About

Contents

Introduction

AMIDOL is designed to support models in a number of scientific, physical, social, and hybrid domains by allowing domain experts to construct meta-models in a novel way, using visual domain specific ontological languages (VDSOLs). These VDSOLs utilize an underlying intermediate abstract representation to give formal meaning to the intuitive process diagrams scientists and domain experts normally create. AMIDOL then provides translations from these VDSOLs into an intermediate representation which can be transformed as appropriate to compose models, apply optimizations, and translate them into executable representations allowing AMIDOL's inference engine to execute prognostic queries on reward models and communicate results to domain experts. In Phase 2 AMIDOL will bind these results to the original ontologies providing more explainability when compared to conventional methods.

AMIDOL addresses the problem of machine-assisted inference with two high-level goals:

1. improving the ability of domain experts to build and maintain models and
2. improving the explainability and agility of the results of machine-inference.

The VDSOLs help achieve these goals by lowering the barrier for entry associated with formal modeling languages. VDSOLs allow the expression of rich mathematical concepts using visual diagrams for systems and processes. The AMIDOL intermediate representation backs these diagrams with mathematical meaning, and allows the creation of new VDSOL elements as necessary with full expressivity, and Turing completeness. The use of the intermediate representation also allows AMIDOL to decouple the problem of model optimization, performance, and implementation from the task of model definition, ensuring any model defined in any VDSOL is translated to the same abstract representation. This abstraction can then be transformed using a set of well defined operations, eventually resulting in executable code which the AMIDOL inference engine can execute. Because AMIDOL uses a universal representation for this intermediate form, models defined with AMIDOL can be easily ported to off the shelf solution techniques which have already been vetted by the community, and optimized for high performance computing.

Documentation

- AMIDOL IR
- W3 Meeting Notes
- AMIDOL Brainstorming
- AMIDOL Ontology

Examples

- SIR Family

Resources

- Readling List

Building

You'll need to:

- install the `sbt` build tool and a recent version of the Oracle JDK for building the `ir`
- install the `python` interpreter and `pip` package manager, then the `pysces` and `numpy` packages (by doing `pip install pysces numpy`) for the PySCeS backend
- install the `python3` interpreter and `pip3` package manager, then the `scipy` and `matplotlib` packages (by doing `pip3 install scipy matplotlib`) for the SciPy backends

Once you have done all of this, build and run the system with:

```
$ git clone https://github.com/GaloisInc/AMIDOL.git && cd AMIDOL
AMIDOL$ sbt run
```

This opens a back-end web server on http://localhost:8080/ . NOTE: This system was only meant for use/tested on Google Chrome. This version of the system does not support other browsers.

Example models can be found in this repository under the `examples` directory. These are JSON files meant to be loaded into the web browser UI, with the cloud-shaped upload button in the upper-right. User-drawn models can also be downloaded with the adjacent download button.