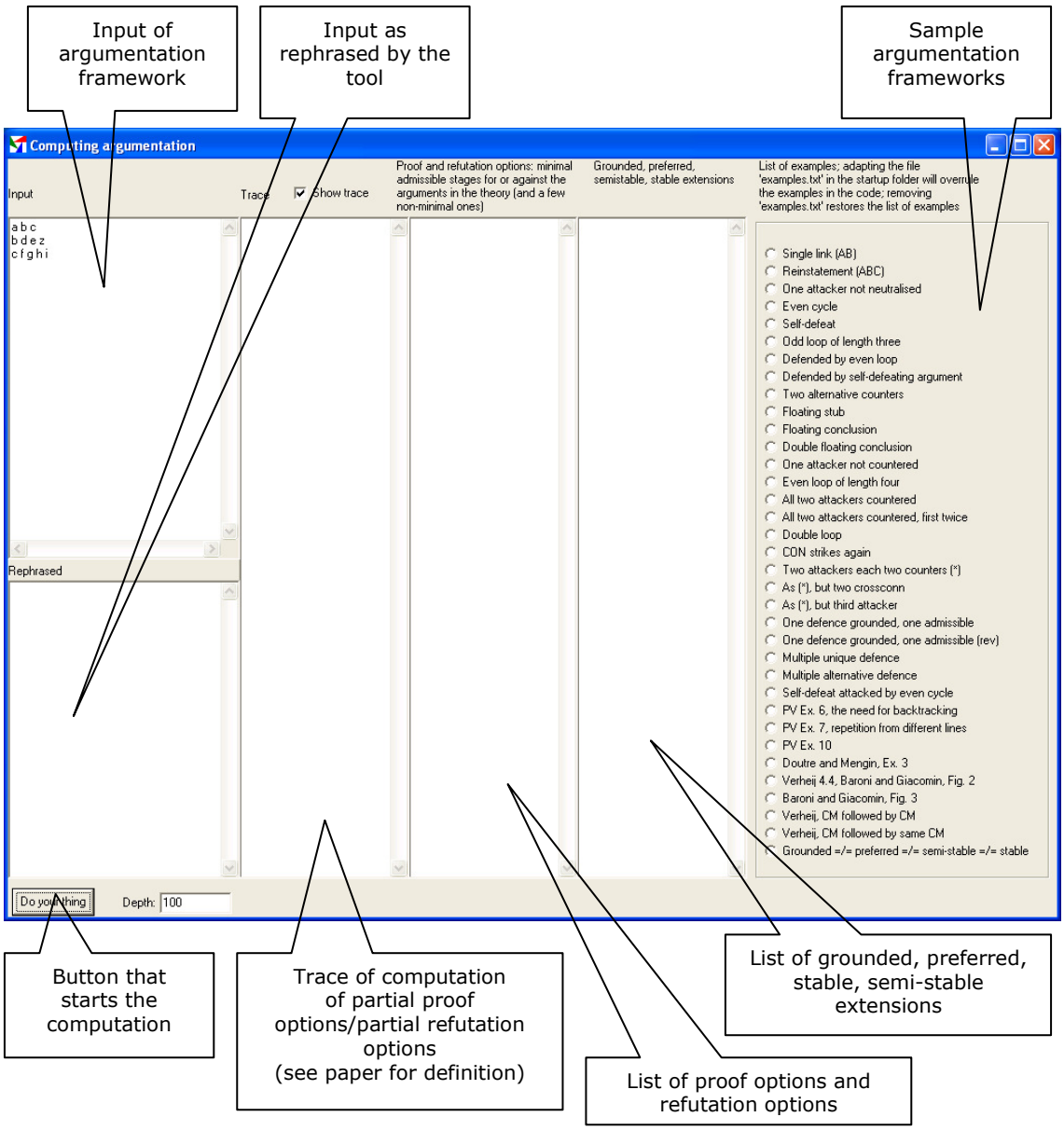
A TOOL FOR THE COMPUTATION OF ACCEPTANCE AND REJECTION FOR DUNG-STYLE ARGUMENTATION FRAMEWORKS

Bart Verheij



General

The tool computes the grounded, preferred, stable and semi-stable[1] extensions of Dung-style argumentation frameworks. Preferred extensions are computed by first constructing the so-called proof options and then gluing them. Proof options are closely related to minimal admissible sets containing an argument. Hence a proof option provides a kind of 'proof' that the argument can be defended against its attackers. Refutation options are a kind of 'refutation' of an argument, showing that an argument has an attacker that can be defended against its attackers. The algorithm and its formal underpinning are described in a paper by Verheij (2007, www.ai.rug.nl/~verheij/publications/ijcai2007.htm).

[1] A semi-stable extension is a preferred extension for which the union of the arguments in the set and those attacked by the set is $\subseteq$-maximal. Verheij 1996 has introduced the notion (using the term 'admissible stage extension'). The term 'semi-stable extension' was proposed by Caminada in 2006.

The tool's screen consists of five text areas. The top left one is used for input, all others are for output. On the right, a list of sample argumentation frameworks is given. Most examples in the list have been collected by Gerard Vreeswijk. Clicking one will use the sample framework as input for computation.

Input

Dung-style argumentation frameworks can be typed in the input field as a list of lists of arguments. Arguments are separated by spaces; lists of arguments by a comma or a new line. Lists of arguments are interpreted as follows: the first argument in a list is attacked by the other arguments in the list. For instance, using 'a b c, b d e' as input means that the argument 'a' is attacked by the arguments 'b' and 'c' and that 'b' is attacked by 'd' and 'e'. Lists of attackers are interpreted cumulatively. For instance, typing 'a b, a c' is equivalent to typing 'a b c'. Both mean that 'a' is attacked by 'b' and by 'c'.

The paper's format matches that of the tool: examples can be copied from the paper to the tool.

The tool rephrases the input in several ways that can sometimes be helpful. The standard style lists arguments with all their attackers, each on a separate line. The comma style lists arguments with all their attackers, separated by commas.

Starting the computation

Computation starts by clicking the button labeled "Do your thing".

Output

The three text areas to the right of the input area provide the output of the computation. The first one shows the trace of the computational process, by listing, for each argument, all partial proof and refutation options (see the definitions in the paper). The second text area gives, for each argument, the proof and refutation options. These include all minimal admissible sets containing or attacking the argument (and for many situations coincide with them). The third, rightmost text area provides the unique grounded extension, all preferred extensions, all stable extensions and all semi-stable extensions) of the argumentation framework. The preferred extensions are computed by 'gluing' proof and refutation options (cf. the paper), the stable and semi-stable extensions by making a selection among the preferred extensions.

Output uses a labeling format, listing arguments that are labeled as justified (IN, +) and that are labeled as defeated (OUT, -). For instance, 'a (b c)' expresses that the argument 'a' is IN, while the arguments 'b' and 'c' are OUT.

References

Verheij, B. (1996). Two Approaches to Dialectical Argumentation: Admissible Sets and Argumentation Stages. *Naic'96. Proceedings of the Eighth Dutch Conference on Artificial Intelligence* (eds. J.-J.C. Meyer & L.C. van der Gaag), pp. 357-368. Utrecht: Universiteit Utrecht. A preliminary version was presented at the Computational Dialectics Workshop at FAPR-96. June 3-7, 1996, Bonn.

Verheij, B. (2007). A Labeling Approach to the Computation of Credulous Acceptance in Argumentation. *Ijcai 2007, Proceedings of the 20th International Joint Conference on Artificial Intelligence, Hyderabad, India, January 6-12, 2007* (ed. M.M. Veloso), pp. 623-628. http://www.ijcai.org/proceedings07.php