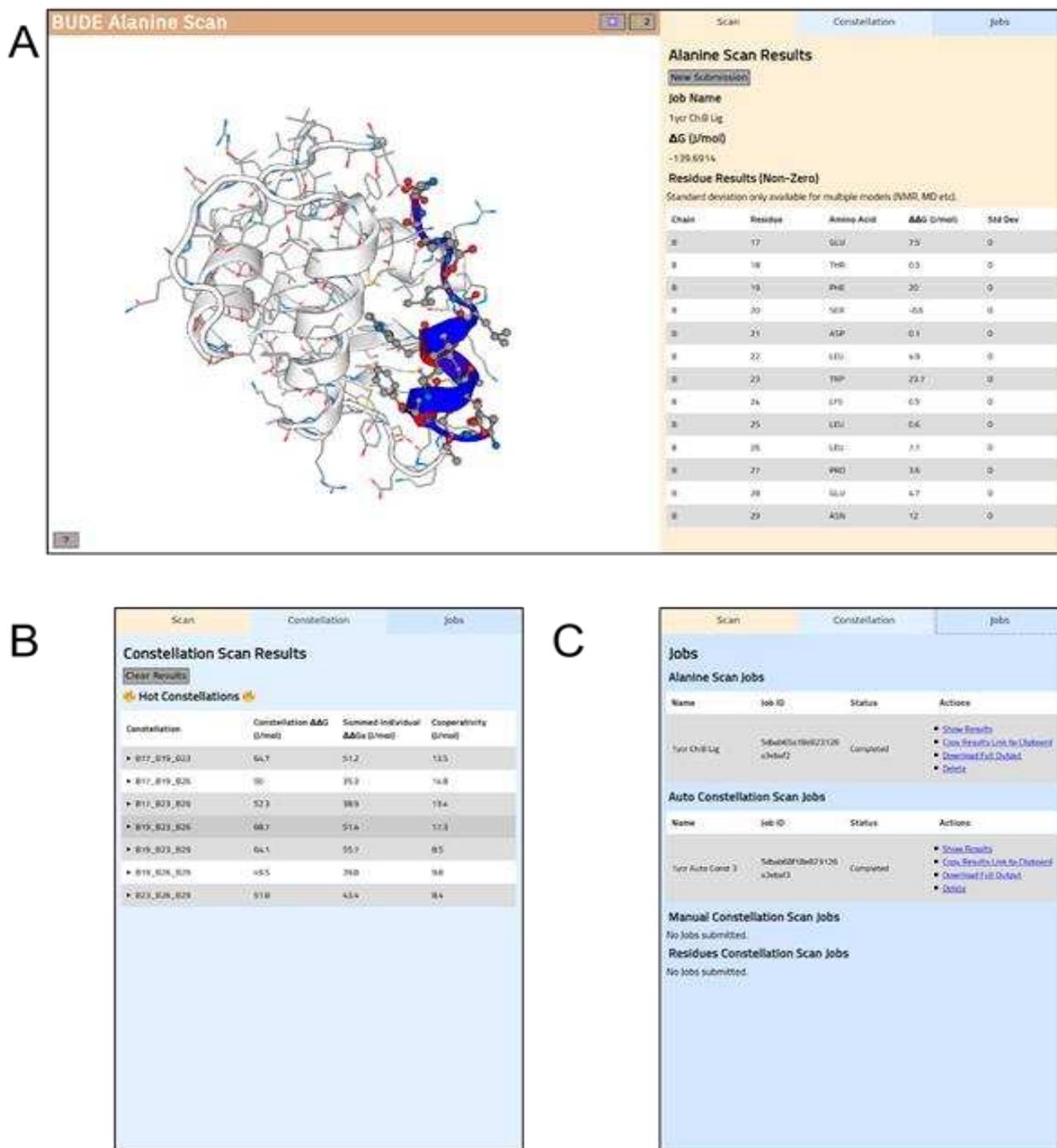
---

## 1 Introduction

Alanine-scanning mutagenesis is an experimental technique where residues in a protein of interest are sequentially mutated to alanine (Cunningham and Wells, 1989). The aim is to identify those residues that contribute the most to the free energy of an interaction. For protein-protein interactions, such residues are commonly referred to as “hot spots” and “hot segments” (Clackson and Wells, 1995; London et al., 2013). Knowledge of such constellations can be used in the development of novel therapeutics that disrupt protein-protein interactions, which are promising targets, given recent successes (Ray-Coquard et al., 2012; Souers et al., 2013; Rosell and Fernández-Recio, 2018).

Experimental alanine-scanning mutagenesis can be costly and time consuming unless directed. Therefore, a computational alternative that can be used to guide experiments is desirable. Since its inception (Massova and Kollman, 1999), a range of methods for performing computational alanine-scanning mutagenesis (CASM) have been presented both on the command line and through webservers (Schymkowitz et al., 2005; Kortemme et al., 2004; Pires et al., 2016; Dehouck et al., 2013). However, there is an unmet need for a fast, interactive and accessible tool that is suitable for non-expert users. We have written BAlaS to meet this need.

user can display the results, create a link to the job that they can share, download the full output, or delete the job.



**Fig. 1. Overview of the BAlaS web application.** (A) The main interface contains a structure visualiser and a panel for submitting jobs and displaying results. The scan panel is visible on the right and displays the results of a CASM job, with a table describing the energetic contribution of each residue. (B) The constellation tab displays energies calculated for each constellation, in this case from a residues mode job. (C) The jobs panel displays all jobs that have been submitted by the user. The user can display the results, create a link to the job that they can share, download the full output, or delete the job.

## 2 Methods

The web application presents a simple user interface (figure 1) that wraps functionality from the BudeAlaScan command-line application, which is described and benchmarked in detail elsewhere (Ibarra et al.,

2019). Briefly, the BudeAlaScan command-line application uses ISAMBARD (Wood et al., 2017) to prepare input files for a custom version of the Bristol University Docking Engine (BUDE) (McIntosh-Smith et al., 2012), which performs CASM using an empirical free-energy forcefield to estimate the free-energy of binding. The accuracy of BudeAlaScan has been experimentally benchmarked and compares favourably with other methods (Ibarra et al., 2019), but is significantly faster than alternatives, making it ideal for use in an interactive web application.

The front end of the BAlaS application is written in Elm (<https://elm-lang.org/>) and handles input validation, job management and results visualisation. Structures are displayed using the PV JavaScript library (Biasini, 2015). The backend architecture is relatively straightforward: NGINX is used to serve the application and the result files; a NoSQL database (MongoDB) is used for the job queue and for storing results; a simple script polls the database and runs BudeAlaScan jobs; and a RESTful API, written using the Flask Python library (<http://flask.pocoo.org/>), is used by the front end to submit jobs and retrieve results. The whole application stack runs inside 3 Docker containers (<https://www.docker.com/>), which enables users to run the web application locally if desired.

The application has three main tabs: Scan, Constellation and Jobs (figure 1). An overview of their functionality follows, but they are also explained in detail in the application through an extensive tutorial. Scan is the main entry point for the application and is where the user can load their structure file, which, currently, must be in PDB format and with canonical amino acids only. Once the structure is loaded, the user can specify the “receptor” chain(s) and the “ligand” chain(s). It is the latter that is(are) mutated during the scan. The energetic contribution of a residue to the interaction energy is calculated using the following equation:

$$\Delta\Delta G_{res} = \Delta G_{int_{mut}} - \Delta G_{int_{wt}}$$

where  $\Delta G_{int}$  is the interaction energy between the “ligand” and the “receptor”. Note that residues making a significant contribution to the overall interface energy will have a positive  $\Delta\Delta G_{res}$ .

The Constellation mode allows the user to combine multiple alanine mutations to find regions of the “ligand” that interact cooperatively with the “receptor”, which we call hot constellations; i.e., the  $\Delta\Delta G$  when all residues are simultaneously mutated to alanine, is greater than the sum of the  $\Delta\Delta G$  values when each residue is mutated to alanine individually. This is useful for highlighting regions of the “ligand” that are key to forming the interaction with the “receptor”. Constellation jobs can be run in one of three modes: manual, residues and auto. In manual mode the user specifies all the residues in a constellation. In residues mode the user selects a list of residues and a constellation size, and all permutations of the residues equal to the size of the constellation are evaluated. In auto mode the user defines a constellation size, a  $\Delta\Delta G$  cut-off (kJ/mol) and a C $\alpha$ -C $\alpha$  distance cut-off (Å). Permutations of all residues with a  $\Delta\Delta G$  above the cut-off are created for the defined constellation size, provided that the residues are within the C $\alpha$ -C $\alpha$  distance cut-off of each other.

Finally, the jobs tab is used to manage the user’s jobs as well as retrieve and share results. Full results from the command-line application can be downloaded as a zip file and contains the  $\Delta\Delta G$  data in text formats, as well as scripts for displaying the results in the molecular graphics program Chimera. Users can plot the graphical data (json files) contained in

the output easily through the `replotAlaScan.py` script in the BudeAlaScan package.

## Conclusions

The BAlaS application allows users, including those who are not necessarily expert in computational modelling, to run and interpret the results of CASM. The web application provides a broad range of modes of operation that cover most use cases, and as it is built upon the powerful BudeAlaScan command-line application (Ibarra et al., 2019), the user can easily move to this to extend and scale up their analysis as required.

## Acknowledgements

CWW thanks Jack O’Shea for feedback on the web application and the School of Biological Sciences IT team for infrastructure support. We thank the Advanced Computing Research Centre at Bristol for infrastructure support and members of the PoPPI team for useful discussions during the development of BAlaS

## Funding

CWW is supported by an Engineering and Physical Sciences Research Council Fellowship (EP/S003002/1) to CWW. DNW is supported by a European Research Council Advanced Grant (340764), a Biotechnology and Biological Sciences Research Council Research Grant (BB/R00661X/1) and held a Royal Society Wolfson Research Merit Award. GJB, AAI, RBS, AJW and DNW are supported by an Engineering and Physical Sciences Research Council Programme Grant (PoPPI, EP/N013573/1).

*Conflict of Interest:* none.

## References

- Biasini, M. (2015) PV is a WebGL-based viewer for proteins and other macromolecular structures (<https://zenodo.org/record/20980>)
- Clackson, T. and Wells, J.A. (1995) A hot spot of binding energy in a hormone-receptor interface. *Science*, 267, 383–386.
- Cunningham, B.C. and Wells, J.A. (1989) High-resolution epitope mapping of hGH-receptor interactions by alanine-scanning mutagenesis. *Science*, 244, 1081–1085.
- Dehouck, Y. et al. (2013) BeAtMuSiC: prediction of changes in protein-protein binding affinity on mutations. *Nucleic Acids Res*, 41, W333–W339.
- Ibarra, A.A. et al. (2019) Predicting and Experimentally Validating Hot-Spot Residues at Protein-Protein Interfaces. *ACS Chem Biol*, 14, 2252–2263.
- Kortemme, T. et al. (2004) Computational Alanine Scanning of Protein-Protein Interfaces. *Sci. STKE*, 2004, p12–p12.
- London, N. et al. (2013) Druggable protein-protein interactions – from hot spots to hot segments. *Current Opinion in Chemical Biology*, 17, 952–959.
- Massova, I. and Kollman, P.A. (1999) Computational Alanine Scanning To Probe Protein-Protein Interactions: A Novel Approach To Evaluate Binding Free Energies. *J. Am. Chem. Soc.*, 121, 8133–8143.
- McIntosh-Smith, S. et al. (2012) Benchmarking Energy Efficiency, Power Costs and Carbon Emissions on Heterogeneous Systems. *The Computer Journal*, 55, 192–205.
- Pires, D.E.V. et al. (2016) mCSM-lig: quantifying the effects of mutations on protein-small molecule affinity in genetic disease and emergence of drug resistance. *Scientific Reports*, 6, 29575.
- Ray-Coquard, J. et al. (2012) Effect of the MDM2 antagonist RG7112 on the P53 pathway in patients with MDM2-amplified, well-differentiated or dedifferentiated liposarcoma: an exploratory proof-of-mechanism study. *The Lancet Oncology*, 13, 1133–1140.
- Rosell, M. and Fernández-Recio, J. (2018) Hot-spot analysis for drug discovery targeting protein-protein interactions. *Expert Opinion on Drug Discovery*, 13, 327–338.

Schymkowitz, J. et al. (2005) The FoldX web server: an online force field. *Nucleic Acids Res*, 33, W382–W388.

Souers, A.J. et al. (2013) ABT-199, a potent and selective BCL-2 inhibitor, achieves antitumor activity while sparing platelets. *Nature Medicine*, 19, 202–208.

Wood, C.W. et al. (2017) ISAMBARD: an open-source computational environment for biomolecular analysis, modelling and design. *Bioinformatics*, 33, 3043–3050.