Proceedings of the 23rd International Conference on Principles of Knowledge Representation and Reasoning
Main Track

Splitting Argumentation Frameworks with Collective Attacks and Supports

Matti Berthold[1], **Lydia Blümel**[1], **Giovanni Buraglio**[2], **Anna Rapberger**[3]
[1]FernUniversität in Hagen, Germany
[2]TU Wien, Austria
[3]TU Dortmund, Germany
{lydia.bluemel, matti.berthold}@fernuni-hagen.de, giovanni.buraglio@tuwien.ac.at,
anna.rapberger@tu-dortmund.de

Abstract

This work proposes novel splitting techniques for argumentation formalisms that incorporate supports between defeasible elements. We base our studies on *bipolar set-based argumentation frameworks (BSAFs)* which generalize argumentation frameworks with collective attacks (SETAFs), as well as bipolar argumentation frameworks (BAFs), by incorporating both collective attacks and supports. Notably, BSAFs establish a crucial link to structured argumentation as they naturally capture general (potentially non-flat) assumption-based argumentation. The increase in expressiveness calls for diverse forms of splitting. We consider splits over collective attacks (thereby generalizing the recently proposed splitting techniques for SETAFs), splits over collective supports, as well as splits over both collective attacks and supports. We establish suitable splitting schemata and prove their correctness for the most common argumentation semantics.

1 Introduction

In the field of knowledge representation and reasoning, formal models of argumentation (Baroni et al. 2018) provide computational approaches for representing and reasoning about argumentative scenarios, and have been successfully applied across a wide range of domains such as medical decision-making, law and explainable AI (Atkinson et al. 2017; Cyras et al. 2021; Leofante et al. 2024). In particular, abstract models of argumentation have gained significant popularity due to their flexible and rich modelling capabilities. Starting from Dung's seminal work (Dung 1995), Argumentation Frameworks (AFs) have become a central formalism for capturing debates and reasoning over them, with the ultimate goal of extracting rational viewpoints. In Dung-style AFs, arguments are treated as primitive entities, while their internal structure, e.g. the premises supporting their conclusions, is abstracted away. Conflicts between arguments are captured by a binary attack relation. Debates are thus represented via simple directed graphs where nodes are the arguments exchanged and edges keep track of their conflicts. Rational viewpoints are then identified by computing sets of jointly acceptable arguments, so-called extensions.

Over the years, numerous generalizations of AFs have been proposed to enrich their simple structure and increase its modelling capabilities (Brewka et al. 2018; Amgoud et al. 2008; Nielsen and Parsons 2006; Modgil 2009). Notably, bipolar argumentation frameworks (BAFs) (Amgoud et al. 2008; Boella et al. 2010) allow also for the explicit representation of support relations, enabling the modelling of scenarios in which arguments may reinforce one another. In parallel, argumentation frameworks with collective attacks (also called SETAFs) (Nielsen and Parsons 2006) have been introduced to capture situations in which an argument is insufficient to attack another on its own, but can do so jointly with others. More recently, bipolar set-based argumentation frameworks (BSAFs) (Berthold, Rapberger, and Ulbricht 2024) have been proposed as a unifying generalization of both BAFs and SETAFs by combining the accrual of arguments with a notion of support. With this, BSAFs can model scenarios in which arguments 'join forces' to support or defeat another argument collectively. In addition, they have been shown to faithfully capture rule-based argumentation formalisms such as general (non-flat) assumption-based argumentation (Berthold, Rapberger, and Ulbricht 2024). As a result, BSAFs offer a simple yet expressive means of representation for complex argumentative scenarios.

However, similarly to AFs, BAFs, and SETAFs, bipolar set-based argumentation frameworks exhibit a high computational complexity for most standard reasoning tasks (Berthold, Rapberger, and Ulbricht 2024). In particular, the size of the search space may grow exponentially with the size of the framework, making direct computation of extensions infeasible in practice. To address this challenge, incremental reasoning techniques have been developed that decompose a framework into smaller sub-frameworks, compute their extensions independently, and then combine the results (Baumann 2011; Liao 2013; Baroni, Giacomin, and Liao 2014; Bengel and Thimm 2025; Blümel et al. 2025; Giacomin, Baroni, and Cerutti 2021). Among these techniques, splitting (Baumann 2011; Baumann et al. 2012; Linsbichler 2014; Buraglio et al. 2024; Buraglio 2025) has proven to be particularly effective in abstract argumentation (Baumann, Brewka, and Wong 2011). The central idea underlying splitting procedures is to compute the extensions of a given framework F by decomposing it into suitable sub-frameworks F_1 and F_2 and then computing the extensions of F_1 and F_2 independently. Naturally, this requires suitable modifications of F_2 to take into account the acceptance status of the arguments in F_1. Moreover, although computational efficiency is the primary motivation for employing splitting techniques, they also provide a useful tool for the study of argumentation dynamics (Rapberger and Ulbricht 2023; Prakken 2023; Cayrol, de Saint-Cyr, and Lagasquie-Schiex 2010), as previously computed parts of the framework can be reused when new information is introduced (Baumann 2011; Liao, Jin, and Koons 2011).

Splitting has been successfully applied to AFs and generalizations thereof; most crucial in our context, to SETAFs. Nevertheless, despite the increasing demand for efficient approaches that take into account positve links, the development of splitting techniques for argumentation formalisms that feature support relations, particularly in the presence of collective relations, remains largely unaddressed.

In this paper, we start to fill this gap by investigating the possibilities and limitations of splitting in AFs that feature both collective attacks and supports. The added expressiveness of BSAFs allows for a richer variety of cuts: a framework may be split along collective attacks, collective supports, or combinations thereof. We introduce the necessary ingredients for splitting BSAFs in a progressive manner, addressing the unique challenges that arise due to the presence of supports. We investigate splitting techniques for attacks and supports separately before we present a unified splitting pipeline that enables the incremental computation of extensions for arbitrary cuts of a BSAF.

Our main contributions are as follows.

- First, we develop a suitable attack splitting procedure for BSAFs by extending and revising the splitting procedure for SETAFs. We prove the correctness of our splitting schema wrt. the most common argumentation semantics. Notably, we show that splitting works only partially under grounded semantics. Section 3
- Next, we investigate the splitting of supports. We define suitable modifications that enable splitting wrt. the common argumentation semantics. Notably, for preferred and grounded semantics, the splitting procedure selects only some, and not all, extensions. Section 4
- Finally, we combine the previous techniques and consider arbitrary splits over collective attacks and supports. We prove a general splitting theorem, thereby paving the way for the incremental computation under most of the semantics. The combined splitting inherits the limitations concerning preferred and grounded semantics. Section 5

All proofs are available at https://arxiv.org/abs/2604.28112.

2 Background

We recall BSAFs (Berthold, Rapberger, and Ulbricht 2024) which generalize SETAFs (Nielsen and Parsons 2006), and basics of splitting for SETAFs (Buraglio et al. 2024).

2.1 Bipolar SETAFs

Bipolar argumentation frameworks with collective attacks (BSAFs) (Berthold, Rapberger, and Ulbricht 2024) are a generalization of Dung's abstract argumentation frameworks (AFs) (Dung 1995). They combine the ideas underlying AFs with collective attacks (SETAFs) (Nielsen and Parsons 2006) and bipolar AFs (BAFs) (Cayrol and Lagasquie-Schiex 2005; Amgoud et al. 2008; Ulbricht et al. 2024). In particular, they incorporate notions of support and attack among arguments, and generalize both SETAFs and BAFs by modeling *collective* attacks and supports.

Definition 1. *A bipolar set-argumentation framework (BSAF) is a tuple $F = (A, R, S)$, where A is a finite set of arguments, $R \subseteq 2^A \times A$ is the attack relation and $S \subseteq 2^A \times A$ is the support relation.*

A *SETAF* is a BSAF $F = (A, R, S)$ with $S = \emptyset$; an *AF* is a SETAF with $|T| = 1$ for all $(T, h) \in R$.

For an attack (support) (T, h), we call T the *tail* of the attack (support) and h the *head* of the attack (support). We call (T, h) a *negative link* if $(T, h) \in R$, a *positive link* if $(T, h) \in S$, and simply a *link* if $(T, h) \in R \cup S$. If $|T| = 1$, we write (t, h) instead of $(\{t\}, h)$, where $T = \{t\}$. Furthermore, given BSAFs $F = (A, R, S)$ and $F' = (A', R', S')$, we define set-operations component-wise; e.g., the union is defined as $F \cup F' = (A \cup A', R \cup R', S \cup S')$.

In contrast to the broad agreement on the interpretation of attacks, there exist several interpretations of support in the literature (Cohen et al. 2014; Cohen et al. 2018; Polberg 2016). BSAFs adopt a form of *deductive support* (Boella et al. 2010; Berthold, Rapberger, and Ulbricht 2024). A central notion in this context is the *closure* of a set.

Definition 2. *Given BSAF $F = (A, R, S)$ and $E \subseteq A$, let*

$$supp_F(E) := E \cup \{h \in A \mid \exists (T, h) \in S : T \subseteq E\}.$$

The closure *of E is defined as $cl_F(E) := \bigcup_{i \geq 1} supp^i_F(E)$; Hence, E is* closed *if $cl_F(E) = E$.*

Definition 3. *Given BSAF $F = (A, R, S)$, a set $E \subseteq A$* defends *$a \in A$ if for each closed attacker $E' \subseteq A$ of a, E attacks E'; E* defends *E' if E defends each $a \in E'$.*

We omit the subscript F for cl if clear from context.

Let us now head to BSAF semantics. A set E is conflict-free ($E \in cf(F)$) if it does not attack itself; E is admissible ($E \in adm(F)$) if it is conflict-free, closed and defends itself.

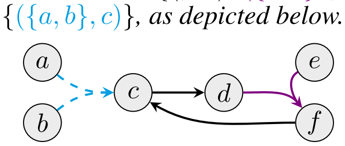
Definition 4. *Let F be a BSAF and let $E \in adm(F)$.*

- *$E \in com(F)$ iff E contains every argument it defends;*
- *$E \in grd(F)$ iff E is $\subseteq$-minimal in $com(F)$;*
- *$E \in pref(F)$ iff E is $\subseteq$-maximal in $adm(F)$.*
- *$E \in stb(F)$ iff E attacks each $x \in A \setminus E$.*

Given a BSAF $F = (A, R, S)$ and a set of arguments $E \subseteq A$, we denote $E^+_R := \{h \mid \exists T \subseteq E : (T, h) \in R\}$ and the range of E; by $E^\oplus_R := E_R \cup E^+_R$. We omit R if clear from the context. Given a σ-extension E of F, we say that an argument $a \in A$ is: *accepted* or *in* if $a \in E$, *rejected*, *defeated* or *out* if $a \in E^+_R$ and *undecided* if $a \notin E^\oplus_R$.

Graphically, we depict the attack and the support relations of a BSAF via solid and dashed edges respectively.

Example 5. *We consider a BSAF F with arguments $A = \{a, b, c, d, e\}$, attacks $R = \{(f, c), (\{d, e\}, f), (c, d)\}$, and support $S = \{(\{a, b\}, c)\}$, as depicted below.*



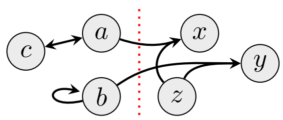
The arguments a and b jointly support c; thus, whenever $\{a, b\}$ is accepted, the argument c must be true as well. Accepting c is, however, impossible: to defend c against f, both arguments d and e are required; only then, the joint attack $(\{d, e\}, f)$ fires. However, accepting both is not possible since c attacks d. Thus, a and b cannot be accepted together. It follows that F has no complete and grounded extension; the admissible sets are $\emptyset$, $\{a\}$, $\{b\}$, $\{e\}$, $\{a, e\}$, and $\{b, e\}$.

2.2 Splitting SETAFs

In their recent work, Buraglio et al. (2024) investigated splitting procedures for SETAFs, extending splitting procedures for AFs (Baumann 2011). A splitting of a SETAF $SF = (A, R)$ separates the framework into two frameworks SF_1 and SF_2 which share attacks directed towards SF_2.

Definition 6. *Let $SF = (A, R)$ be a SETAF, $SF_1 = (A_1, R_1)$ and $SF_2 = (A_2, R_2)$ two sub-frameworks of SF such that $A_1 \cap A_2 = \emptyset$, $A = A_1 \cup A_2$ and $R = R_1 \cup R_2 \cup R_3$ with $R_3 \subseteq \{(T, h) \in R \mid T \cap A_1 \neq \emptyset, T \subseteq A, h \in A_2\}$. The triple (SF_1, SF_2, R_3) is called a* splitting *of SF and R_3 its set of* negative links. *A link is* undecided *if no argument in its tail is defeated, but at least one is undecided.*

Example 7. *We consider a splitting (SF_1, SF_2, R_3) of a SETAF SF, as depicted below; SF_1 and SF_2 are left resp. right of the dotted line; the set of shared negative links of SF_1 and SF_2 is given by $R_3 = \{(\{a, z\}, x), (\{b, z\}, y)\}$.*

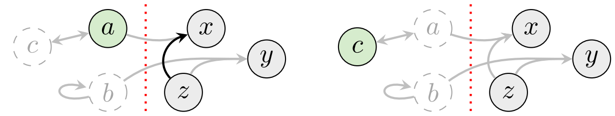


Buraglio et al. (2024) developed a procedure that enables the incremental computation of the extensions of SF wrt. a given semantics σ. The extensions of SF can be computed as a combination of extensions of SF_1 and (an adjusted version of) SF_2. The second sub-framework SF_2 gets modified on the basis of the information contained in the extension(s) of SF_1. Such alteration is tailored to account for the prior accepted and rejected, and undecided arguments. In the literature (Baumann 2011; Buraglio et al. 2024), this is achieved in a two-step procedure, by appealing to the notions of so-called *reduct* and *modification* of the sub-framework SF_2. In the first step, the reduct takes care of the arguments in SF_2 that are rejected wrt. E_1 by deleting them, and modifies the links by projecting the part of the attack to SF_2.

Definition 8 (Reduct)**.** *Let (SF_1, SF_2, R_3) be a splitting for a SETAF SF. We define the (E_1, R_3)*-reduct *(or simply* reduct*) of SF_2 for some extension E_1 of SF_1 as the SETAF $SF_2^{E_1} = (A_2^{E_1}, R_2^{E_1})$ where,*

$$
\begin{aligned}
A_2^{E_1} =& \{a \in A_2 \mid a \notin (E_1)^+_{R_3}\} \text{ and} \\
R_2^{E_1} =& \{(T, h) \in R_2 \mid T \subseteq A_2^{E_1}, h \in A_2^{E_1}\} \cup \\
& \{(T \cap A_2, h) \mid (T, h) \in R_3,\ T \cap A_1 \subseteq E_1, \\
& \qquad T \cap (E_1)^+_{R_3} = \emptyset,\ T \setminus A_1 \neq \emptyset,\ h \in A_2^{E_1}\}
\end{aligned}
$$

Example 9. *We continue Example 7. Below, we depict the reduct wrt. $E_1 = \{a\}$ on the left-hand side, the reduct wrt. $E_1' = \{c\}$ on the right-hand side; the attacks and arguments not contained in the reducts are grayed out.*



We begin by computing the reduct wrt. E_1. Since $R_2 = \emptyset$, thus we only deal with the links in R_3. For the attack $(\{a, z\}, x)$, we have $\{a, z\} \cap A_1 = \{a\} \subseteq E_1$, $\{a, z\} \setminus A_1 \neq \emptyset$, and $\{a, z\}$ is not attacked by E_1, thus the attack (z, x) is in the reduct. The attack $(\{b, z\}, y)$ does not satisfy these conditions, thus it is not part of the reduct. The latter also holds for the reduct for E_1'; in addition, the attack $(\{a, z\}, x)$ is defeated since a is defeated.

As per the second step, the notion of *modification* is introduced to account for possibly undecided arguments wrt. E_1. As these may affect other arguments in SF_2 via the links, a notion of undecided links is introduce to keep track of such external influences on arguments in (the reduct of) SF_2.

Definition 10 (Undecided Links)**.** *Given a splitting (SF_1, SF_2, R_3) for a SETAF SF and a set $E_1 \subseteq A_1$ we define the* set of undecided links *wrt. E_1 as:*

$$U_{R_3}^{E_1} := \{(T, h) \in R_3 \mid T \cap (E_1)^+_R = \emptyset, (T \cap A_1) \not\subseteq E_1\}.$$

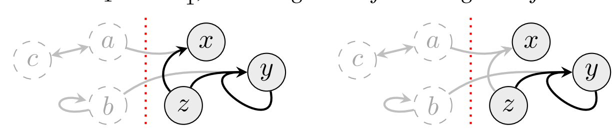
It holds that $(E_1)^+_R = (E_1)^+_{R_1 \cup R_3}$ since A_1 is not attacked by links in R_2. Next, we recall the *modification* which accounts for the effects of the undecided links.

Definition 11 (Modification)**.** *Let (SF_1, SF_2, R_3) be a splitting for SETAF SF and $E_1 \subseteq A_1$. Let $SF_2^{E_1} = (A_2^{E_1}, R_2^{E_1})$ be the (E_1, R_3)-reduct of SF_2 and $U_{R_3}^{E_1}$ the set of undecided links wrt. E_1. By $mod_{R_3}^{E_1}(SF_2^{E_1}) := SF_2^\star = (A_2^\star, R_2^\star)$ we denote the $U_{R_3}^{E_1}$*-modification *(or simply* modification*) of $SF_2^{E_1}$ s.t. $A_2^\star := A_2^{E_1}$ and $R_2^\star$ is given by:*

$$R_2^{E_1} \cup \{((T \cap A_2^{E_1}) \cup \{h\}, h) \mid (T, h) \in U_{R_3^c}^{E_1},\ h \in A_2^{E_1}\}.$$

Taken together, when splitting up a SETAF, we either separate the tail of collective attacks or remove them altogether.

Example 12. *In Example 9, the link $(\{b, z\}, y)$ is undecided for both E_1 and E_1'; resulting in the following modifications.*



Buraglio et al. (2024) proved the correctness of the procedure wrt. the semantics under consideration. They showed that (1) combining extensions of SF_1 and (the altered version of) SF_2 yields extensions of SF, and (2) extensions of SF induce extensions in SF_1 and (the altered version of) SF_2, when restricted to the respective argument-sets.

Theorem 13. *Let (SF_1, SF_2, R_3) be a splitting for a SETAF $SF = (A, R)$ with $SF_1 = (A_1, R_1)$, $SF_2 = (A_2, R_2)$, and $\sigma \in \{stb, adm, com, pref, grd\}$. Below, we let $SF_2^\star = mod_{R_3}^{E_1}(SF_2^{E_1})$.*

1. $E_1 \in \sigma(SF_1)$ *and* $E_2 \in \sigma(SF_2^\star)$ *implies* $E_1 \cup E_2 \in \sigma(SF)$.
2. *If* $E \in \sigma(SF)$, *then* $E_1 = E \cap A_1 \in \sigma(SF_1)$ *and* $E_2 = E \cap A_2 \in \sigma(SF_2^\star)$.

Example 14. *We use the result to compute the admissible extensions of* SF *from Example 7 incrementally. Both sets* E_1 *and* E_1' *are admissible in* SF_1. *For* E_1, $\{z\}$ *is the only (non-empty) admissible extension of* $SF_2^\star$*; therefore,* $\{a, z\} \in adm(SF)$. *For* E_1', *the sets* $\{x\}$ *and* $\{x, z\}$ *are admissible. Using the splitting theorem, we correctly deduce that* $\{c, x\}$, $\{c, z\}$ *and* $\{c, x, z\}$ *are admissible in* SF.

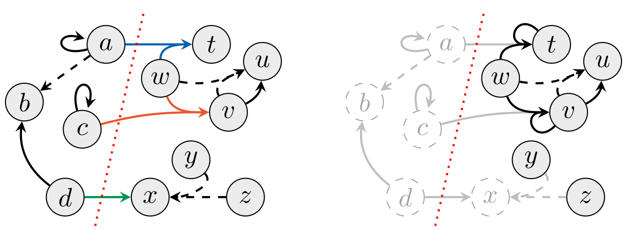
3 Splitting over Collective Attacks

In this section, we introduce a splitting algorithm for BSAFs. In particular, we take into account the situation where we can split over collective attacks only. Notably, the presence of support relations forces significant changes with respect to the SETAF notions of reduct and modification.

Definition 15 (Attack Splitting)**.** *Let* $F = (A, R, S)$ *be a BSAF,* $F_1 = (A_1, R_1, S_1)$ *and* $F_2 = (A_2, R_2, S_2)$ *two sub-frameworks of* F *such that* $A_1 \cap A_2 = \emptyset$, $A = A_1 \cup A_2$, $S = S_1 \cup S_2$, *and* $R = R_1 \cup R_2 \cup R_3$ *with* $R_3 \subseteq \{(T, h) \in R \mid T \cap A_1 \neq \emptyset, T \subseteq A, h \in A_2\}$. *We call the triple* (F_1, F_2, R_3) *an* attack splitting of F. *Moreover, we call* R_3 *the set of* negative links *wrt.* (F_1, F_2, R_3).

Ideally, we would like to transfer the results for SETAFs to the case of attack splitting for BSAFs. There are, however, certain subtleties that need to be taken care of. We illustrate these in the following example.

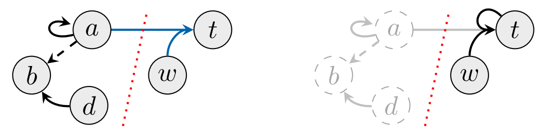
Example 16. *Consider the BSAF* F *below left and attack splitting* (F_1, F_2, R_3)*;* F_1 *is left and* F_2 *right of the dashed line. Let* $\{d\} = E_1 \in adm(F_1)$. *We illustrate in the Figure below right how the notions of SETAF reduct and modification apply to the present example.*



First, we consider $(\{a, w\}, t)$. *By definition, it is an undecided link, therefore we apply the modification to create the attack* $(\{w, t\}, t)$. *However, this makes* $\{t\}$ *not admissible. Further, the same happens for the undecided link* $(\{c, w\}, v)$. *However, the presence of support* $(\{v, w\}, u)$ *makes* $\{v\}$ *admissible again by defeating its closed attacker* $\{v, w, u\}$. *Hence, although* $(\{c, w\}, v)$ *is truly an undecided link, the SETAF modification is not adequate. Finally, the link* (d, x) *triggers the elimination of* x *via the reduct, together with its support, leaving* $\{y, z\}$ *as an admissible extension of* $F_2^{E_1}$. *However,* $\{d, y, z\}$ *is not admissible in* F *since it is not even closed.*

In what follows, we adapt the splitting schema and show how to solve the aforementioned issues.

Example 17. *Let us have a closer look into a part of the BSAF from Example 16. Consider the following attack splitting* (G_1, G_2, U_3) *of the BSAF* $G \subseteq F$ *(displayed left).*



The set $E = \{d, t, w\}$ *is admissible in* G *since* d *defends* t *against the closed set* $\{a, b, w\}$. *Note that* $\{a, w\}$ *is not closed thus it is not necessary to defend* t *against* $(\{a, w\}, t)$. *We depict* $G_2^\star$ *wrt.* $E_1 = \{d\}$ *above on the right-hand side.*

Observe that $\{d\}$ *is admissible in* G_1, *but* $\{t\}$ *is not admissible in* $G_2^\star$. *The issue is that the attack on* t *is indirectly defeated since its closure is attacked by* d. *This information, however, is lost when computing the modification* $G_2^\star$.

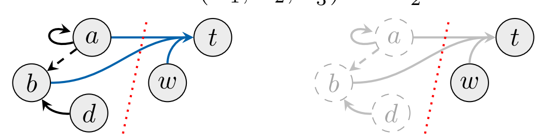
A negative link may be deemed as undecided under Definition 10, but become defeated due to some support in F_1, as the example shows. To account for this issue, we must deal with the negative links first in order to make explicit the information encoded in the support relation. Before we can construct the reduct, we alter F and apply the *link-closing* procedure, replacing all attacks $(T, a) \in R_3$ with $(cl(T), a)$.

Definition 18 (Closed Negative Links)**.** *Let* $F = (A, R, S)$ *be a BSAF,* $F_1 = (A_1, R_1, S_1)$ *and* $F_2 = (A_2, R_2, S_2)$ *two sub-frameworks of* F *such that* (F_1, F_2, R_3) *is an attack splitting of* F. *We define the set of* closed negative links *as*

$$R_3^c := \{(cl_S(T), h) \mid (T, h) \in R_3\}.$$

As a result, we are able to adapt the SETAF splitting schema while avoiding aforementioned issues concerning the interplay between negative links and supports in SF_1.

Example 19. *Let* (G_1, G_2, U_3) *be as in Example 17. We modify* $U_3 = \{(\{a, w\}, t)\}$ *as follows: first, we compute the closure of* $\{a, w\}$ *and replace the result with the source of the attack, obtaining* $U_3^c = \{(\{a, b, w\}, t)\}$ *(left). Then, the reduct wrt.* $E_1 = \{d\}$ *makes the attack defeated. Thus, the closed negative link* $(\{a, b, w\}, t)$ *is not undecided. We depict the link-closure* (G_1, G_2, U_3^c) *and* $F_2^\star$ *below.*



Now, $\{w, t\}$ *is admissible in* $G_2^\star$ *and we retrieve the admissible extension* $\{d, w, t\}$ *of* G.

Observe that closing the negative links preserves the semantics of the original framework.

Proposition 20. *Let* $F = (A, R, S)$ *be a BSAF,* $(T, h) \in R$ *an attack and let* $F' = (A, R', S)$ *with* $R' = (R \setminus \{(T, h)\}) \cup \{(cl(T), h)\}$. *Then* $\sigma(F) = \sigma(F')$ *for all* $\sigma \in \{stb, adm, com, pref, grd\}$.

After this preliminary step, we follow the splitting schema and appeal to the notions of reduct and modification. As observed in Example 16, the SETAF reduct cannot be directly inherited in our context. The reason is that the removal of arguments attacked by an extension of F_1 imposed by the reduct may interfere with the closed sets in F_2. If defeated arguments in F_2 are deleted, all their supporting sets would wrongly be considered closed in the reduct of F_2.

Example 21. *We consider another problematic part of the BSAF F from Example 16. Let $H \subseteq F$ with a splitting (H_1, H_2, V_3) as below left. Note that the link in V_3 is closed. The only preferred extension of H_1 is $E_1 = \{d\}$. By simply applying the original notion of (E, V_3)-reduct, we obtain the framework $H_2^{E_1}$ (right) which has one preferred extension $E_2 = \{y, z\}$, from which we would erroneously derive that $E_1 \cup E_2 = \{d, y, z\}$ is a preferred extension of H.*

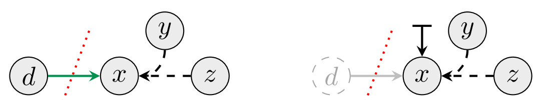
To account for this issue, we change the reduct as follows: instead of removing arguments defeated by E_1, we let them be attacked by the empty set. This syntactical difference preserves the defeated status of such arguments, while preserving its supports in F_2. In fact, for each defeated argument h which is supported by some set of arguments T in F_2, such an addition works as constraint: not all elements in T can be accepted, otherwise, h should be accepted as well.

Definition 22 (R-reduct)**.** *Let (F_1, F_2, R_3) be an attack splitting for a BSAF F; let R_3^c denote the closed negative links. We define the* (E_1, R_3^c)-reduct *(or simply* R-reduct*) of F_2 for some extension E_1 of F_1 as the BSAF $F_2^{E_1} := (A_2, R_2^{E_1}, S_2)$ with:*

$$R_2^{E_1} := R_2 \cup \{(T \cap A_2, h) \mid (T, h) \in R_3^c,\; T \cap (E_1)^+_{R_1 \cup R_3^c} = \emptyset,\; T \cap A_1 \subseteq E_1\}$$

Note that in the updated version of the reduct, no arguments are deleted. On top of that, negative links are projected to their (possibly empty) part in F_2.

Example 23. *Let (H_1, H_2, V_3) (left) be as in Example 21. The (E_1, V_3^c)-reduct consists of the framework $H_2^{E_1}$ (right) which has two preferred extensions $E_2 = \{y\}$ and $E_2' = \{z\}$, from which we now obtain the two preferred extensions of H, i.e. $E = \{a, y\}$ and $E' = \{a, z\}$.*



It remains to state the final modification to deal with the *closed* undecided links in R_3^c. For a given attack splitting (F_1, F_2, R_3) for a BSAF F and a set E_1 of arguments in F_1, we use the notion of undecided links as defined for SETAFs (see Definition 10, but applied to the BSAF splitting (F_1, F_2, R_3^c), i.e., after closing the links in R_3. We are now almost ready to apply the modification relative to $U_{R_3^c}^{E_1}$. There is, however, a final subtlety that we need to deal with.

Example 24. *Let us zoom into another part of the BSAF F from Example 16, as displayed below. We consider the attack splitting (I_1, I_2, W_3) for the BSAF $I \subseteq F$ (left), with $W_3^c = W_3 = \{(\{c, w\}, v)\}$, i.e., the attack is closed. Since $E_1 = \emptyset$ is an admissible extension of I_1, we know that the attack $(\{c, w\}, v)$ is an undecided link in W_3^c. Via the standard SETAF $U_{W_3^c}^{E_1}$-modification, we would add the set-self attack $(\{w, v\}, v)$ to obtain $I_2^\star$ (right).*

As previously noticed, the argument v now defend itself against its only closed *attacker $\{w, v, u\}$ by attacking u. Hence, we erroneously conclude that $\{v\} \in adm(I)$.*

A new version of the $U_{R_3^c}^{E_1}$-modification is then needed to deal with scenarios of this kind. Instead of using the target v of the attack to build a new self-attack in F_2, we introduce a self-attacking dummy argument $*_0$ that jointly attacks v together with the right-most part of the undecided link. Formally, the BSAF $U_{R_3^c}^{E_1}$-modification is defined as follows.

Definition 25 (Modification)**.** *Let (F_1, F_2, R_3) be an attack splitting for a BSAF F; let R_3^c denote the closed negative links. Let E_1 an extension of F_1. Take $F_2^{E_1}$ as the (E_1, R_3^c)-reduct of F_2 and $U_{R_3^c}^{E_1}$ as the set of undecided links wrt. E_1. By $mod_{R_3^c}^{E_1}(F_2^{E_1}) := F_2^\star = (A_2^\star, R_2^\star, S_2^\star)$ we denote the* $U_{R_3^c}^{E_1}$-modification *(or simply* modification*) of $F_2^{E_1} = (A_2, R_2^{E_1}, S_2)$ s.t. $A_2^\star := A_2 \cup \{*_0\}$, $S_2^\star := S_2$ and $R_2^\star$ is given by:*

$$R_2^{E_1} \cup \{(*_0, *_0)\} \cup \{((T \cap A_2) \cup \{*_0\}, h) \mid (T, h) \in U_{R_3}^{E_1}\}.$$

Example 26. *Let (I_1, I_2, W_3) as in Example 24 (left). The BSAF $U_{W_3^c}^{E_1}$-modification of I_2 is $I_2^\star$ (right); here, v does not defend itself against $\{w, *_0\}$ and is therefore not admissible.*

We are now in the position to state the overall splitting procedure. Given an attack splitting (F_1, F_2, R_3) for a BSAF F, we proceed as follows.

1. We close the negative links following Definition 18. We obtain a set of new negative closed links R_3^c.
2. We apply our BSAF reduct following Definition 22. As a result, we obtain two separate frameworks F_1 and $F_2^{E_1}$. They have the same arguments A_1 and A_2 as before; all arguments in F_2 that are defeated by attacks in F_1 are attacked by the empty set.
3. To account for undecided links, we apply the modification from Definition 25 and preserve undecidedness in F_2. The BSAF F_2 has been extended by the self-attacker $*_0$.

After these considerations, we are now in possession of all the ingredients to prove the correctness of our splitting schema for BSAFs. As a first step towards it, we ensure that the adjusted version of R-reduct is not in conflict with the notion of closed set of arguments.

Proposition 27. *Let (F_1, F_2, R_3) be an attack splitting for a BSAF $F = (A, R, S)$ with $F_1 = (A_1, R_1, S_1)$, $F_2 = (A_2, R_2, S_2)$; let R_3^c denote the closed negative links. Further, let $E_1 \subseteq A_1$, $F_2^{E_1}$ the (E_1, R_3^c)-reduct of F_2, $F_2^\star = mod_{R_3^c}^{E_1}(F_2^{E_1})$, and $E_2 \subseteq A_2^\star$.*

1. *If $E_1 = cl_{F_1}(E_1)$ and $E_2 = cl_{F_2^\star}(E_2)$, then $E = (E_1 \cup E_2) \setminus \{*_0\} = cl_F(E)$.*
2. *If $E = cl_F(E)$ for some $E \subseteq A$, then $E \cap A_1 = E_1 = cl_{F_1}(E_1)$ and $E \cap A_2 = E_2 = cl_{F_2^\star}(E_2)$.*

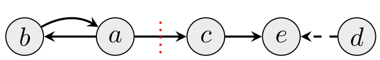
From this, we get an attack splitting theorem for BSAFs.

Theorem 28. *Let (F_1, F_2, R_3) be an attack splitting for a BSAF $F = (A, R, S)$ with $F_1 = (A_1, R_1, S_1)$, $F_2 = (A_2, R_2, S_2)$; let R_3^c denote the closed negative links. Let $\sigma \in \{stb, adm, com, pref\}$. Further, let $F_2^\star = mod_{R_3^c}^{E_1}(F_2^{E_1})$ where $F_2^{E_1}$ is the R-reduct wrt. E_1.*

1. *If $E_1 \in \sigma(F_1)$ and $E_2 \in \sigma(F_2^\star)$, then $E_1 \cup E_2 \in \sigma(F)$.*
2. *If $E \in \sigma(F)$, then $E_1 = E \cap A_1 \in \sigma(F_1)$ and $E \cap A_2 \in \sigma(F_2^\star)$.*

For grounded semantics, only the first direction of the splitting theorem can be guaranteed, due to a clash between the minimality and completeness conditions.

Example 29. *Consider the BSAF F with splitting (F_1, F_2, R_3) as below. The empty set defends d in F, so it is not complete. However, $\{d\}$ does not defend e, so that $E = \{a, d, e\}$ is the only grounded extension of F. Note, however, that the empty set is grounded in F_1, making it impossible to reconstruct E.*



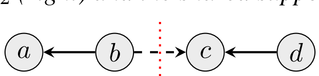
Theorem 30. *Let (F_1, F_2, R_3) be an attack splitting for a BSAF $F = (A, R, S)$ and R_3^c the set of closed negative links. Let $F_2^\star = mod_{R_3^c}^{E_1}(F_2^{E_1})$ where $F_2^{E_1}$ is the (E_1, R_3^c)-reduct. If $E_1 \in grd(F_1)$ and $E_2 \in grd(F_2^\star)$, then $E_1 \cup E_2 \in grd(F)$.*

4 Splitting over Collective Supports

We continue our investigation of how splitting translates from SETAFs to full BSAFs by focusing on the support relation. Analogously to attack splitting, we first consider support splitting in an *isolated setting*; i.e., we split a BSAF F with sub-BSAFs F_1 and F_2 which share *only positive links*.

In the case of attack splitting, our schema relies on the subsequent evaluation of F_1 and (a modification of) F_2 for a given BSAF F, unpacking the evaluation of the entire framework in an incremental fashion. Our goal is to find appropriate modifications that enable the modular computation of the semantics if F_1 and F_2 share positive links. However, a closer investigation of the effects of the supporting links reveals a fundamental issue.

Example 31. *Consider the BSAF F with the subframeworks F_1 (left) and F_2 (right) and the shared support (b, c) below.*



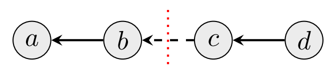
In F, the set $\{a, d\}$ is admissible since d defends a against the closed attacker $\{b, c\}$. In F_1, however, the argument a is defeated by b. The acceptability of a in F_1 cannot be accomplished without modifying F_1; this is, however, not foreseen in the traditional approach to splitting.

The underlying issue is that sets in F_2 are not closed; as we have demonstrated in the above example, the acceptance status of arguments in F_1 may thus depend on those of the arguments in F_2. Intuitively, the support in Example 31 has the effect of directing an attack from F_2 to F_1.

If we consider shared supports directed in the opposite direction, where the head h of each positive link (T, h) is contained in F_1, all sets in F_1 are closed. Therefore, this problem cannot occur in this case. We will therefore focus our efforts on *backward* support splitting.

Definition 32 (Support Splitting)**.** *Let $F = (A, R, S)$ be a BSAF, $F_1 = (A_1, R_1, S_1)$ and $F_2 = (A_2, R_2, S_2)$ two subframeworks of F such that $A_1 \cap A_2 = \emptyset$, $A = A_1 \cup A_2$, $R = R_1 \cup R_2$, and $S = S_1 \cup S_2 \cup S_3$ with $S_3 \subseteq \{(T, h) \in S \mid T \cap A_2 \neq \emptyset, T \subseteq A, h \in A_1\}$. We call the triple (F_1, F_2, S_3)* support splitting *of F. Moreover, we call S_3 the set of* positive links *wrt. (F_1, F_2, S_3).*

Example 33. *We consider a similar example as before; now, the support is directed in the other direction. The support splitting (F_1, F_2, S_3) for the BSAF F below consists of F_1 (left), F_2 (right) and a single shared support $S_3 = \{(c, b)\}$.*

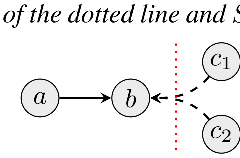


Note that d defends a against the closed attacker $\{b, c\}$ as before; however, in the given BSAF, the set $\{b\}$ is closed as well. Since b is not attacked, a cannot be defended.

Our goal is to develop a suitable splitting procedure analogous to the case for splitting attacks. We consider a σ-extension of F_1 and modify F_2 so that, for each σ-extension E_2 of (the modified version of) F_2, it holds that $E_1 \cup E_2$ is a σ-extension of F. Likewise, each σ-extension E of F should satisfy $E \cap A_1$ and $E \cap A_2$ both are σ-extensions of F_1 and (the modified version of) F_2, respectively.

Given the BSAF from Example 33 and the set $E_1 = \{b\}$, we can perform the split by simply removing the shared support. The split correctly deduces that $\{b\}$ and $\{b, d\}$ are admissible in F. Such an easy solution of simply removing the shared support is, however, not always possible.

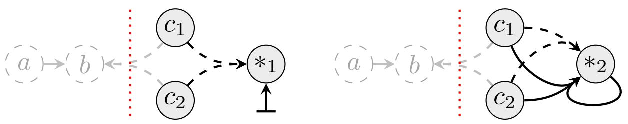
Example 34. *We consider the BSAF F below, with F_1 and F_2 left resp. right of the dotted line and $S_3 = \{(\{c_1, c_2\}, b)\}$.*



Let $E_1 = \{a\}$. Since b is defeated, not both c_1 and c_2 can be accepted in F_2; otherwise, b would be accepted in F.

The example above shows that defeating the head of a shared support induces a constraint. There are two natural options to encode constraints in BSAFs; and we will make use of both of them. Given $(T, h) \in S_3$, the first option is to add a new argument $*_1$ which is attacked by the empty set and supported by $T \cap A_2$ (*type-1-constraint*); the second option is to add a new argument $*_2$ which is attacked by $(T \cap A_2) \cup \{*_2\}$ and supported by $T \cap A_2$ (*type-2-constraint*).

Example 35. *Consider the BSAF from Example 34. We can prevent the joint acceptance of c_1 and c_2 in F_2 by applying one of the following modifications. On the left-hand side, we add an argument that is attacked by the empty set ($*_1$), on the right-hand side, we add an argument ($*_2$) which is the head of the self-attacking set.*



We observe that the type-1-constraint (left) defeats all attacks originating from (supersets of) $T = \{c_1, c_2\}$. This is because the closure of T is attacked by the empty set.

We break down which constraint is appropriate in which situation. First, we observe that a shared link (T, h) induces a constraint whenever all arguments in $T \cap A_1$ are contained in the selected extension E_1, but $h \notin E_1$.

Definition 36. *Let (F_1, F_2, S_3) be a support splitting for BSAF F, where $F_1 = (A_1, R_1, S_1)$. Let $E_1 \subseteq A_1$. By*

$$\mathcal{T}_{S_3}(E_1) := \{T \subseteq A \mid \exists h \in A_1 : (T, h) \in S_3, \; T \cap A_1 \subseteq E_1, \; h \notin E_1\}$$

we denote the collection of sets of arguments that are support-incompatible *with respect to E_1 and S_3.*

As observed in Example 35, adding a type-1-constraint for a shared link (T, h) by adding $*_1$ which is attacked by the empty set and a support from $T \cap A_2$ implies that all attacks originating from $T \cap A_2$ are defeated. We can apply this construction therefore only in the very restricted setting in which the set T is already defeated in F_1. We also require $T \cap A_1 = \emptyset$ because in all other cases, no outgoing attacks from T exist (recall that F_1 and F_2 do not share attacks).

Definition 37. *Let (F_1, F_2, S_3) be a support splitting for BSAF F, where $F_1 = (A_1, R_1, S_1)$. Let $E_1 \subseteq A_1$. By*

$$\mathcal{D}_{S_3}(E_1) := \{T \subseteq A \mid T \in \mathcal{T}_{S_3}(E_1), \; T \cap A_1 = \emptyset, \; cl_F(T) \cap (E_1)^+_R \neq \emptyset\}$$

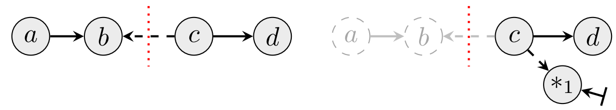
we denote the collection of sets of arguments that are closure-defeated *with respect to E_1 and S_3.*

We are ready to state our first modification.

Definition 38 (Type-1-Modification)**.** *Let (F_1, F_2, S_3) be a support splitting for a BSAF F, let E_1 be an extension of F_1 and let $\mathcal{D}_{S_3}(E_1)$ denote all sets which are closure-defeated wrt. E_1 and S_3. We define the* type-1-modification *of F_2 as the BSAF $F_2^{C_1} := F_2$ if $\mathcal{D}_{S_3}(E_1) = \emptyset$ and, otherwise, $F_2^{C_1} := (A_2 \cup \{*_1\}, R_2^{C_1}, S_2^{C_1})$ with*

$$R_2^{C_1} := R_2 \cup \{(\emptyset, *_1)\}$$
$$S_2^{C_1} := S_2 \cup \{(T \cap A_2, *_1) \mid T \in \mathcal{D}_{S_3}(E_1)\}.$$

Example 39. *Consider the BSAF F with F_1 (left) and F_2 (right) and the shared support (c, b) below. Let $E_1 = \{a\}$.*



In F, $cl_F(\{c\}) = \{c, b\}$, is attacked, thus, d is defeated. This is correctly captured by the type-1-modification $F_2^{C_1}$: the set $cl_{F_2^{C_1}}(\{c\}) = \{c, *_1\}$ is attacked by the empty set.

We consider an example where this construction fails.

Example 40. *Consider the BSAF F with F_1 (left) and F_2 (right) and the shared support $(\{c, e\}, b)$ below. Let $E_1 \in \{\{e\}, \{a, e\}\}$, then $\{c, e\} \in \mathcal{T}_{S_3}(E_1)$.*

Neither $\{e, d\}$ nor $\{a, e, d\}$ are admissible in F. However, in $F_2^{C_1}$, argument d is defended since $cl_{F_2^{C_1}}(c)$ is attacked.

We thus require constraints of the second type in this case.

Definition 41 (Type-2-Modification)**.** *Let (F_1, F_2, S_3) be a support splitting for a BSAF F, let E_1 be an extension of F_1 and let $\mathcal{D}_{S_3}(E_1)$ and $\mathcal{T}_{S_3}(E_1)$ denote all sets which are closure-defeated resp. support-incompatible wrt. E_1 and S_3. Let $\mathcal{T}_{S_3}^{-\mathcal{D}}(E_1) := \mathcal{T}_{S_3}(E_1) \setminus \mathcal{D}_{S_3}(E_1)$ We define the* type-2-modification *of F_2 as the BSAF $F_2^{C_2} := F_2$ if $\mathcal{T}_{S_3}^{-\mathcal{D}}(E_1) = \emptyset$ and, otherwise, $F_2^{C_2} := (A_2 \cup \{*_2\}, R_2^{C_2}, S_2^{C_2})$ with*

$$R_2^{C_2} := R_2 \cup \{((T \cap A_2) \cup \{*_2\}, *_2) \mid T \in \mathcal{T}_{S_3}^{-\mathcal{D}}(E_1)\}$$
$$S_2^{C_2} := S_2 \cup \{(T \cap A_2, *_2) \mid T \in \mathcal{T}_{S_3}^{-\mathcal{D}}(E_1)\}$$

Example 42. *We continue Example 40 and apply the type-2-modification for $E_1 \in \{\{e\}, \{a, e\}\}$.*

Now, in $F_2^{C_2}$, the argument c supports $*_2$ and d is attacked; thus, no argument can be accepted and the only admissible extension is $\emptyset$. This correctly detects the admissible extensions $\{e\}$ and $\{a, e\}$ of F. For $E_1 = \{a\}$, we have $\{e, c\} \notin \mathcal{T}_{S_3}(E_1)$, thus the support is simply removed.

We define the *S-reduct* as a combination of both modifications. The union of two BSAFs is defined componentwise.

Definition 43 (S-reduct)**.** *Let (F_1, F_2, S_3) be a support splitting for a BSAF F, let E_1 be an extension of F_1 and let $\mathcal{D}_{S_3}(E_1)$ and $\mathcal{T}_{S_3}(E_1)$ denote all sets which are closure-defeated resp. support-incompatible wrt. E_1 and S_3. We define the* (E_1, S_3)-reduct *(or simply* S-reduct*) of F_2 as the BSAF $F_2^{E_1} = (A_2^{E_1}, R_2^{E_1}, S_2^{E_1})$ where $F_2^{E_1} := F_2^{C_1} \cup F_2^{C_2}$.*

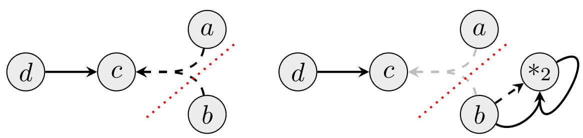
The splitting procedure preserves admissible, complete and stable semantics.

Theorem 44. *Let (F_1, F_2, S_3) be a support splitting for a BSAF $F = (A, R, S)$ with $F_1 = (A_1, R_1, S_1)$, $F_2 = (A_2, R_2, S_2)$, and $\sigma \in \{stb, adm, com\}$. Further, let $F_2^{E_1}$ be the S-reduct of F_2.*

1. *If $E_1 \in \sigma(F_1)$ and $E_2 \in \sigma(F_2^{E_1})$, then $E_1 \cup (E_2 \setminus \{*_2\}) \in \sigma(F)$.*
2. *If $E \in \sigma(F)$, then $E_1 = E \cap A_1 \in \sigma(F_1)$ and $(E \cap A_2 \in \sigma(F_2^{E_1})$ or $(E \cap A_2) \cup \{*_2\} \in \sigma(F_2^{E_1}))$.*

For preferred and grounded semantics, our procedure constructs only some of the extensions of the entire framework. We illustrate this in the following example.

Example 45. *Consider the BSAF F and a support splitting (F_1, F_2, S_3) as depicted below. Let $E_1 = \{a, d\}$. We compute the reduct $F_2^{E_1}$ by adding the dummy argument $*_2$ which is supported by $\{b\}$ and attacked by $\{b, *_2\}$ (depicted right).*



In $F_2^{E_1}$, we get $pref(F_2^{E_1}) = \{\emptyset\}$, obtaining only $\{a, d\}$ as preferred extension of the whole framework.

We get a splitting theorem for preferred and grounded semantics that considers only the first direction.

Theorem 46. *Let (F_1, F_2, S_3) be a support splitting for a BSAF $F = (A, R, S)$, and let $F_2^{E_1}$ be the S-reduct of F_2 and $\sigma \in \{grd, pref\}$. If $E_1 \in \sigma(F_1)$ and $E_2 \in \sigma(F_2^{E_1})$, then $E_1 \cup (E_2 \setminus \{*_2\}) \in \sigma(F)$.*

5 Splitting Collective Attacks and Supports

Now that we have the building blocks of our approach, we combine the notions of attack and support splitting and provide results similar to the section before.

Definition 47 (Splitting)**.** *Let $F = (A, R, S)$ be a BSAF, $F_1 = (A_1, R_1, S_1)$ and $F_2 = (A_2, R_2, S_2)$ two subframeworks of F such that $A_1 \cap A_2 = \emptyset$, $A = A_1 \cup A_2$, $R = R_1 \cup R_2 \cup R_3$, and $S = S_1 \cup S_2 \cup S_3$ with:*

$$R_3 \subseteq \{(T, h) \in R \mid T \cap A_1 \neq \emptyset, T \subseteq A, h \in A_2\},$$
$$S_3 \subseteq \{(T, h) \in S \mid T \cap A_2 \neq \emptyset, T \subseteq A, h \in A_1\}.$$

We call the 4-tuple (F_1, F_2, R_3, S_3) a splitting *of F.*

We apply the splits we developed in the previous two sections consecutively. Since attacks (T, h) in F_2 can support arguments in F_1, we start our splitting procedure by closing attacks in R_2, before we apply the modifications we developed in Sections 3 and 4. Overall, we perform the following splitting procedure. Given a splitting (F_1, F_2, R_3, S_3) for a BSAF F and a set $E_1 \subseteq A_1$, we proceed as follows.

1. We close the attacks in $R_2 \cup R_3$. Note that this may transform R_2 attacks into shared attacks between F_1 and F_2 since the tail of an attack R_2 can support an argument in A_1. To account for this, we move the attacks in question into the set R_3; by $\hat{F}_2$ and $\hat{R}_3$, we denote the adjusted BSAF and set of common negative links, respectively.
2. We proceed by constructing the R-reduct $\hat{F}_2^{E_1}$ of the attack splitting $(F_1, \hat{F}_2, \hat{R}_3)$ wrt. E_1, following Definition 22.
3. Next, we compute the modification $F_2^\star = mod_{\hat{R}_3}^{E_1}(\hat{F}_2^{E_1})$ as in Definition 25. The tuple $(F_1, F_2^\star, S_3)$ is a support splitting for the BSAF $(A_1 \cup A_2^\star, R_1 \cup R_2^\star, S)$.
4. Finally, we construct the S-reduct $F_2^\circledast$ of $F_2^\star$ for the support splitting $(F_1, F_2^\star, S_3)$ wrt. E_1, using Definition 43.

Definition 48. *Let (F_1, F_2, R_3, S_3) be a splitting for a BSAF F with $F_1 = (A_1, R_1, S_1)$, $F_2 = (A_2, R_2, S_2)$. Let R_3^c and R_2^c denote the sets of closed negative links wrt. R_3 and R_2, respectively. Define*

$$\hat{R}_2 := \{(T, h) \in R_2^c \mid T \cap A_1 = \emptyset\}$$

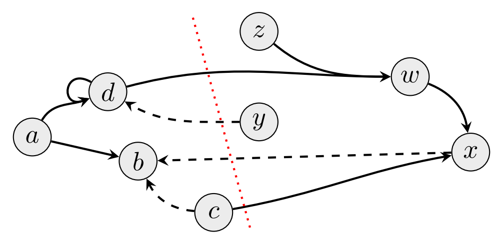
and let $\hat{F}_2 := (A_2, \hat{R}_2, S_2)$ be the updated BSAF. Further, let

$$\hat{R}_3 := \{(T, h) \mid (T, h) \in R_3^c \cup R_2^c, T \cap A_1 \neq \emptyset\}.$$

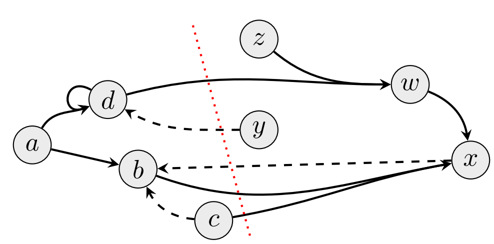
Let $\hat{F}_2^{E_1}$ be the R-reduct of the attack splitting $(F_1, \hat{F}_2, \hat{R}_3)$ wrt. E_1 and $F_2^\star := mod_{\hat{R}_3}^{E_1}(\hat{F}_2^{E_1})$. Finally, we let $F_2^\circledast$ denote the S-reduct for the support splitting $(F_1, F_2^\star, S_3)$ of the BSAF $(A_1 \cup A_2^\star, A_1 \cup A_2^\star, S)$ wrt. E_1.

We demonstrate the procedure in the following example.

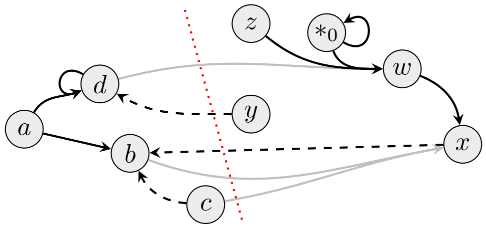
Example 49. *We consider a BSAF F with splitting (F_1, F_2, R_3, S_3), as depicted below. Further, let $E_1 = \{a\}$.*



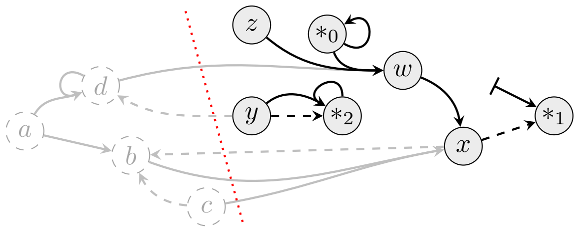
First, we proceed by closing the negative links in R_2 and R_3, as depicted below. The set R_3 contains the single attack (c, x). Since c supports b, the attack $(\{b, c\}, x)$ is contained in $\hat{R}_3$. Moreover, we get $\hat{R}_2 = R_2$ because x and y do not attack any argument in F_2. Thus we consider the following attack splitting $(F_1, F_2, \hat{R}_3)$.



We apply the R-reduct and the modification for the attack splitting (see Section 3). We depict F_1 (left) and the result of the procedure, $F_2^\star$, (right) below.



*Finally, we apply the S-reduct to handle the remaining links. Thus, we obtain $F_2^{\circledast}$ (*right*) adding a type-1-constraint on x and a type-2-constraint on y.*



We are ready to state our main result for splitting BSAFs.

Theorem 50. *Let (F_1, F_2, R_3, S_3) be a splitting for a BSAF $F = (A, R, S)$ with $F_1 = (A_1, R_1, S_1)$, $F_2 = (A_2, R_2, S_2)$, and $\sigma \in \{stb, adm, com\}$. Let $F_2^{\circledast}$ be defined as in Definition 48.*

1. *If $E_1 \in \sigma(F_1)$ and $E_2 \in \sigma(F_2^{\circledast})$, then $E_1 \cup (E_2 \setminus \{*_2\}) \in \sigma(F)$.*
2. *If $E \in \sigma(F)$, then $E_1 = E \cap A_1 \in \sigma(F_1)$ and $E \cap A_2 \in \sigma(F_2^{\circledast})$ or $(E \cap A_2) \cup \{*_2\} \in \sigma(F_2^{\circledast})$.*

Example 51. *We use the splitting theorem to compute the extensions of BSAF F from Example 49. First, note that the set $E_1 = \{a\}$ is admissible in F_1. Next, we compute the admissible extensions of $F_2^{\circledast}$ and obtain the $\emptyset$ and $\{z\}$. Hence, we retrieve $\{a\}$ and $\{a, z\}$ in adm(F).*

Note that we inherit the limitations with respect to preferred and grounded semantics.

Theorem 52. *Let (F_1, F_2, R_3, S_3) be a splitting for the BSAF $F = (A, R, S)$, let $F_2^{\circledast}$ be as in Definition 48 and $\sigma \in \{grd, pref\}$. If $E_1 \in \sigma(F_1)$ and $E_2 \in \sigma(F_2^{\circledast})$, then $E_1 \cup (E_2 \setminus \{*_2\}) \in \sigma(F)$.*

6 Related Work

Splitting has a long tradition in the field of knowledge representation and reasoning, in particular, for non-monotonic formalisms. The first splitting procedure has been proposed for logic programs (Lifschitz and Turner 1994) and has received much attention since then, e.g., in the context of answer set programming (Beiser et al. 2024), extended fragments of logic programming (Ben-Eliyahu-Zohary 2025; Cabalar, Fandinno, and del Cerro 2021), for conditional knowledge bases (Heyninck et al. 2023), and for abstract and structured forms of argumentation (Baumann 2011; Baumann, Brewka, and Wong 2011; Buraglio et al. 2024; Linsbichler 2014; Buraglio 2025).

Incremental Computation for Supports While most works in computational argumentation focus on splitting attacks, ADFs constitute a notable exception. In ADFs, each argument has an acceptance condition, capable of expressing positive relations between arguments. Linsbichler (2014) investigated splitting techniques for ADFs, while more recent work has explored serialization (Bengel and Thimm 2025) for abstract dialectical frameworks. The idea behind serialization is based on finding so-called *initial models* of a given ADF in an iterative way, thereby constructing serialization sequences that allow the incremental computation of its extensions. We notice, however, that even though ADFs have been proven to capture collective attacks (Dvořák, Keshavarzi Zafarghandi, and Woltran 2023), this does not hold for collective supports under the closure condition. From a semantic standpoint, this constitutes a substantial difference with respect to BSAFs. Consequently, existing incremental and splitting results for ADFs cannot be transferred directly to our setting.

Directionality, Modularization, and SCC-recursiveness Further approaches that compute extensions in an incremental manner closely related to splitting techniques include schemes based on strongly connected components (SCCs) (Baroni, Giacomin, and Guida 2005), modularization (Baumann, Brewka, and Ulbricht 2022) and directionality (Baroni and Giacomin 2007). Modularization is directly connected to the splitting schema via a similar notion of reduct (Berthold, Blümel, and Rapberger 2025). The principle of directionality is connected with the possibility of evaluating sub-frameworks independently from the remaining parts, provided that there is no incoming negative link towards them. The relation to attack splitting is thus clear given that we can project extensions of F to those of F_1 without losing information. The SCC-recursive property (Baroni, Giacomin, and Guida 2005) of argumentation semantics defines the possibility to retrieve extensions of a given framework by breaking down the computation along its SCCs. Such decomposition ensures that the evaluation of the whole graph can be assessed via a generalized semantics that takes into account previous decisions concerning already evaluated SCCs. Splitting represents a more flexible alternative as it does not dictate that F_1 and F_2 be strongly connected components, while ensuring that every SCC of the graph is in either of the two.

7 Conclusion

In this paper, we introduced a splitting schema for bipolar set-based argumentation frameworks. Similar to existing literature on splitting for abstract forms of argumentation, we developed a modification-based approach for the incremental computation of extensions of a given BSAF. In particular, after splitting the framework F into two sub-frameworks F_1 and F_2, we defined syntactic modifications of F_2 that allow one to reconstruct extensions of the original framework from those of the components. By leveraging the richer syntax of BSAFs, we have considered splits over collective attacks, supports, and a combination of both. Importantly, each step of our procedure is computationally feasible and does not induce an exponential blow-up in the size of the modified sub-frameworks, making our approach a promising starting point for efficient algorithms.

For future work, we plan to extend our splitting techniques to cases where (a certain number of) backward attacks and forward supports are present. In the context of AFs, similar settings have been studied under the term *parameterized splitting* (Baumann et al. 2012). In addition, we plan to implement our procedure and investigate its benefits for dynamic programming approaches in argumentation. In this regard, observe that for Dung-style AFs, all possible splittings can be computed in linear time by identifying the strongly connected components of the graph (Baumann, Brewka, and Wong 2011; Tarjan 1972). We anticipate that for BSAFs, a similar approach to computing splittings in AFs can be adapted, by treating the support and attack relations separately and by exploiting the so-called *primal graph* (Dvořák et al. 2024). On top of this, Baumann, Brewka, and Wong (2011) register an average acceleration of 60% for executions with splitting in comparison to an execution without splitting in the context of Dung-style AFs.

Crucially, BSAFs are closely related to general (non-flat) ABAFs, as demonstrated by Berthold, Rapberger, and Ulbricht (2024), which is one of the most popular formalisms in the area of structured argumentation. Recent work exploits the close relation of SETAFs and flat ABAFs to translate SETAF splitting results to this fragment of ABA (Buraglio 2025). We thus expect that our results for BSAF splitting can be transferred to general ABA in a similar fashion. Our hope is that this will enhance existing non-flat ABA solvers (Lehtonen et al. 2024). We are furthermore optimistic that our results could provide further valuable insights into structured argumentation dynamics (Rapberger and Ulbricht 2024; Rapberger and Ulbricht 2023; Prakken 2023; Baumann and Brewka 2010; Cayrol, de Saint-Cyr, and Lagasquie-Schiex 2010; Berthold, Rapberger, and Ulbricht 2023) and other popular forms of structured argumentation (Modgil and Prakken 2013).

While the present work has focused on the canonical semantics for BSAFs (which capture ABA semantics), the literature on bipolar argumentation provides a wide range of alternative interpretations of support (Cohen et al. 2014; Cohen et al. 2018; Amgoud et al. 2008; Polberg 2016; Boella et al. 2010). Furthermore, (Berthold, Rapberger, and Ulbricht 2024) develop novel BSAF semantics aimed at addressing certain undesirable properties of the canonical approach. Extending established notions of support to the setting of set-supports, as well as developing corresponding splitting techniques for these alternative BSAF semantics, is a promising direction for future research.

Lastly, we plan to study the relation of our splitting techniques to the aforementioned principles of directionality, modularity and SCC-recursiveness. For doing so, appropriate notions of these principles in the presence of supports need to be identified, which we deem an interesting direction of future work on its own. As suitable starting points for these studies, we identify the principle-based analyzes for SETAFs (Dvořák et al. 2024) and BAFs (Yu et al. 2023).

AI Declaration

The authors have not employed any Generative AI tools.

Acknowledgments

This work has been supported by the European Union's Horizon 2020 research and innovation programme (under grant agreement 101034440) and by the Deutsche Forschungsgemeinschaft (projects "Argumentative Reasoning in Nonsensical Situations", grant 550735820, and "Explainable Belief Merging", grant 465447331).

References

Amgoud, L.; Cayrol, C.; Lagasquie, M.-C.; and Livet, P. 2008. On bipolarity in argumentation frameworks. *International Journal of Intelligent Systems* 23:1–32.

Atkinson, K.; Baroni, P.; Giacomin, M.; Hunter, A.; Prakken, H.; Reed, C.; Simari, G. R.; Thimm, M.; and Villata, S. 2017. Towards artificial argumentation. *AI Magazine* 38(3):25–36.

Baroni, P., and Giacomin, M. 2007. On principle-based evaluation of extension-based argumentation semantics. *Artificial Intelligence* 171(10-15):675–700.

Baroni, P.; Gabbay, D.; Giacomin, M.; and van der Torre, L., eds. 2018. *Handbook of Formal Argumentation*. College Publications.

Baroni, P.; Giacomin, M.; and Guida, G. 2005. SCC-recursiveness: A general schema for argumentation semantics. *Artificial Intelligence* 168(1-2):162–210.

Baroni, P.; Giacomin, M.; and Liao, B. 2014. On topology-related properties of abstract argumentation semantics. A correction and extension to dynamics of argumentation systems: A division-based method. *Artificial Intelligence* 212:104–115.

Baumann, R., and Brewka, G. 2010. Expanding argumentation frameworks: Enforcing and monotonicity results. In Baroni, P.; Cerutti, F.; Giacomin, M.; and Simari, G. R., eds., *Proceedings of (COMMA-10)*, volume 216 of *Frontiers in Artificial Intelligence and Applications*, 75–86. IOS Press.

Baumann, R.; Brewka, G.; Dvořák, W.; and Woltran, S. 2012. Parameterized splitting: A simple modification-based approach. In *Correct Reasoning - Essays on Logic-Based AI in Honour of Vladimir Lifschitz*, volume 7265 of *LNCS*, 57–71. Springer.

Baumann, R.; Brewka, G.; and Ulbricht, M. 2022. Shedding new light on the foundations of abstract argumentation: Modularization and weak admissibility. *Artificial Intelligence* 310:103742.

Baumann, R.; Brewka, G.; and Wong, R. 2011. Splitting argumentation frameworks: An empirical evaluation. In Modgil, S.; Oren, N.; and Toni, F., eds., *Proceedings of (TAFA-11)*, volume 7132 of *Lecture Notes in Computer Science*, 17–31. Springer.

Baumann, R. 2011. Splitting an argumentation framework. In *Proceedings of (LPNMR-11)*, volume 6645 of *LNCS*, 40–53. Springer.

Beiser, A.; Hecher, M.; Unalan, K.; and Woltran, S. 2024. Bypassing the ASP bottleneck: Hybrid grounding by splitting and rewriting. In *Proceedings of (IJCAI-24)*, 3250–3258. ijcai.org.

Ben-Eliyahu-Zohary, R. 2025. Splitting a disjunctive logic program. *Theory and Practice of Logic Programming* 25(4):507–521.

Bengel, L., and Thimm, M. 2025. Initial models and serialisability in abstract dialectical frameworks. In *Proceedings of (IJCAI-25)*, 4365–4373. ijcai.org.

Berthold, M.; Blümel, L.; and Rapberger, A. 2025. On strong and weak admissibility in non-flat assumption-based argumentation. In *Proceedings of (KR-25)*, 137–147.

Berthold, M.; Rapberger, A.; and Ulbricht, M. 2023. Forgetting aspects in assumption-based argumentation. In Marquis, P.; Son, T. C.; and Kern-Isberner, G., eds., *Proceedings of (KR-23)*, 86–96.

Berthold, M.; Rapberger, A.; and Ulbricht, M. 2024. Capturing non-flat assumption-based argumentation with bipolar setafs. In Marquis, P.; Ortiz, M.; and Pagnucco, M., eds., *Proceedings of (KR-24)*.

Blümel, L.; Rapberger, A.; Thimm, M.; and Toni, F. 2025. On independence and scc-recursiveness in assumption-based argumentation. In *Proceedings of (IJCAI-25)*, 4382–4390. ijcai.org.

Boella, G.; Gabbay, D. M.; van der Torre, L. W. N.; and Villata, S. 2010. Support in abstract argumentation. In Baroni, P.; Cerutti, F.; aspects in assumption-based argumenno Giacomin, M.; and Simari, G. R., eds., *Proceedings of (COMMA-10)*, Frontiers in Artificial Intelligence and Applications, 111–122. IOS Press.

Brewka, G.; Ellmauthaler, S.; Strass, H.; Wallner, J. P.; and Woltran, S. 2018. Abstract dialectical frameworks. In *Handbook of Formal Argumentation*. College Publications. chapter 5, 237–285. also appears in IfCoLog Journal of Logics and their Applications 4(8):2263–2318.

Buraglio, G.; Dvorák, W.; König, M.; and Woltran, S. 2024. Splitting argumentation frameworks with collective attacks. In Borg, A.; Ellmauthaler, S.; Mailly, J.; and Niskanen, A., eds., *Proceedings of (COMMA-24)*, volume 3757 of *CEUR Workshop Proceedings*, 41–55. CEUR-WS.org.

Buraglio, G. 2025. Splitting assumption-based argumentation frameworks. In Rapberger, A., and Rudolph, S., eds., *Proceedings of (NMR-25)*, volume 4071 of *CEUR Workshop Proceedings*, 17–31. CEUR-WS.org.

Cabalar, P.; Fandinno, J.; and del Cerro, L. F. 2021. Splitting epistemic logic programs. *Theory and Practice of Logic Programming* 21(3):296–316.

Cayrol, C., and Lagasquie-Schiex, M.-C. 2005. On the acceptability of arguments in bipolar argumentation frameworks. In Godo, L., ed., *Proceedings of (ECSQARU-05)*, volume 3571 of *LNCS*, 378–389. Springer.

Cayrol, C.; de Saint-Cyr, F. D.; and Lagasquie-Schiex, M.-C. 2010. Change in abstract argumentation frameworks: Adding an argument. *Journal of Artificial Intelligence Research (JAIR)* 38:49–84.

Cohen, A.; Gottifredi, S.; García, A. J.; and Simari, G. R. 2014. A survey of different approaches to support in argumentation systems. *The Knowledge Engineering Review* 29(5):513–550.

Cohen, A.; Parsons, S.; Sklar, E. I.; and McBurney, P. 2018. A characterization of types of support between structured arguments and their relationship with support in abstract argumentation. *International Journal of Approximate Reasoning* 94:76–104.

Cyras, K.; Rago, A.; Albini, E.; Baroni, P.; and Toni, F. 2021. Argumentative XAI: A survey. In Zhou, Z., ed., *Proceedings of (IJCAI-21)*, 4392–4399. ijcai.org.

Dung, P. M. 1995. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence* 77(2):321–358.

Dvořák, W.; König, M.; Ulbricht, M.; and Woltran, S. 2024. Principles and their computational consequences for argumentation frameworks with collective attacks. *Journal of Artificial Intelligence Research (JAIR)* 79:69–136.

Dvořák, W.; Keshavarzi Zafarghandi, A.; and Woltran, S. 2023. Expressiveness of SETAFs and support-free ADFs under 3-valued semantics. *Journal of Applied Non-Classical Logics* 33(3-4):298–327.

Giacomin, M.; Baroni, P.; and Cerutti, F. 2021. Towards a general theory of decomposability in abstract argumentation. In Baroni, P.; Benzmüller, C.; and Wáng, Y. N., eds., *Proceedings of (CLAR-21)*, Lecture Notes in Computer Science, 169–189. Springer.

Heyninck, J.; Kern-Isberner, G.; Meyer, T. A.; Haldimann, J. P.; and Beierle, C. 2023. Conditional syntax splitting for non-monotonic inference operators. In Williams, B.; Chen, Y.; and Neville, J., eds., *Proceedings of (AAAI-23)*, 6416–6424. AAAI Press.

Lehtonen, T.; Rapberger, A.; Toni, F.; Ulbricht, M.; and Wallner, J. P. 2024. Instantiations and computational aspects of non-flat assumption-based argumentation. In *Proceedings of (IJCAI-24)*.

Leofante, F.; Ayoobi, H.; Dejl, A.; Freedman, G.; Gorur, D.; Jiang, J.; Paulino-Passos, G.; Rago, A.; Rapberger, A.; Russo, F.; Yin, X.; Zhang, D.; and Toni, F. 2024. Contestable AI needs computational argumentation. In Marquis, P.; Ortiz, M.; and Pagnucco, M., eds., *Proceedings of (KR-24)*.

Liao, B. S.; Jin, L.; and Koons, R. C. 2011. Dynamics of argumentation systems: A division-based method. *Artificial Intelligence* 175(11):1790–1814.

Liao, B. 2013. Toward incremental computation of argumentation semantics: A decomposition-based approach. *Annals of Mathematics and Artificial Intelligence* 67(3-4):319–358.

Lifschitz, V., and Turner, H. 1994. Splitting a logic program. In Hentenryck, P. V., ed., *Proceedings of (ICLP-94)*, 23–37. MIT Press.

Linsbichler, T. 2014. Splitting abstract dialectical frameworks. In *Proceedings of (COMMA-14)*, volume 266 of

Frontiers in Artificial Intelligence and Applications, 357–368. IOS Press.

Modgil, S., and Prakken, H. 2013. A general account of argumentation with preferences. *Artificial Intelligence* 195:361–397.

Modgil, S. 2009. Reasoning about preferences in argumentation frameworks. *Artificial Intelligence* 173(9-10):901–934.

Nielsen, S. H., and Parsons, S. 2006. A generalization of Dung's abstract framework for argumentation: Arguing with sets of attacking arguments. In *Proceedings of (ArgMAS-06)*, 54–73. Springer.

Polberg, S. 2016. Understanding the abstract dialectical framework. In *Proceedings of (JELIA-16)*, Lecture Notes in Computer Science, 430–446.

Prakken, H. 2023. Relating abstract and structured accounts of argumentation dynamics: the case of expansions. In Marquis, P.; Son, T. C.; and Kern-Isberner, G., eds., *Proceedings of (KR-23)*, 562–571.

Rapberger, A., and Ulbricht, M. 2023. On dynamics in structured argumentation formalisms. *Journal of Artificial Intelligence Research (JAIR)* 77:563–643.

Rapberger, A., and Ulbricht, M. 2024. Repairing assumption-based argumentation frameworks. In Marquis, P.; Ortiz, M.; and Pagnucco, M., eds., *Proceedings of (KR-24)*.

Tarjan, R. E. 1972. Depth-first search and linear graph algorithms. *SIAM Journal on Computing* 1(2):146–160.

Ulbricht, M.; Potyka, N.; Rapberger, A.; and Toni, F. 2024. Non-flat ABA is an instance of bipolar argumentation. In Wooldridge, M. J.; Dy, J. G.; and Natarajan, S., eds., *Proceedings of (AAAI-24)*, 10723–10731. AAAI Press.

Yu, L.; Anaissy, C. A.; Vesic, S.; Li, X.; and van der Torre, L. 2023. A principle-based analysis of bipolar argumentation semantics. In Gaggl, S. A.; Martinez, M. V.; and Ortiz, M., eds., *Proceedings of (JELIA-23)*, volume 14281 of *Lecture Notes in Computer Science*, 209–224. Springer.