SAND2015-1516C

VISUALIZATION FOR HYPER-HEURISTICS

Students: Luke Simon, Lauren Kroenung
Natural Computation Laboratory
Department of Computer Science

Mentor: Dr. Samuel Mulder
Sandia National Laboratories

Faculty Advisor: Dr. Daniel Tauritz
Natural Computation Laboratory
Department of Computer Science

Motivation

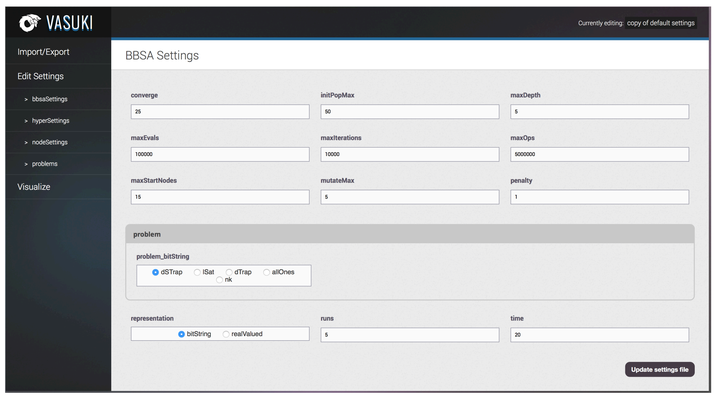
Many complex problems can be formulated as generate-and-test optimization problems. As general-purpose optimization algorithms do not target specific problems, they are not well suited for real-world scenarios where many instances of the same problem need to be repeatedly and efficiently solved.

Hyper-heuristics automate the design of algorithms to create a custom algorithm for a particular scenario. Automated designs can often be hard to apply to real-world problems and it can be difficult to understand exactly *how a design was derived and why it should be trusted*. **This project aims to address these issues of usability and understandability.**

Project Objective

- Develop an easy-to-use graphical user interface for hyper-heuristics to support practitioners and researchers
- Change run settings easily and then visualize key components of the framework output in a clear and understandable way
- Create digestible scientific visualization of the produced automated designs
- Interactive, in-depth analysis of the evolved algorithms and their experimental results

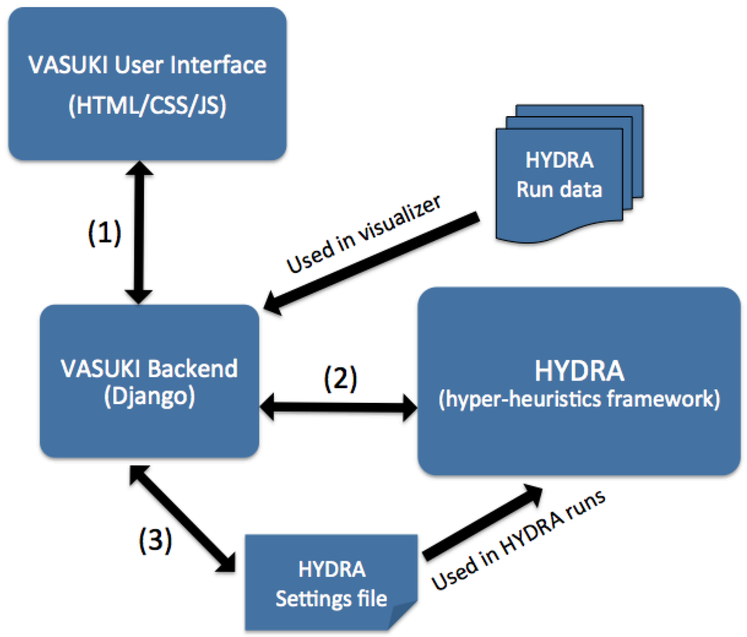
The system we have created is named **VASUKI** (***Visualization And Scientific User Kontrol Interface)***.



Users can edit the underlying framework's settings file visually, as shown here. This is one of the dynamically generated pages.

Approach

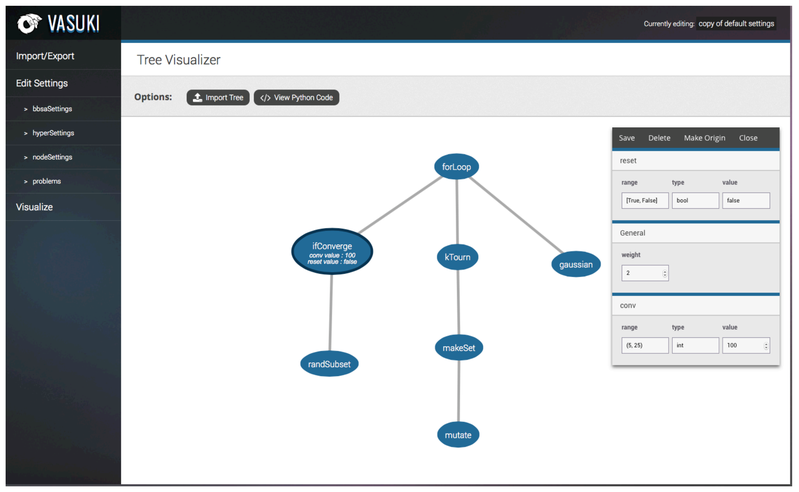
Web interface flow



1. Data transport between front and back end
2. Render components for visualizer from hyper-heuristics framework
3. Based on the settings provided by the current hyper-heuristic framework, dynamically generate pages that can directly modify the settings file

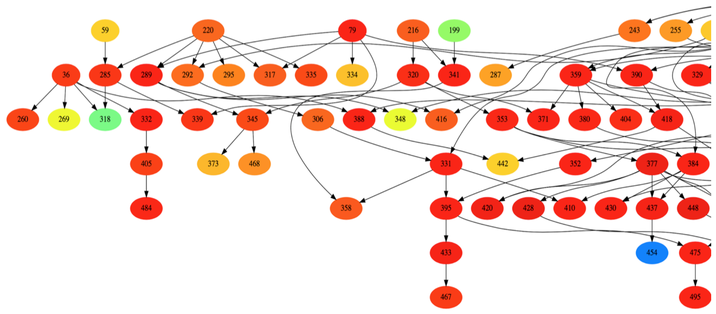
VASUKI Visualizer

The visualizer has two core functionalities. The first is a visual algorithm builder and editor.



Users are able to create tree-represented algorithms from building blocks respective to the hyper-heuristic infrastructure. These algorithms can then be exported for use in system runs.

The second functionality is utilized after executing a run on the framework. The run data can be displayed as a **phylogenetic tree** with which the user can interact for insightful analysis. An algorithm's ancestry can easily be discerned through various visualizer functions.



VASUKI's visualizer also allows users to display hyper-heuristic framework run data. We diagram this as a phylogenetic tree and provide tools that help in understanding how a specific design was derived.

Challenges

One challenge was designing VASUKI to work for a variety of different hyper-heuristic driven frameworks, to allow for future growth. Choosing to build pages dynamically means that when VASUKI needs to integrate with a different hyper-heuristic framework, the only action necessary is to feed it a different settings file for that framework.

Conclusion

Runs performed in hyper-heuristic frameworks produce complex data that can be difficult to interpret. The visualizations produced by VASUKI facilitate understanding evolutionary paths. **VASUKI's strength lies in its interactive creation of visualizations,** rather than just displaying static diagrams. The application allows users to interact with resulting data, independent of what framework the data originates from.

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

1. Data transport between front and backend

2. Render components for visualizer from hyper-heuristic framework

3. Based on the default settings provided by the current hyper-heuristic framework, VASUKI dynamically builds pages that can directly modify the settings file.

1) Data transport between front and backend

2) Render components for visualizer from hyper-heuristic framework

3) Based on the default settings provided by the current hyper-heuristic framework, VASUKI dynamically builds pages that can directly modify the settings file.

Users are able to create tree-represented algorithms from building blocks respective to the hyper-heuristic infrastructure. These visual algorithms can then be exported for use in the system runs.