# `flexABle`– System Description for ICCMA 2023


Martin Diller
*Logic Programming and Argumentation Group*
*TU Dresden, Germany*
martin.diller@tu-dresden.de

Sarah Alice Gaggl
*Logic Programming and Argumentation Group*
*TU Dresden, Germany*
sarah.gaggl@tu-dresden.de

Piotr Gorczyca
*Computational Logic Group*
*TU Dresden, Germany*
piotr.gorczyca@tu-dresden.de



***Abstract*—We describe `flexABle`, a system for (interactively & automatically) computing dialectical justifications of claims in assumption-based argumentation or ABA. The system participates in the ICCMA 2023 ABA tracks for determining credulous acceptance of claims for the complete and stable semantics.**

***Index Terms*—argumentation competition, assumption-based argumentation, dispute derivations, explanation**


## I. Introduction

Dispute derivations [1]–[3] are one of the main native reasoning methods for assumption-based argumentation or ABA [4][1]. Based on games for abstract argumentation [5], they are conceived of as a dispute with arguments for and against some claims under scrutiny being exchanged by a proponent and an opponent of the claims. The arguments themselves are proof trees: they consist of a claim (the conclusion) derived from assumptions (uncertain premisses) and facts (certain premisses) via chaining of rules (modus ponens). Disputes, more than just a method for reasoning about acceptance of claims, provide a means of dialectical explication [6].

The system `flexABle` implements flexible dispute derivations for ABA [3]; these build on previous forms of dispute derivations [1], [2], while also differing in the following substantial aspects:

- In flexible dispute derivations the proponent and opponent are completely omniscient in that they remember all arguments (and sub-arguments) put forward during a dispute; this minimizes redundancy in the sense that repetition of arguments is avoided.
- Related to the above, flexible disputes are defined in both an argumentation-based as well as a rule-based version. In the first, at each move, the proponent and opponent start the construction of new arguments to the dispute (by putting forward assumptions or facts), or expand arguments which already form part of an ongoing dispute (by making use of a rule). In the second, implementation-oriented representation (in that arguments do not need to be explicitly constructed or stored), dispute states consist simply in the set of claims and rules which have been put forward up until that state and a move consists in adding a new claim or rule to this set. To the extent that the rule-based representation includes also information about the relationships between claims and rules, it can also be thought of as operating on a graph. In this aspect flexible disputes take graph-based disputes [2] one step further: all arguments put forward in a dispute are represented as a graph and not only those of the proponent.
- In addition to backward moves (or top-down, from claims to premisses) as in previous versions of disputes, flexible disputes also allow forward (or bottom-up) moves. Specifically, new assumptions can be added to disputes and claims deduced from claims previously put forward. This often leads to shorter disputes for deciding acceptance of claims wrt the admissible (and, hence, complete) semantics (notably in that the proponent can deduce further claims from the assumptions it has committed to and, thus, preemptively block further lines of attack from the opponent). Moreover, via forward moves, variants of disputes can be defined not only for deciding acceptance, but also finding complete and stable assumption sets congruous with the goal claims.
- Finally, flexible dispute derivations are also more flexible in the moves allowed at each step. In particular, if Dung's abstract argumentation frameworks are encoded as ABA frameworks, then via flexible disputes one obtains games for deciding acceptance of arguments that are not dispute-tree-based [5].

## II. An example

Consider the ABA framework $\mathcal{F} = (\mathcal{L}, \mathcal{A}, ^{-}, \mathcal{R})$, with assumptions $\mathcal{A} = \{a, b, c, d, e, f, g, h\}$, contrary relation $^{-}(\alpha) = \bar{\alpha}$ for $\alpha \in \mathcal{A}$ and rules $\mathcal{R} = \{s \leftarrow a, f, \bar{g}; \bar{g} \leftarrow b, u; u \leftarrow d; \bar{f} \leftarrow g, w; w \leftarrow e; \bar{a} \leftarrow c, \bar{g}; \bar{c} \leftarrow h, x; \bar{c} \leftarrow u; \bar{h} \leftarrow y, z\}$. The underlying language is $\mathcal{L} = \mathcal{A} \cup \bigcup_{h \leftarrow B \in \mathcal{R}} \{h\} \cup B$.

The argument-based representation of a flexible dispute that shows credulous acceptance of $s$ for the complete semantics is shown in Figure 1. To obtain this dispute the proponent first performs three (conservative) backward moves from the goal $s$ using rules 1) $s \leftarrow a, f, \bar{g}$, 2) $\bar{g} \leftarrow b, u$, and 3) $u \leftarrow d$. Thus the proponent has a complete argument for $s$: argument $A_1$ in the figure; i.e. one in which the premisses are all assumptions or facts (the piecewise construction of $A_1$ being shown via the boxes with dashed lines labelled 0, 1, 2, 3). Now the opponent moves (non-conservatively) backward one step only by making


The authors are funded by the German Federal Ministry of Education and Research (BMBF) via the Center for Scalable Data Analytics and Artificial Intelligence (ScaDS.AI), grant 01IS20056_NAVAS, and project 13GW0552B (KIMEDS, KI-assistierte Zertifizierung medizinischer Software) respectively.


[1] More concretely, flat propositional Horn-ABA.

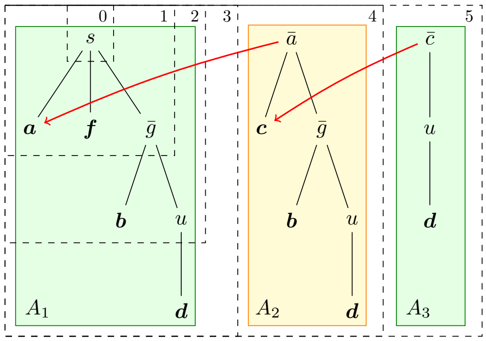


Fig. 1: A dispute in five steps justifying acceptance of the claim $s$ for the complete semantics.

use of the rule 4) $\bar{a} \leftarrow c, \bar{g}$ attacking assumption $a$ of argument $A_1$. Immediately the complete argument $A_2$ attacking $A_1$ in the figure becomes part of the dispute, since a sub-argument supporting $\bar{g}$ has already been put forward (a sub-argument of $A_1$). There are no further arguments attacking $A_1$ that can be put forward by the opponent, since the other rule supporting a contrary of an assumption appearing in $A_1$ ($\bar{f} \leftarrow g, w$) is blocked in virtue of the proponent already having an argument questioning assumption $g$ (the sub-argument of $A_1$ with conclusion $\bar{g}$). Now the proponent could try to attack $A_2$ by exploring argument lines against the assumption $c$ making use of either rules $\bar{c} \leftarrow h, x$ or $\bar{c} \leftarrow u$, but there is no need to guess if rather the proponent makes one (conservative) forward move deriving $\bar{c}$ from the sub-argument of $A_1$ concluding $u$ by making use of the rule 5) $\bar{c} \leftarrow u$ (argument $A_3$ in the figure). In fact, if the proponent would have followed a more eager strategy and constructed $A_3$ right after $A_1$ then the dispute would have ended there (in four steps rather than five), since this move would have preemptively blocked the rule $\bar{a} \leftarrow c, \bar{g}$ and thus the construction of the opponent's argument $A_2$.

## III. System description, reasoning tasks

The most obvious benefits of dispute derivations are for interactively exploring grounds for accepting claims in argumentative fashion. The system `flexABle`, implemented in the programming language `Scala`[2], succeeds its predecessor `aba-dd-rule-based` (presented in [3]) in also having an automatic mode. Moreover, it includes several other features not relevant to the ICCMA 2023 competition, as different interactive modes, combining interactive and automatic modes, supporting approximate reasoning, as well as generating graphical outputs of disputes both in their rule as well as argument-based representation (the rule-based one being used internally by `flexABle`). For further details we refer to [7], the github page[3] for `flexABle`, as well as a youtube overview[4].

As to the automatic mode of `flexABle`, which is what is evaluated at ICCMA 2023, this is a relatively naive implementation of a search for a dispute justifying acceptance of a set of claims. Underlying such a search is a strategy, which consists in 1) a preference order on move types (which move types to prioritize at each step), 2) several parameters for selecting among possible rules to be made use of in a move, and 3) the option of making use of breath-first or depth-first search. We refer to [7], [8] for details.

At ICCMA 2023 `flexABle` participates in the subtracks of the ABA track solving the credulous acceptance problem:

- DC-$\sigma$ with $\sigma \in \{CO, ST\}$ (complete and stable semantics): Given an ABA framework $\mathcal{F} = (\mathcal{L}, \mathcal{A}, ^{-}, \mathcal{R})$ and $s \in \mathcal{L}$, check whether $s$ is credulously accepted under $\sigma$.

For each of the semantics, we set the strategy to be made use of at ICCMA 2023 to be the best performing (total running time) according to our own initial experiments (also comparing to the system `abagraph` from [2]). This eager strategy prioritizes moves from the proponent, in particular conservative forward moves (moves deducing further claims from claims already committed to); it also uses depth-first search. For the complete semantics, non-conservative forward moves (guessing new assumptions to be put forward) are not used. For the stable semantics, the approach we submitted to ICCMA 2023 is that which first tries to find an admissible set of assumptions congruous with the goal claim without making use of non-conservative forward moves; only when such a set of assumptions has been found this set is extended by also making use of non-conservative forward moves to obtain a stable set (when possible). For further details we refer to [7], [8] where the dispute variant for the complete semantics is called DABF+TA and the strategy $S_2$+DFS, while for the stable semantics this is the approach that starts with the aforementioned combination of dispute variant and strategy for the complete semantics (rather than using the dispute variant for the stable semantics DS+TS from the start).

[2] https://www.scala-lang.org/
[3] https://github.com/gorczyca/aba-dd-rule-based
[4] https://www.youtube.com/watch?v=Q_28eSAjoqw

## References

[1] F. Toni, "A generalised framework for dispute derivations in assumption-based argumentation," *Artif. Intell.*, vol. 195, pp. 1–43, 2013.
[2] R. Craven and F. Toni, "Argument graphs and assumption-based argumentation," *Artif. Intell.*, vol. 233, pp. 1–59, 2016.
[3] M. Diller, S. A. Gaggl, and P. Gorczyca, "Flexible dispute derivations with forward and backward arguments for assumption-based argumentation," in *CLAR*, ser. LNCS, vol. 13040, 2021, pp. 147–168.
[4] K. Cyras, X. Fan, C. Schulz, and F. Toni, "Assumption-based argumentation: Disputes, explanations, preferences," in *Handbook of Formal Argumentation*, P. Baroni, D. Gabbay, and M. Giacomin, Eds., 2018, pp. 365–408.
[5] M. Caminada, "Argumentation semantics as formal discussion," in *Handbook of Formal Argumentation*, P. Baroni, D. Gabbay, and M. Giacomin, Eds., 2018, pp. 487–518.
[6] K. Cyras, A. Rago, E. Albini, P. Baroni, and F. Toni, "Argumentative XAI: A survey," in *IJCAI*, 2021, pp. 4392–4399.
[7] P. Gorczyca, "Automatic and interactive search in flexible dispute derivations for assumption-based argumentation: Analysis, implementation, evaluation." Master's thesis, TU Dresden, Dresden, 07 2022. [Online]. Available: https://iccl.inf.tu-dresden.de/w/images/e/e6/Gorczyca_MScThesis_Signed.pdf
[8] M. Diller, S. A. Gaggl, and P. Gorczyca, "Strategies in flexible dispute derivations for assumption-based argumentation," in *SAFA@COMMA*, ser. CEUR Workshop Proceedings, vol. 3236, 2022, pp. 59–72.