IRIS-HEP Fellowship Proposal

Ed van Bruggen

University of Washington, Seattle

Duration: January 2020 - June 2021
WBS: Analysis Systems (AS)
Project: Expanding Subworkflow Catalog of RECAST-wf For Event Generators
Funding Period: 6 months part time

RECAST[1] is a framework for reinterpreting LHC analyses using Yadage computational workflows. These workflows can be run on the researcher's own computer or through the cloud application REANA[2].

RECAST-workflow[3] builds on RECAST in order to run truth-level reinterpretations which achieve much faster results by sacrificing complexity. It also allows for workflows to be modularized through subworkflows which encapsulate each step (generation, selection, analysis). RECAST-workflow maps every input and output file formats to generate valid combinations of subworkflows that can be run as a complete workflow. These combinations can be filtered by common inputs and automatically generated, allowing researchers to customize each step of the process to best fit their needs.

This new system has the ability to quickly build novel truth-level reinterpretations allowing researchers to scan regions of phase space that would be interesting for a full reinterpretation that is much more computationally expensive and difficult to use. A command line interface for RECAST-workflow has previously been developed to provide a simple user-interface for creating and executing new workflows. Currently RECAST-workflow's truth reinterpretation workflows can only run on the user's local machine using RECAST-cli. The goal of this project is to improve the command line usability and documentation, improve MadGraph integration to support custom models, and add the additional event generators Sherpa[4] and Herwig[5]. If time permits another goal would be to start work on running the workflows in REANA on the cloud through building a simple web interface for RECAST.

Under the supervision at the University of Washington by Professor Shih-Chieh Hsu and PhD student Alex Schuy, Ed van Bruggen will build on the work of Vladimir Ovechkin[6] on statistical analysis by expanding the usability of the command line interface and improving the event generation subworkflows to support Sherpa and Herwig. He will then add to the existing documentation and tutorials to cover the entire project. This project will allow Ed to expand his knowledge on conducting collider analyses and working with large Python projects. Ed's work will be made available within the public software repositories of RECAST-workflow[7].

Schedule

- **Month 1**: Learn the details of RECAST-workflow through reading documentation and source code, completing tutorials
    - Integrate tutorials into existing documentation
    - Reproduce plots for example Dark Matter model
- **Month 2**: Improve interface of RECAST-workflow and integration of MadGraph
    - Provide more command line options and autofill input configurations
    - Allow custom MadGraph models to be used without needing to build a new Docker image.
- **Month 3-4**: Learn Sherpa event generation and integrate it into RECAST
- **Month 4-5**: Learn Herwig event generation and integrate it into RECAST
- **Month 6**: Build on documentation and unit tests to cover all code and use cases, write tutorials. Prepare for final presentation

References

[1] K. Cranmer, I. Yavin, “RECAST: Extending the Impact of Existing Analyses”, JHEP1104:038, 2011
[2] REANA http://reanahub.io/
[3] Schuy, Alex, “Extending RECAST for Truth-Level Reinterpretations”, arXiv:1910.10289 2019.
[4] Sherpa: https://sherpa.hepforge.org/trac/wiki
[5] Herwig: https://herwig.hepforge.org/
[6] Vladimir Ovechkin: https://iris-hep.org/fellows/vovechkin.html
[7] RECAST-workflow: https://github.com/recast-hep/recast-workflow