

# ArgTools: a Labelling-based Solver for Abstract Argumentation

Samer Nofal<sup>1</sup>, Katie Atkinson<sup>2</sup>, and Paul E. Dunne<sup>2</sup>

<sup>1</sup> Department of Computer Science, German Jordanian University, Jordan

<sup>2</sup> Department of Computer Science, University of Liverpool, UK

**Abstract.** We present the second version of ArgTools for solving problems under a number of argumentation semantics. In addition to preferred, stable, complete, grounded semantics ArgTools is extended to handle argumentation problems under stage, semi-stable, and ideal semantics. ArgTools may find one/all solution(s) and may decide if there is a solution containing a given argument (the credulous acceptance problem) or decide whether all solutions contain a given argument or not (the sceptical acceptance problem.) For an introduction to argumentation semantics we refer to [2].

**Keywords:** computational argumentation, backtracking algorithms, argumentation semantics, Dung frameworks.

## 1 The Notion of ArgTools

ArgTools is based on a backtracking procedure that traverses an abstract binary tree in a depth-first manner. Starting from the root node, where all arguments are not labelled yet, the procedure moves to a left (respectively right) child by a left (respectively right) transition. The left transition involves labelling a selected argument with *in*, trying to build a solution including the newly *in* argument. The right transition requires labelling the selected argument with *undecided*, trying to build a solution without the argument. Every time an argument is labelled with *in* the attacked arguments are labelled *out* while the attacking arguments are labelled *must-out*. The search continues in the same way until no argument is left unlabelled. At this point a solution is found if there is no *must-out* argument. Then the process might backtrack to find a(nother) solution. For a full description of the algorithms of ArgTools we refer the reader to [1].

## 2 The Usage of ArgTools

ArgTools can be found at <https://sourceforge.net/projects/argtools>. ArgTools is implemented in C++. For compiling the source code you may use g++ (with the option `-std=c++11`.) The usage of ArgTools adheres to the requirements of ICCMA 2015 [3], which are available at <http://argumentationcompetition.org>.

### 3 A Remark on ArgTools

The new version of ArgTools is optimised by further enhancements. For the lack of space we aim to illustrate these new enhancements in future in an extended article.

#### References

1. Samer Nofal, Katie Atkinson, Paul E. Dunne: Looking-ahead in backtracking algorithms for abstract argumentation. *Int. J. Approx. Reasoning* 78: 265-282 (2016).
2. Pietro Baroni, Martin Caminada, Massimiliano Giacomin: An introduction to argumentation semantics. *Knowledge Eng. Review* 26(4): 365-410 (2011).
3. Matthias Thimm, Serena Villata, Federico Cerutti, Nir Oren, Hannes Strass, Mauro Vallati: Summary Report of The First International Competition on Computational Models of Argumentation. *AI Magazine* 37(1): 102 (2016).