A Discussion Protocol for Grounded Semantics
(proofs)

Martin Caminada

University of Aberdeen

Abstract. We introduce an argument-based discussion game where the ability to win the game for a particular argument coincides with the argument being in the grounded extension. Our game differs from previous work in that (i) the number of moves is *linear* (instead of exponential) w.r.t. the strongly admissible set that the game is constructing, (ii) winning the game does not rely on cooperation from the other player (that is, the game is winning strategy based), (iii) a *single* game won by the proponent is sufficient to show grounded membership, and (iv) the game has a number of properties that make it more in line with natural discussion.

1 Introduction

In informal, human style argumentation, discussions play a prominent role. Yet the aspect of discussion has received relatively little attention in formal argumentation theory, especially within the research line of Dung-style argumentation [13]. Whereas other aspects of informal argumentation, like argument schemes [21], claims and conclusions [21, 15], assumptions [2, 14] and preferences [18, 20] have successfully been modelled in the context of (instantiated) Dung-style argumentation, dialectical aspects are often regarded as being part of a research field separate from inference-based argumentation [22, 24]. The scarce work that does consider dialectical aspects in the context of argument-based entailment tends to do so for the purpose of defining proof procedures [12, 25] that, although useful for software implementations [23] are not meant to actually resemble informal discussion.

One exception to this is the Grounded Persuasion Game of Caminada and Podlaszewski [10], which provides a labelling-based discussion game for grounded semantics. The game is defined in such a way that an argument is in the grounded extension iff there exists at least one game for it that is won by the proponent [10]. However, the Grounded Persuasion Game has a number of shortcomings. For instance, it can be that an argument is in the grounded extension but the proponent does not have a winning strategy for it. That is, although it is possible to win the game, this depends partly on the cooperation of the opponent. Furthermore, in the Grounded Persuasion Game it is the proponent who first introduces the arguments that he later needs to defend against, a phenomenon that rarely occurs in natural discussions other than by mistake.

In the current paper, we present a modified and slightly simplified discussion game for grounded semantics, called the Grounded Discussion Game, that addresses above mentioned shortcomings. Overall, our aim is to provide a discussion game that can be used in the context of human-computer interaction, for the purpose of explaining argument-based inference. This can be helpful to allow users to understand why a particular advice was given by a knowledge-based system, and to examine whether particular objections the user might have can properly be addressed. In this way, we see interactive discussion as an alternative for argument visualisation [26, 27]. Our current work, which is focussed on grounded semantics, fits in a line of research where similar discussion games have been stated also for preferred [8] and stable [11]. With respect to the previously stated games for grounded semantics [25, 4, 19, 10] our aim is to satisfy the following properties:

1. Correctness and completeness for grounded semantics w.r.t. the presence of a winning strategy. It should be the case that an argument is in the grounded extension iff the proponent has a winning strategy for it (unlike is the case in for instance [10]).
2. Similarity to natural discussion. No party should be required to introduce arguments that he subsequently has to argue against (unlike for instance in [10]). Also, there should be moves in which a player can indicate agreement (“fair enough”) at specific points of the discussion (unlike is the case in for instance the Standard Grounded Game [25, 4, 19], where such moves are absent).

3. Efficiency. The number of moves should be *linear* in relation to the size[1] of the strongly admissible labelling [7] the game is constructing. This is for instance violated in the Standard Grounded Game [25, 4, 19], where the number of moves can be *exponential* in relation to the size of the strictly admissible labelling the game is constructing (see [7, Section 5.3] for details). A similar observation can be made for other tree-based proof procedures [12].

The remaining part of this paper is structured as follows. First, in Section 2 we provide some preliminaries of argumentation theory. Then, in Section 3 we present our new Grounded Discussion Game, and show that it satisfies the above mentioned properties. We round off in Section 4 with a discussion of the obtained results how these relate to previous research.

2 Formal Preliminaries

Abstract argumentation theory [13] is in essence about how to select nodes from a graph (called an argumentation framework). In the current paper, we restrict ourselves to finite graphs.

Definition 1 ([13]). *An* argumentation framework *is a pair* (Ar, att) *where* Ar *is a finite set of entities, called arguments, whose internal structure can be left unspecified, and* att *is a binary relation on* Ar*. We say that* A *attacks* B *iff* $(A, B) \in att$.

For current purposes, we apply the labelling-based version of argumentation semantics [5, 9], instead of the original extension-based version of [13]. It should be noticed, however, that an extension is essentially the `in` labelled part of a labelling [5, 9].

Definition 2 ([9]). *Let* (Ar, att) *be an argumentation framework. An* argument labelling *is a total function* $\mathcal{L}ab : Ar \rightarrow \{\texttt{in}, \texttt{out}, \texttt{undec}\}$*. An argument labelling is called an* admissible labelling *iff for each* $A \in Ar$ *it holds that:*

- *if* $\mathcal{L}ab(A) = \texttt{in}$ *then for each* B *that attacks* A *it holds that* $\mathcal{L}ab(B) = \texttt{out}$
- *if* $\mathcal{L}ab(A) = \texttt{out}$ *then there exists a* B *that attacks* A *such that* $\mathcal{L}ab(B) = \texttt{in}$

$\mathcal{L}ab$ *is called a* complete labelling *iff it is an admissible labelling and for each* $A \in Ar$ *it also holds that:*

- *if* $\mathcal{L}ab(A) = \texttt{undec}$ *then not for each* B *that attacks* A *it holds that* $\mathcal{L}ab(B) = \texttt{out}$*, and there exists no* B *that attacks* A *such that* $\mathcal{L}ab(B) = \texttt{in}$

As a labelling is essentially a function, we sometimes write it as a set of pairs. Also, if $\mathcal{L}ab$ is a labelling, we write $\texttt{in}(\mathcal{L}ab)$ for $\{A \in Ar \mid \mathcal{L}ab(A) = \texttt{in}\}$, $\texttt{out}(\mathcal{L}ab)$ for $\{A \in Ar \mid \mathcal{L}ab(A) = \texttt{out}\}$ and $\texttt{undec}(\mathcal{L}ab)$ for $\{A \in Ar \mid \mathcal{L}ab(A) = \texttt{undec}\}$. As a labelling is also a partition of the arguments into sets of `in`-labelled arguments, `out`-labelled arguments and `undec`-labelled arguments, we sometimes write it as a triplet $(\texttt{in}(\mathcal{L}ab), \texttt{out}(\mathcal{L}ab), \texttt{undec}(\mathcal{L}ab))$.

Definition 3 ([9]). *Let* $\mathcal{L}ab$ *be a complete labelling of argumentation framework* $AF = (Ar, att)$*.* $\mathcal{L}ab$ *is said to be*

- *a grounded labelling iff* $\texttt{in}(\mathcal{L}ab)$ *is minimal (w.r.t. set inclusion) among all complete labellings of* AF*.*
- *a preferred labelling iff* $\texttt{in}(\mathcal{L}ab)$ *is a maximal (w.r.t. set inclusion) among all complete labellings of* AF*.*

The discussion game to be presented in Section 3 of this paper is based on the concept of strong admissibility [1, 7]. Hence, we will briefly recall some of its basic definitions.

Definition 4 ([7]). *Let* $\mathcal{L}ab$ *be an admissible labelling of argumentation framework* (Ar, att)*. A* min-max numbering *is a total function* $\mathcal{MM}_{\mathcal{L}ab} : \texttt{in}(\mathcal{L}ab) \cup \texttt{out}(\mathcal{L}ab) \rightarrow \mathbb{N} \cup \{\infty\}$ *such that for each* $A \in \texttt{in}(\mathcal{L}ab) \cup \texttt{out}(\mathcal{L}ab)$ *it holds that:*

---

[1] With the size of a labelling $\mathcal{L}ab$ we mean $|\texttt{in}(\mathcal{L}ab) \cup \texttt{out}(\mathcal{L}ab)|$.

- *if* $\mathcal{L}ab(A) = \texttt{in}$ *then* $\mathcal{MM}_{\mathcal{L}ab}(A) = max(\{\mathcal{MM}_{\mathcal{L}ab}(B) \mid B$ *attacks* A *and* $\mathcal{L}ab(B) = \texttt{out}\}) + 1$ *(with* $max(\emptyset)$ *defined as* 0*)*
- *if* $\mathcal{L}ab(A) = \texttt{out}$ *then* $\mathcal{MM}_{\mathcal{L}ab}(A) = min(\{\mathcal{MM}_{\mathcal{L}ab}(B) \mid B$ *attacks* A *and* $\mathcal{L}ab(B) = \texttt{in}\}) + 1$ *(with* $min(\emptyset)$ *defined as* ∞*)*

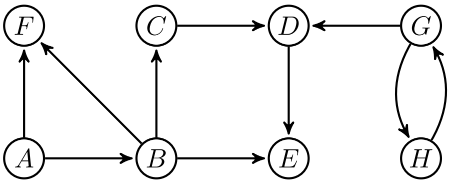
If $A \in Ar$ is labelled `in`, we sometimes refer to $max(\{\mathcal{MM}_{\mathcal{L}ab}(B) \mid B$ attacks A and $\mathcal{L}ab(B) = \texttt{out}\}) + 1$ as the MAX+1 value of the `out` labelled attackers of A. Also, if $A \in Ar$ is labelled `out`, we sometimes refer to $min(\{\mathcal{MM}_{\mathcal{L}ab}(B) \mid B$ attacks A and $\mathcal{L}ab(B) = \texttt{in}\}) + 1$ as the MIN+1 value of the `in` labelled attackers of A. Also, we sometimes write $\mathcal{MM}$ instead of $\mathcal{MM}_{\mathcal{L}ab}$ when it is clear what labelling the min-max numbers relate to.

Theorem 1 ([7]). *Every admissible labelling has a unique min-max numbering.*

Definition 5 ([7]). *A* strongly admissible labelling *is an admissible labelling whose min-max numbering yields natural numbers only (so no argument is numbered* ∞*).*

Theorem 2 ([7]). *An argument is labelled* `in` *by at least one strongly admissible labelling iff it is labelled* `in` *by the grounded labelling.*

As an example, consider the argumentation framework shown below, which we refer to as AF_{ex}. Here $\mathcal{L}ab_1 = (\{A, C, E, G\}, \{B, D, H\}, \{F\})$ is an admissible (though not complete) labelling with associated min-max numbering $\mathcal{MM}_{\mathcal{L}ab_1} = \{(A: 1), (B: 2), (C: 3), (D: 4), (E: 5), (G: \infty), (H: \infty)\}$, which implies that $\mathcal{L}ab_1$ is not strongly admissible. Furthermore, $\mathcal{L}ab_2 = (\{A, C, E\}, \{B, D, F\}, \{G, H\})$ is an admissible (and complete) labelling with associated min-max numbering $\mathcal{MM}_{\mathcal{L}ab_2} = \{(A{:}1), (B{:}2), (C{:}3), (D{:}4), (E{:}5), (F{:}2)\}$, which implies that $\mathcal{L}ab_2$ is indeed a strongly admissible labelling.



From Theorem 2, together with the fact that the grounded extension consists of the `in`-labelled arguments of the grounded labelling [9], it follows that to show that an argument is in the grounded extension, it is sufficient to construct a strongly admissible labelling where the argument is labelled `in`.

The following two lemmas about strongly admissible labellings will be used further on in the paper.

Lemma 1. *Let* $\mathcal{L}ab$ *be a strongly admissible labelling of argumentation framework* (Ar, att)*, and let* $A \in Ar$ *such that* $\mathcal{L}ab(A) = \texttt{undec}$ *and for each* $B \in Ar$ *that attacks* A *it holds that* $\mathcal{L}ab(B) = \texttt{out}$*. Let* $\mathcal{L}ab' = (\texttt{in}(\mathcal{L}ab) \cup \{A\}, \texttt{out}(\mathcal{L}ab), \texttt{undec}(\mathcal{L}ab) \setminus \{A\})$*. It holds that* $\mathcal{L}ab'$ *is a strongly admissible labelling.*

Proof. We first observe that $\mathcal{L}ab'$ is a well-defined labelling in the sense that it defines a partition of Ar. We proceed to show that $\mathcal{L}ab'$ is an admissible labelling. Let $C \in \texttt{in}(\mathcal{L}ab')$. Then either $C \in \texttt{in}(\mathcal{L}ab)$ or $C = A$. In the former case, the fact that $\mathcal{L}ab$ is a (strongly) admissible labelling implies that all attackers of C are labelled `out` by $\mathcal{L}ab$, and therefore also labelled `out` by $\mathcal{L}ab'$ (since $\texttt{out}(\mathcal{L}ab') = \texttt{out}(\mathcal{L}ab)$). In the latter case, the fact that all attackers of A are labelled `out` by $\mathcal{L}ab$ implies that all attackers of C ($= A$) are labelled `out` by $\mathcal{L}ab'$. Alternatively, let $C \in \texttt{out}(\mathcal{L}ab')$. Then (since $\texttt{out}(\mathcal{L}ab') = \texttt{out}(\mathcal{L}ab)$) $C \in \texttt{out}(\mathcal{L}ab)$, so from the fact that $\mathcal{L}ab$ is an admissible labelling, it follows that there is an attacker of C that is labelled `in` by $\mathcal{L}ab$. Since $\texttt{in}(\mathcal{L}ab') \supseteq \texttt{in}(\mathcal{L}ab)$, it follows that this attacker is also labelled `in` by $\mathcal{L}ab'$.

The next thing to show is that $\mathcal{L}ab'$ is also a *strongly* admissible labelling. Suppose, towards a contradiction, that this is not the case. Then there exists at least one `in` or `out` labelled (by $\mathcal{L}ab'$) argument that is numbered with ∞. It follows that this argument is either labelled `in` or `out` by $\mathcal{L}ab$ or it is actually A itself. However, even in the latter case, it follows that there exists at least one `in` or `out` labelled (by

$\mathcal{L}ab$) argument that is numbered with ∞ (since from the fact that A is labelled `in` and numbered with ∞, it follows that all its `out` labelled attackers must be numbered with ∞, and A must have at least one `out` labelled attacker, for otherwise A would be numbered with 1). Let $C \in \texttt{in}(\mathcal{L}ab) \cup \texttt{out}(\mathcal{L}ab)$ be an argument that is numbered with ∞ (w.r.t. $\mathcal{L}ab'$) and whose min-max number (w.r.t. $\mathcal{L}ab$) is minimal among all arguments numbered with ∞ w.r.t. $\mathcal{L}ab'$. We distinguish two cases.

- $\mathcal{L}ab(C) = \texttt{in}$. Then, from $\mathcal{L}ab'$ being an admissible labelling, it follows that all attackers of C are labelled `out` by $\mathcal{L}ab'$. However, since all these attackers have lower min-max numbers (w.r.t. $\mathcal{L}ab$), it follows that none of these is numbered with ∞ (w.r.t. $\mathcal{L}ab'$). After all, C has a *minimal* min-max number (w.r.t. $\mathcal{L}ab$) among all arguments numbered with ∞ (w.r.t. $\mathcal{L}ab'$). This means that the MAX+1 value of the attackers of A cannot be ∞ (w.r.t. $\mathcal{L}ab'$). But then A cannot be numbered with ∞ (w.r.t. $\mathcal{L}ab'$). Contradiction.
- $\mathcal{L}ab(C) = \texttt{out}$. As C is numbered with a natural number (w.r.t. $\mathcal{L}ab$) it follows that the MIN+1 value of all its `in` labelled attackers (w.r.t. $\mathcal{L}ab$) is also a natural number. Let D be an `in` labelled attacker of C with minimal min-max number (w.r.t. $\mathcal{L}ab$). It follows that the min-max number of D (w.r.t. $\mathcal{L}ab$) is smaller than that of C. Hence, D cannot be numbered with ∞ w.r.t. $\mathcal{L}ab'$ (recall that C is the lowest numbered argument (w.r.t. $\mathcal{L}ab$ that is numbered with ∞ w.r.t. $\mathcal{L}ab'$). Hence, D has to have a natural min-max number w.r.t. $\mathcal{L}ab'$. But the the MIN+1 value (w.r.t. $\mathcal{L}ab'$) of the attackers of C is a natural number, so D has to be numbered with a natural number (w.r.t. $\mathcal{L}ab'$). Contradiction.

Lemma 2. *Let $\mathcal{L}ab$ be a strongly admissible labelling of argumentation framework (Ar, att), and let $A \in Ar$ such that $\mathcal{L}ab(A) = \texttt{undec}$ and there exists a $B \in Ar$ that attacks A such that $\mathcal{L}ab(B) = \texttt{in}$. Let $\mathcal{L}ab' = (\texttt{in}(\mathcal{L}ab), \texttt{out}(\mathcal{L}ab) \cup \{A\}, \texttt{undec}(\mathcal{L}ab) \setminus \{A\})$. It holds that $\mathcal{L}ab'$ is a strongly admissible labelling.*

Proof. We first observe that $\mathcal{L}ab'$ is a well-defined labelling in the sense that it defines a partition of Ar. We proceed to show that $\mathcal{L}ab'$ is an admissible labelling. Let $C \in \texttt{in}(\mathcal{L}ab')$. Then $C \in \texttt{in}(\mathcal{L}ab)$, so from $\mathcal{L}ab$ being a (strongly) admissible labelling, it follows that all attackers of C are labelled `out` by $\mathcal{L}ab$. From $\texttt{out}(\mathcal{L}ab') \supseteq \texttt{out}(\mathcal{L}ab)$ it then follows that all attackers of C are also labelled `out` by $\mathcal{L}ab'$. Alternatively, let $C \in \texttt{out}(\mathcal{L}ab')$. Then either $C \in \texttt{out}(\mathcal{L}ab)$ or $C = A$. In the former case, from the fact that $\mathcal{L}ab$ is a (strongly) admissible labelling, it follows that C has an attacker that is labelled `in` by $\mathcal{L}ab$, whch then implies (since $\texttt{in}(\mathcal{L}ab') = \texttt{in}(\mathcal{L}ab)$) that the same attacker is also labelled `in` by $\mathcal{L}ab'$. In the latter case ($C = A$) there exists a $B \in Ar$ that attacks A ($= C$) such that $\mathcal{L}ab(B) = \texttt{in}$. From the fact that $\texttt{in}(\mathcal{L}ab') = \texttt{in}(\mathcal{L}ab)$ it then follows that $\mathcal{L}ab'(B) = \texttt{in}$.
The next thing to show is that $\mathcal{L}ab'$ is also a *strongly* admissible labelling. Suppose, towards a contradiction, that this is not the case. Then there exists at least one `in` or `out` labelled argument (by $\mathcal{L}ab'$) that is numbered with ∞. It follows that this argument is either labelled `in` or `out` by $\mathcal{L}ab$ or it is actually A itself. However, even in the latter case, it follows that there exists at least one `in` or `out` labelled (by $\mathcal{L}ab$) argument that is numbered with ∞ (since from the fact that A is labelled `out` and numbered with ∞, it follows that all of its `in` labelled attackers are numbered with ∞, including B). Let $C \in \texttt{in}(\mathcal{L}ab) \cup \texttt{out}(\mathcal{L}ab)$ be an argument that is numbered with ∞ (w.r.t. $\mathcal{L}ab'$) and whose min-max number (w.r.t. $\mathcal{L}ab$) is minimal among all arguments numbered with ∞ w.r.t. $\mathcal{L}ab'$. We distinguish two cases.

- $\mathcal{L}ab(C) = \texttt{in}$. Then, using similar reasoning as in the proof of Lemma 1 (first bullet) we obtain a contradiction.
- $\mathcal{L}ab(C) = \texttt{out}$. Then, using similar reasoning as in the proof of Lemma 1 (first bullet) we obtain a contradiction.

3 The Grounded Discussion Game

The Grounded Discussion Game that we will define in the current section has two players (proponent and opponent) and is based on four different moves, each of which has an argument as a parameter.

$HTB(A)$ ("A has to be the case")
With this move, the proponent claims that argument A has to be labelled `in` by every complete labelling (and hence also has to be labelled `in` by the grounded labelling).

$CB(B)$ ("B can be the case, or at least cannot be ruled out")
With this move, the opponent claims that argument B does not have to be labelled `out` by every complete labelling. That is, the opponent claims there exists at least one complete labelling where B is labelled `in` or `undec`, and that B is therefore not labelled `out` by the grounded labelling.

$CONCEDE(A)$ ("Fair enough, I agree that A has to be the case")
With this move, the opponent indicates that he now agrees with the proponent (who previously did a $HTB(A)$ move) that A has to be the case (labelled `in` by every complete labelling, including the grounded labelling).

$RETRACT(B)$ ("Fair enough, I give up that B can be the case")
With this move, the opponent indicates that he no longer beliefs that argument B can be `in` or `undec`. That is, the opponent acknowledges that B has to be labelled `out` by every complete labelling, including the grounded labelling.

One of the key ideas of the discussion game is that the proponent has burden of proof. He has to establish the acceptance of the main argument. The opponent merely has to cast sufficient doubts. Also, the proponent has to make sure that the discussion does not go around in circles.

The game starts with the proponent uttering a HTB statement. After each HTB statement (either the first one or a subsequent one) the opponent utters a sequence of one or more CB, $CONCEDE$ and $RETRACT$ statements, after which the proponent again utters an HTB statement, etc. In AF_{ex} the discussion could go as follows.

(1) P: $HTB(C)$
(2) O: $CB(B)$
(3) P: $HTB(A)$
(4) O: $CONCEDE(A)$
(5) O: $RETRACT(B)$
(6) O: $CONCEDE(C)$

In the above discussion, C is called *the main argument* (the argument the discussion starts with). The discussion ends with the main argument being conceded by the opponent, so we say that the proponent wins the discussion.

As an example of a discussion that is lost by the opponent, it can be illustrative to examine what happens if, still in AF_{ex}, the proponent claims that B has to be the case.

(1) P: $HTB(B)$
(2) O: $CB(A)$

After the second move, the discussion is terminated, as the proponent cannot move anymore, since A does not have any attackers. This brings us to the precise preconditions of the discussion moves.

$HTB(A)$ This is either the first move, or the previous move was $CB(B)$, where A attacks B, and no $CONCEDE$ or $RETRACT$ move is applicable.

$CB(A)$ A is an attacker of the last $HTB(B)$ statement that is not yet conceded, the directly preceeding move was not a CB statement, argument A has not yet been retracted, and no $CONCEDE$ or $RETRACT$ move is applicable.

$CONCEDE(A)$ There has been a $HTB(A)$ statement in the past, of which every attacker has been retracted, and $CONCEDE(A)$ has not yet been moved.

$RETRACT(A)$ There has been a $CB(A)$ statement in the past, of which there exists an attacker that has been conceded, and $RETRACT(A)$ has not yet been moved.

Apart from the preconditions mentioned above, all four statements also have the additional precondition that no HTB-CB repeats have occurred. That is, there should be no argument for which HTB has been uttered more than once, CB has been uttered more than once, or both HTB and CB have been uttered. In the first and second case, the discussion is going around in circles (which the proponent has to prevent, since he has burden of proof). In the third case, the proponent has been contradicting himself, as his statements are not conflict-free. In each of these three cases, the discussion comes to an end with no move being applicable anymore.

The above conditions are made formal in the following definition.

Definition 6. *Let $AF = (Ar, att)$ be an argumentation framework. A* grounded discussion *is a sequence of discussion moves constructed by applying the following principles.*

BASIS (HTB**)** *If* $A \in Ar$ *then* $[HTB(A)]$ *is a grounded discussion.*

STEP (HTB**)** *If* $[M_1, \ldots, M_n]$ *(*$n \geq 1$*) is a grounded discussion without* HTB*-*CB *repeats,*[2] *and no* $CONCEDE$ *or* $RETRACT$ *move is applicable,*[3] *and* $M_n = CB(A)$ *and* B *is an attacker of* A *then* $[M_1, \ldots, M_n, HTB(B)]$ *is also a grounded discussion.*

STEP (CB**)** *If* $[M_1, \ldots, M_n]$ *(*$n \geq 1$*) is a grounded discussion without* HTB*-*CB *repeats, and no* $CONCEDE$ *or* $RETRACT$ *move is applicable, and* M_n *is not a* CB *move, and there is a move* $M_i = HTB(A)$ *(*$i \in \{1 \ldots n\}$*) such that the discussion does not contain* $CONCEDE(A)$*, and for each move* $M_j = HTB(A')$ *(*$j > i$*) the discussion contains a move* $CONCEDE(A')$*, and* B *is an attacker of* A *such that the discussion does not contain a move* $RETRACT(B)$*, then* $[M_1, \ldots, M_n, CB(B)]$ *is a grounded discussion.*

STEP ($CONCEDE$**)** *If* $[M_1, \ldots, M_n]$ *(*$n \geq 1$*) is a grounded discussion without* HTB*-*CB *repeats, and* $CONCEDE(B)$ *is applicable then* $[M_1, \ldots, M_n, CONCEDE(B)]$ *is a grounded discussion.*

STEP ($RETRACT$**)** *If* $[M_1, \ldots, M_n]$ *(*$n \geq 1$*) is a grounded discussion without* HTB*-*CB *repeats, and* $RETRACT(B)$ *is applicable then* $[M_1, \ldots, M_n, RETRACT(B)]$ *is a grounded discussion.*

It can be observed that the preconditions of the moves are such that a proponent move (HTB) can never be applicable at the same moment as an opponent move (CB, $CONCEDE$ or $RETRACT$). That is, proponent and opponent essentially take turns in which each proponent turn consists of a single HTB statement, and every opponent turn consists of a sequence of $CONCEDE$, $RETRACT$ and CB moves.

Definition 7. *A grounded discussion* $[M_1, M_2, \ldots, M_n]$ *is called* terminated *iff there exists no move* M_{n+1} *such that* $[M_1, M_2, \ldots, M_n, M_{n+1}]$ *is a grounded discussion. A terminated grounded discussion (with* M_1 *being* $HTB(A)$ *for some* $A \in Ar$*) is won by the proponent iff the discussion contains* $CONCEDE(A)$*, otherwise it is won by the opponent.*

To illustrate why the discussion has to be terminated after the occurrence of a HTB-CB repeat, consider the following discussion in AF_{ex}.

(1) P: $HTB(G)$
(2) O: $CB(H)$
(3) P: $HTB(G)$

After the third move, an HTB-CB repeat occurs and the discussion is terminated (opponent wins). Hence, termination after a HTB-CB repeat is necessary to prevent the discussion from going on perpetually.

Theorem 3. *Every discussion will terminate after a finite number of steps.*

Proof. $CONCEDE$ and $RETRACT$ by definition cannot be repeated for the same argument. HTB and CB can be repeated at most once for the same argument (because when this happens the game will terminate). This, together with the fact that the set of arguments is finite (as we only consider finite argumentation frameworks) implies that the number of moves will be finite and therefore the game will terminate.

From the fact that a discussion terminates after an HTB-CB repeat, the following result follows immediately.

Lemma 3. *No discussion can contain a* $CONCEDE$ *and* $RETRACT$ *move for the same argument.*

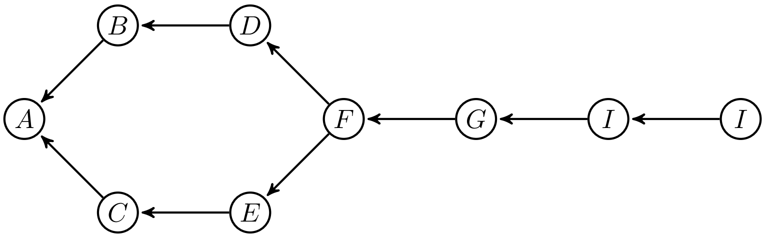
---

[2] We say that there is a HTB-CB repeat iff $\exists i, j \in \{1, \ldots, n\} \exists A \in Ar : (M_i = HTB(A) \vee M_i = CB(A)) \wedge (M_j = HTB(A) \vee M_j = CB(A)) \wedge i \neq j$.

[3] A move $CONCEDE(B)$ is applicable iff the discussion contains a move $HTB(A)$ and for every attacker A of B the discussion contains a move $RETRACT(B)$, and the discussion does not already contain a move $CONCEDE(B)$. A move $RETRACT(B)$ is applicable iff the discussion contains a move $CB(B)$ and there is an attacker A of B such that the discussion contains a move $CONCEDE(A)$, and the discussion does not already contain a move $RETRACT(B)$.

Proof. Suppose, towards a contradiction that there exists a $C \in Ar$ such that both a move $CONCEDE(C)$ and a move $RETRACT(C)$ occurs in the discussion. From the precondition of the $CONCEDE$ move, it follows that the discussion contains the move $HTB(C)$. From the precondition of the $RETRACT$ move, it follows that the discussion contains the move $CB(C)$. But after both the $HTB(C)$ and $CB(C)$ moves have been made, the discussion is terminated, so there is no possibility to do the $CONCEDE(C)$ move (if the $RETRACT(C)$ move was first) or to perform the $RETRACT(C)$ move (if the $CONCEDE(C)$ move was first). Contradiction.

A particular property of the game that is worthwhile emphasizing is that each CB move has to be a reply to the *last* HTB move that is not yet conceded. To illustrate why this is useful, consider the following argumentation framework, which we refer to as AF_{ex2}



Here, the discussion could go as follows.

(01) P: $HTB(A)$
(02) O: $CB(B)$
(03) P: $HTB(D)$
(04) O: $CB(F)$
(05) P: $HTB(G)$
(06) O: $CB(H)$
(07) P: $HTB(I)$
(08) O: $CONCEDE(I)$
(09) O: $RETRACT(H)$
(10) O: $CONCEDE(G)$
(11) O: $RETRACT(F)$
(12) O: $CONCEDE(D)$
(13) O: $RETRACT(B)$
(14) O: $CB(C)$
(15) P: $HTB(E)$
(16) O: $CONCEDE(E)$
(17) O: $RETRACT(C)$
(18) O: $CONCEDE(A)$

Let us consider what would happen when a CB statement is allowed to reply to an *arbitrary* unconceded HTB statement (instead of to the *last* unconceded HTB statement). In that case, at the 6^{th} move, instead of doing $CB(H)$, the opponent could also have done $CB(C)$. In that case, the discussion would have continued as follows.

(06') O: $CB(C)$
(07') P: $HTB(E)$
(08') O: $CB(F)$

Now, there is a HTB-CB repeat ($CB(F)$ at both move (04) and move (08')) so the discussion is terminated. As the main claim is not conceded, the proponent has lost, and no strategy of the proponent could have prevented this. This shows that without the requirement that each CB statement has to reply to the *last* unconceded HTB statement, the proponent could be prevented from winning the game, even though the main argument is in the grounded extension.

3.1 Soundness

Now that the workings of the game have been outlined, and some of its design decisions have been explained, the next step will be to formally proof its correctness and completeness w.r.t. grounded semantics. We start with correctness: if a discussion is won by the proponent, then the main argument is in the grounded extension. In order to prove this, we first have to introduce the notions of the proponent's labelling and the opponent's labelling.

Definition 8. *Let* $[M_1 \dots M_n]$ *be a grounded discussion (in argumentation framework* (Ar, att)*) without any* HTB-CB *repeats.*
The proponent labelling $\mathcal{L}ab_P$ *is defined as*

$\texttt{in}(\mathcal{L}ab_P) = \{A \mid \exists i \in \{1 \dots n\}\colon M_i = \mathit{HTB}(A)\}$
$\texttt{out}(\mathcal{L}ab_P) = \{A \mid \exists i \in \{1 \dots n\}\colon M_i = \mathit{CB}(A)\}$
$\texttt{undec}(\mathcal{L}ab_P) = Ar \setminus (\texttt{in}(\mathcal{L}ab_P) \cup \texttt{out}(\mathcal{L}ab_P))$
The opponent labelling $\mathcal{L}ab_O$ *is defined as*
$\texttt{in}(\mathcal{L}ab_O) = \{A \mid \exists i \in \{1 \dots n\}\colon M_i = \mathit{CONCEDE}(A)\}$
$\texttt{out}(\mathcal{L}ab_O) = \{A \mid \exists i \in \{1 \dots n\}\colon M_i = \mathit{RETRACT}(A)\}$
$\texttt{undec}(\mathcal{L}ab_O) = Ar \setminus (\texttt{in}(\mathcal{L}ab_O) \cup \texttt{out}(\mathcal{L}ab_O))$

Notice that the well-definedness of $\mathcal{L}ab_O$ in Definition 8 does not depend on the absence of HTB-CB repeats (this is due to Lemma 3) whereas the well-definedness of $\mathcal{L}ab_P$ does. When applying $\mathcal{L}ab_O$, we will therefore often do so without having ruled out any HTB-CB repeats, as for instance in the following theorem.

Theorem 4. *Let* $\mathcal{L}ab_O$ *be the opponent's labelling related to discussion* $[M1, \dots, M_n]$. *It holds that* $\mathcal{L}ab_O$ *is strongly admissible.*

Proof. By induction over the number of $\mathit{CONCEDE}$ and $\mathit{RETRACT}$ statements. Let i_1 be the index of the first $\mathit{CONCEDE}$ or $\mathit{RETRACT}$ statement, i_2 be the index of the second $\mathit{CONCEDE}$ or $\mathit{RETRACT}$ statement, etc.

BASIS Suppose the number of $\mathit{CONCEDE}$ and $\mathit{RETRACT}$ statements is zero. In that case, $\mathcal{L}ab_O$ is the all-`undec` labelling, which by definition is strongly admissible.

STEP Suppose that for every discussion with up to j $\mathit{CONCEDE}$ and $\mathit{RETRACT}$ statements, the associated $\mathcal{L}ab_{O_j}$ is strongly admissible. We now prove that also for every discussion with up to $j+1$ $\mathit{CONCEDE}$ and $\mathit{RETRACT}$ statements, the associated $\mathcal{L}ab_{O_{j+1}}$ is strongly admissible. We distinguish two possibilities:

- The last $\mathit{CONCEDE}$ or $\mathit{RETRACT}$ statement was a $\mathit{CONCEDE}$ statement, say, $\mathit{CONCEDE}(B)$ ($B \in Ar$). Let $\mathcal{L}ab_{O_j}$ be the opponent labelling of the sub-discussion $[M_1, \dots, M_{i_{j+1}-1}]$. This discussion contains j $\mathit{CONCEDE}$ and $\mathit{RETRACT}$ statements, so the induction hypothesis says that the associated opponent labelling $\mathcal{L}ab_{O_j}$ is strongly admissible. From the preconditions of $\mathit{CONCEDE}(B)$ it follows that for each attacker $A \in Ar$ of B, the discussion contains the move $\mathit{RETRACT}(A)$. Hence, for each $A \in Ar$ that attacks B, it holds that $A \in \texttt{out}(\mathcal{L}ab_{O_j})$. Also, notice that $\mathcal{L}ab_{O_{j+1}} = (\texttt{in}(\mathcal{L}ab_{O_j}) \cup \{B\}, \texttt{out}(\mathcal{L}ab_{O_j}), \texttt{undec}(\mathcal{L}ab_{O_j}) \setminus \{B\})$. Lemma 1 then implies that $\mathcal{L}ab_{O_{j+1}}$ is strongly admissible.
- The last $\mathit{CONCEDE}$ or $\mathit{RETRACT}$ statement was a $\mathit{RETRACT}$ statement, say, $\mathit{RETRACT}(B)$ ($B \in Ar$). Let $\mathcal{L}ab_{O_j}$ be the opponent labelling of the sub-discussion $[M_1, \dots, M_{i_{j+1}-1}]$. This discussion contains j $\mathit{CONCEDE}$ and $\mathit{RETRACT}$ statements, so the induction hypothesis says that the associated opponent labelling $\mathcal{L}ab_{O_j}$ is strongly admissible. From the preconditions of the $\mathit{RETRACT}(B)$ move, it follows that there is an attacker $A \in Ar$ of B such that the discussion contains the move $\mathit{CONCEDE}(A)$. Hence, $A \in \texttt{in}(\mathcal{L}ab_{O_j})$. Also, notice that $\mathcal{L}ab_{O_{j+1}} = (\texttt{in}(\mathcal{L}ab_{O_j}), \texttt{out}(\mathcal{L}ab_{O_j}) \cup \{B\}, \texttt{undec}(\mathcal{L}ab_{O_j}) \setminus \{B\})$. Lemma 2 then implies that $\mathcal{L}ab_{O_{j+1}}$ is strongly admissible.

Theorem 5. *Let* $[M_1, \dots, M_n]$ *be a terminated grounded discussion that is won by the proponent, and let* $M_1 = \mathit{HTB}(A)$ *for some* $A \in Ar$. *It holds that* A *is in the grounded extension.*

Proof. The fact that the discussion is won by the proponent implies (Definition 7) that there has been a move $\mathit{CONCEDE}(A)$. Hence, $A \in \texttt{in}(\mathcal{L}ab_O)$ (with $\mathcal{L}ab_O$ being the opponent's labelling). Since $\mathcal{L}ab_O$ is strongly admissible (Theorem 4) it follows that A is labelled `in` by the grounded labelling (Theorem 2). Hence, A is in the grounded extension.

As an aside, although it is possible to infer that an argument is in the grounded extension when the proponent wins a discussion (Theorem 5) we cannot infer that an argument is *not* in the grounded extension when the proponent loses a discussion. This is because loss of a game could be due to the proponent following a flawed strategy. For instance, in AF_{ex} one could have the following discussion:

(1) P: $HTB(E)$	(4) O: $CB(H)$
(2) O: $CB(D)$	(5) P: $HTB(G)$
(3) P: $HTB(G)$	

The discussion is terminated at step (5) due to a HTB-CB repeat ($HTB(G)$). The main argument is not conceded, so the proponent loses. Still the proponent could have won by moving $HTB(C)$ instead of $HTB(G)$ at step (3).

3.2 Completeness

Now that the soundness of the game has been proved, we shift our attention to completeness. The obvious thing to prove regarding completeness would be the converse of Theorem 5: if A is in the grounded extension, then there exists a discussion won by the proponent with A as the main argument.[4] However, our aim is to prove a slightly stronger property. Instead of there being just a single discussion won by the proponent, which might be due to the opponent actually providing cooperation during the game, we require the proponent to have a winning strategy. That is, when an argument is in the grounded extension, the proponent will be able to win the game, irrespective of how the opponent choses to play it.

The idea is that the grounded labelling with its associated min-max numbering can serve as a roadmap for winning the discussion. The proponent will be able to win if, whenever he has to do a HTB move, he prefers to use an `in` argument with the lowest min-max number that attacks the directly preceding CB move. We will refer to this as a *lowest number strategy*.[5]

We start by pointing out that using this strategy, the game stays within the bounderies of the grounded labelling (that is, within its `in` and `out` labelled part).

Lemma 4. *If the proponent uses a lowest number strategy, then for every $HTB(A)$ move ($A \in Ar$) it holds that $A \in \mathtt{in}(\mathcal{L}ab_{gr})$ and for every $CB(B)$ move ($B \in Ar$) it holds that $B \in \mathtt{out}(\mathcal{L}ab_{gr})$.*

Proof. This can be proved by induction over the HTB and CB moves in the discussion.

BASIS Let $HTB(A)$ be the first move in the discussion. This means that A is in the grounded extension, so $A \in \mathtt{in}(\mathcal{L}ab_{gr})$.

STEP (CB) Suppose that at a certain stage of the discussion for each $HTB(A)$ move it holds that $A \in \mathtt{in}(\mathcal{L}ab_{gr})$ and for each $CB(B)$ move it holds that $B \in \mathtt{out}(\mathcal{L}ab_{gr})$. If the next move is $CB(C)$ then from the definition of the CB move, it follows that there is a previous $HTB(A)$ move where C attacks A. Our induction hypothesis says that $A \in \mathtt{in}(\mathcal{L}ab_{gr})$. From $\mathcal{L}ab_{gr}$ being an admissible labelling, it follows that each attacker of A (including C) is in $\mathtt{out}(\mathcal{L}ab_{gr})$.

STEP (HTB) Suppose that at a certain stage of the discussion for each $HTB(A)$ move it holds that $A \in \mathtt{in}(\mathcal{L}ab_{gr})$ and for each $CB(B)$ move it holds that $B \in \mathtt{out}(\mathcal{L}ab_{gr})$. If the next move is $HTB(C)$ then from the definition of the HTB move, it follows that there is a previous $CB(B)$ move where C attacks B. Our induction hypothesis says that $B \in \mathtt{out}(\mathcal{L}ab_{gr})$. From $\mathcal{L}ab_{gr}$ being an admissible labelling, it follows that there is at least one attacker of B that is in $\mathtt{in}(\mathcal{L}ab_{gr})$. This means it has been possible for the proponent to follow his strategy of selecting an `in` labelled argument for the HTB move. Hence, $C \in \mathtt{in}(\mathcal{L}ab_{gr})$.

The next thing to be proved is that when the proponent applies a lowest number strategy, the game will not terminate due to any HTB-CB repeats. For this, we first need to prove two lemmas regarding the numbers of the argument moved after a HTB or CB move.

Lemma 5. *If the proponent uses a lowest number strategy, then after an $HTB(A)$ ($A \in Ar$) move is played, all subsequent CB and HTB moves will be related to arguments with lower min-max numbers than A, until a move $CONCEDE(A)$ is played.*

---

[4] A similar strategy is used in [10].

[5] We write "*a* lowest number strategy" instead of "*the* lowest number strategy", as a lowest number strategy might not be unique due to different lowest numbered `in`-labelled arguments being applicable at a specific point. In that case, it suffices to pick an arbitrary one.

Proof. We prove this by induction over the subsequent CB and HTB moves, played in the absence of a $CONCEDE(A)$ move.

BASIS If there are not yet any subsequent CB and HTB moves, then the property trivially holds.

STEP (CB**)** Suppose that at a certain point of the discussion each subsequent CB and HTB move is related to an argument with a lower min-max number than A, and that there has not been any $CONCEDE(A)$ move. Let the next move be $CB(C)$ ($C \in Ar$). From the preconditions of the CB move, it follows that $CB(C)$ responds to the last HTB move that is not yet conceded (say, $HTB(B)$). From the fact that $HTB(A)$ is not yet conceded, it follows that $HTB(B)$ cannot come before $HTB(A)$ (otherwise $CB(C)$ would need to respond to $HTB(A)$ instead of to $HTB(B)$). This leaves just two options: either $HTB(B)$ comes after $HTB(A)$ or $HTB(B) = HTB(A)$. In the former case, the induction hypothesis tells us that $\mathcal{MM}(A) > \mathcal{MM}(B)$. In the latter case, it trivially holds that $\mathcal{MM}(A) = \mathcal{MM}(B)$. So overall, we obtain that $\mathcal{MM}(A) \geq \mathcal{MM}(B)$. As $B \in \mathtt{in}(\mathcal{L}ab_{gr})$ (Lemma 4) it follows that $\mathcal{MM}(B)$ is the MAX+1 value of the (`out` labelled) attackers of B. This implies that B's attacker C has a lower min-max number than B. That is, $\mathcal{MM}(B) > \mathcal{MM}(C)$. This, together with the earlier observed fact that $\mathcal{MM}(A) \geq \mathcal{MM}(B)$ implies that $\mathcal{MM}(A) \geq \mathcal{MM}(B) > \mathcal{MM}(C)$ so $\mathcal{MM}(A) > \mathcal{MM}(C)$, which is precisely what we need to prove.

STEP (HTB**)** Suppose that at a certain point of the discussion each subsequent CB and HTB move is related to an argument with a lower min-max number than A, and that there has not been any $CONCEDE(A)$ move. Let the next move be $HTB(C)$ $C \in Ar$. From the preconditions of the HTB move, it follows that $HTB(C)$ comes directly after a CB move (say, $CB(B)$). From the induction hypothesis, it follows that $\mathcal{MM}(A) > \mathcal{MM}(B)$. Also, it holds that $B \in \mathtt{out}(\mathcal{L}ab_{gr})$ (Lemma 4), so $\mathcal{MM}(B)$ is the MIN+1 value of all its `in` labelled attackers. Since the proponent's strategy is always to play HTB moves for `in` labelled attackers with a minimal min-max number, it follows that $\mathcal{MM}(B) > \mathcal{MM}(C)$. This, together with the earlier observed fact that $\mathcal{MM}(A) > \mathcal{MM}(B)$ implies that $\mathcal{MM}(A) > \mathcal{MM}(B) > \mathcal{MM}(C)$ so $\mathcal{MM}(A) > \mathcal{MM}(C)$, which is precisely what we need to prove.

Lemma 6. *If the proponent uses a lowest number strategy, then after a $CB(A)$ move ($A \in Ar$) is played, all subsequent HTB and CB moves will be related to arguments with lower min-max numbers than A, until a move $RETRACT(A)$ is played.*

Proof. We prove this by induction over the subsequent HTB and CB moves, played in the absence of a $RETRACT(A)$ move.

BASIS If there are not yet any subsequent HTB and CB moves, then the property trivially holds.

STEP (HTB**)** Suppose that at a certain point of the discussion each subsequent HTB and CB move is related to an argument with a lower min-max number than A, and that there has not been any $RETRACT(A)$ move. Let the next move be $HTB(C)$ ($C \in Ar$). From the preconditions of the HTB move, it follows that $HTB(C)$ comes directly after a CB move (say, $CB(B)$). It follows that this $CB(B)$ move cannot come before the $CB(A)$ move (otherwise $HTB(C)$ would have to come before $CB(A)$ as well). This leaves just two options: either $CB(B)$ comes after $CB(A)$ or $CB(B) = CB(A)$. In the former case, the induction hypothesis tells us that $\mathcal{MM}(A) > \mathcal{MM}(B)$. In the latter case, it trivially holds that $\mathcal{MM}(A) = \mathcal{MM}(B)$. So overall, we obtain that $\mathcal{MM}(A) \geq \mathcal{MM}(B)$. As $B \in \mathtt{out}(\mathcal{L}ab_{gr})$ (Lemma 4) it follows that $\mathcal{MM}(B)$ is the MIN+1 value of the `in` labelled attackers of B. Since the proponent's strategy is always to play HTB moves for `in` labelled attackers with a minimal min-max number, it follows that $\mathcal{MM}(B) > \mathcal{MM}(C)$. This, together with the earlier observed fact that $\mathcal{MM}(A) \geq \mathcal{MM}(B)$ implies that $\mathcal{MM}(A) \geq \mathcal{MM}(B) > \mathcal{MM}(C)$ so $\mathcal{MM}(A) > \mathcal{MM}(C)$, which is precisely what we need to prove.

STEP (CB**)** Suppose that at a certain point of the discussion each subsequent HTB and CB move is related to an argument with a lower min-max number than A, and that there has not been any $RETRACT(A)$ move. Let the next move be $CB(C)$ $C \in Ar$). From the preconditions of the CB move, it follows that $CB(C)$ responds to the last HTB move that is not yet conceded (say, $HTB(B)$). From Lemma 5 it then follows that $\mathcal{MM}(B) > \mathcal{MM}(C)$. As for the position of $HTB(B)$ in the discussion, we distinguish two possibilities:

  - $HTB(B)$ comes before $CB(A)$. Let $HTB(Z)$ be the move that $CB(A)$ replies to. $HTB(B)$ cannot come before $HTB(Z)$ because otherwise $HTB(Z)$ (and not $HTB(B)$) would be the last unconceded HTB move at the time $CB(C)$ was played, which is in contraction with $CB(C)$ being a reaction to $HTB(B)$. This leaves just two options: either $HTB(B)$ comes after $HTB(Z)$ or $HTB(B) = HTB(Z)$. In the former case, $HTB(B)$ (and not $HTB(Z)$) would be the last unconceded HTB move at the time $CB(A)$ was played (recall that $HTB(B)$ comes before $CB(A)$), which is in contradiction with $CB(A)$ being a reaction to $HTB(Z)$. In the latter case ($B = Z$) it follows that all HTB moves after $HTB(Z)$ have been conceded, to make $HTB(Z)$ the last unconceded HTB move at the time $CB(C)$ is played. As $CB(A)$ comes after $HTB(Z)$, it follows that also all HTB moves after $CB(A)$ have been conceded (and this includes the HTB move that immediately followed $CB(A)$). But this would mean that $CB(A)$ has to have been retracted. Contradiction. So in both cases, we obtain a contradicion. Hence, the option of $HTB(B)$ coming before $CB(A)$ is not actually possible.
  - $HTB(B)$ comes after $CB(A)$. In case $HTB(B)$ comes *directly* after $CB(A)$, it follows that $\mathcal{MM}(A) > \mathcal{MM}(B)$. This, together with the earlier observed fact that $\mathcal{MM}(B) > \mathcal{MM}(C)$, implies $\mathcal{MM}(A) > \mathcal{MM}(B) > \mathcal{MM}(C)$, so $\mathcal{MM}(A) > \mathcal{MM}(C)$. In case $HTB(B)$ comes *not* directly after $CB(A)$, let $HTB(B')$ be the move directly following $CB(A)$ (the fact that $CB(A)$ is unretracted means that a $RETRACT$ move cannot be the next move, so the next move has to be a HTB move). The fact that $CB(A)$ is unretracted implies that $HTB(B')$ is unconceded. Hence, we can apply the finding of Lemma 5 and obtain that all CB and HTB moves after $HTB(B')$ are related to arguments with lower min-max numbers than B'. This implies $\mathcal{MM}(B') > \mathcal{MM}(C)$. Since $\mathcal{MM}(A) > \mathcal{MM}(B')$ (as $\mathcal{MM}(A)$ is the MIN+1 value of the `in` labelled attackers of A, and B' has a minimal min-max number among the `in` labelled attackers of A, as this conforms with the strategy of the proponent) it follows that $\mathcal{MM}(A) > \mathcal{MM}(B') > \mathcal{MM}(C)$, so $\mathcal{MM}(A) > \mathcal{MM}(C)$. So in both cases, we obtain that $\mathcal{MM}(A) > \mathcal{MM}(C)$, which is precisely what we need the prove.

Lemma 7. *If the proponent uses a lowest number strategy, then no HTB-CB repeats occur.*

Proof. We prove this using three observations.

- The discussion does not contain does not contain an $HTB(A)$ move and a $CB(B)$ move with $A = B$. This follows from the fact that (Lemma 4) for every $HTB(A)$ move it holds that $A \in \mathtt{in}(\mathcal{L}ab_{gr})$ and for every $CB(B)$ move it holds that $B \in \mathtt{out}(\mathcal{L}ab_{gr})$, together with the fact that $\mathtt{in}(\mathcal{L}ab_{gr}) \cap \mathtt{out}(\mathcal{L}ab_{gr}) = \emptyset$.
- The discussion does not contain any repeated $HTB(A)$ moves (for the same argument A).
  Suppose, towards a contradiction, that the discussion *does* contain a repeated $HTB(A)$ move. It can be observed (Lemma 5) that after the first $HTB(A)$ is played, all subsequent HTB moves will be related to arguments with lower min-max numbers than A, until a move $CONCEDE(A)$ is played. A direct consequence of this is that the second $HTB(A)$ move has to be played *after* $CONCEDE(A)$ (as A doesn't have a lower min-max number than itself). From the preconditions of the HTB move, it follows that the second $HTB(A)$ move has to be a reaction to a CB move (say, $CB(B)$ with $B \in Ar$) that directly precedes it. But that means that at the moment the $CB(B)$ move is played, there has already been a $CONCEDE(A)$ move, so the move $RETRACT(B)$ would be applicable immediately afterwards, which is in contradiction with the preconditions of the $HTB(A)$ move.
- The discussion does not contain any repeated $CB(A)$ moves (for the same argument A).
  Suppose, towards a contradiction, that the discussion *does* contain a repeated $CB(A)$ move. It can be observed that after the first $CB(A)$ move has been played, all subsequent CB moves will be related to arguments with lower min-max numbers than A, until a move $RETRACT(A)$ is played (Lemma 6). A direct consequence of this is that the second $CB(A)$ move has to be played *after* $RETRACT(A)$ (as A doesn't have a lower min-max number than itself). But that means that at the moment the second $CB(A)$ move is played, there is already a $RETRACT(A)$ move, which is in contradiction with the preconditions of the CB move.

From the above three observations, it directly follows that the discussion does not contain any HTB-CB repeats.

We are now ready to present the main result regarding completeness of the discussion game.

Theorem 6. *Let A be an argument in the grounded extension of argumentation framework (Ar, att). If the proponent uses a lowest number strategy, he will win the discussion for main argument A.*

Proof. As we have observed before (Theorem 3) every game has to terminate in a finite number of steps. This, by definition, means that at some point, one of the conditions for termination has to hold. Lemma 7 tells us that this cannot be due to any HTB-CB repeats.

We proceed to show that termination also cannot be due to the proponent not being able to react on a CB move. Let $CB(C)$ ($C \in Ar$) be the last move in a particular (possibly unterminated) discussion, and assume that no subsequent $CONCEDE$ or $RETRACT$ move is applicable immediately after it. From Lemma 4 it follows that $C \in \mathtt{out}(\mathcal{L}ab_{gr})$, so from $\mathcal{L}ab_{gr}$ being an admissible labelling, there will be at least one argument that attacks C and is labelled `in` by $\mathcal{L}ab_{gr}$. This, together with the fact that no $CONCEDE$ or $RETRACT$ moves are applicable, and the earlier observed fact that there have been no HTB-CB repeats (Lemma 7) implies that the preconditions for the move $HTB(D)$ are satisfied, where D is an `in` labelled argument with minimal min-max number. Hence, the last move of a terminated discussion cannot be a CB move.

From the thus observed fact that the last move of a terminated discussion cannot be a CB move, it directly follows that the last move has to be a $CONCEDE$, $RETRACT$ or HTB move. Of these moves, HTB is not actually possible, because it can always be followed with a CB or $CONCEDE$ statement (this is due to the fact that an HTB statement cannot be repeated for the same argument). This means the last move has to be $CONCEDE$ or $RETRACT$. The fact that no CB statement is applicable (has its precondition satisfied) then by definition means that for every previous $HTB(C)$ move, either there has been a $CONCEDE(C)$ move, or for every attacker B of C there has been a $RETRACT(B)$ move. Suppose, towards a contradiction that there has been a $HTB(C)$ move ($C \in Ar$) without any subsequent $CONCEDE(C)$ move. It then follows that for every attacker B of C there has been a $RETRACT(B)$ move. But then there exists a next move ($CONCEDE(C)$) so the discussion would not be terminated. Contradiction. Hence, for every move $HTB(C)$ ($C \in Ar$) that has been played in the discussion, an associated $CONCEDE(C)$ move has also been played. Since this includes the main argument (A) it follows that the game is won by the proponent.

As the presence of a winning strategy trivially implies the presence of at least one discussion that is won by the proponent, we immediately obtain the following result.

Corollary 1. *Let A be an argument in the grounded extension of argumentation framework (Ar, att). There exists at least one terminated grounded discussion, won by the proponent, for main argument A.*

3.3 Efficiency (Communication)

Now that soundness and completeness of the game have been shown, we proceed to examine its efficiency. Theorem 3 states that every discussion will terminate, and we are interested in how many steps are required for this. For this, we need the following lemma.

Lemma 8. *Let A be an argument in the grounded extension of argumentation framework (Ar, att). When the proponent uses a lowest number strategy for the discussion of A, then once the game is terminated it holds that $\mathcal{L}ab_O = \mathcal{L}ab_P$.*

Proof. We prove this by showing the following points:

$\mathtt{in}(\mathcal{L}ab_O) \subseteq \mathtt{in}(\mathcal{L}ab(P)$ Let $A \in \mathtt{in}(\mathcal{L}ab_O)$. This means the discussion contains a move $CONCEDE(A)$. From the preconditions of the $CONCEDE$ move it follows that the discussion also contains a move $HTB(A)$. That is, $A \in \mathtt{in}(\mathcal{L}ab_P)$.

$\mathtt{out}(\mathcal{L}ab_O) \subseteq \mathtt{out}(\mathcal{L}ab_P)$ Let $A \in \mathtt{out}(\mathcal{L}ab_O)$. This means the discussion contains a move $RETRACT(A)$. From the preconditions of the $RETRACT$ move it follows that the discussion also contains a move $CB(A)$. That is, $A \in \mathtt{out}(\mathcal{L}ab_P)$.

$\mathtt{in}(\mathcal{L}ab_P) \subseteq \mathtt{in}(\mathcal{L}ab_O)$ Let $A \in \mathtt{in}(\mathcal{L}ab_P)$. This means the discussion contains a move $HTB(A)$. In the proof of Theorem 6 (last paragraph) it was shown that for every HTB in the discussion has been conceded. Hence, the dicussion contains a $CONCEDE(A)$ move. That is $A \in \mathtt{in}(\mathcal{L}ab_O)$.

$\mathtt{out}(\mathcal{L}ab_P) \subseteq \mathtt{out}(\mathcal{L}ab_O)$ Let $A \in \mathtt{out}(\mathcal{L}ab_P)$. This means the discussion contains a move $CB(A)$. Let $HTB(B)$ be the move that $CB(A)$ reacted to. From the previous point, it follows that there also has been a $CONCEDE(B)$ move. But the preconditions of the $CONCEDE$ move require that all attackers (including A) have been retracted. Hence, there has been a $RETRACT(A)$ statement. That is, $A \in \mathtt{out}(\mathcal{L}ab_O)$.

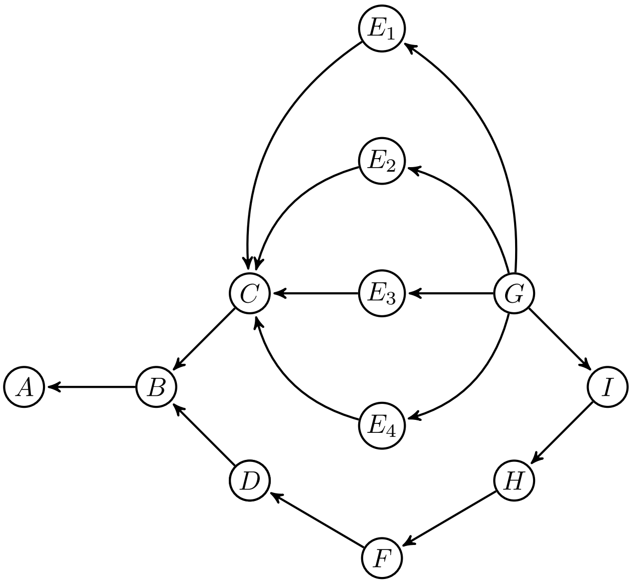
From the first and third point, it follows that $\mathtt{in}(\mathcal{L}ab_O) = \mathtt{in}(\mathcal{L}ab_P)$. From the second and fourth point, it follows that $\mathtt{out}(\mathcal{L}ab_O) = \mathtt{out}(\mathcal{L}ab_P)$. It then follows that also $\mathtt{undec}(\mathcal{L}ab_O) = \mathtt{undec}(\mathcal{L}ab_P)$ (since a labelling essentially defines a partition of Ar). Hence, $\mathcal{L}ab_O = \mathcal{L}ab_P$.

The following theorem states that the discussion game requires a relatively low number of moves.

Theorem 7. *Let A be an argument in the grounded extension of argumentation framework $AF = (Ar, att)$. When the proponent uses a lowest number strategy for A, the resulting terminated disussion will have a number of moves that is linear w.r.t. the size of the strongly admissible labelling that is has been constructed.*

Proof. Let $\mathcal{L}ab_P$ and $\mathcal{L}ab_O$ be the proponent and opponent labelling when the discussion is terminated. For every $B \in \mathtt{in}(\mathcal{L}ab_P)$ there exists precisely one $HTB(B)$ statement in the discussion (because no $HTB(B)$ statement can be repeated, Lemma 7) and for every $B \in \mathtt{out}(\mathcal{L}ab_P)$ there exists precisely one $CB(B)$ statement (because no $CB(B)$ statement can be repeated, Lemma 7). Also, for every $B \in \mathtt{in}(\mathcal{L}ab_O)$ there exists precisely one $CONCEDE(B)$ statement in the discussion (because no $CONCEDE(B)$ can be repeated), and for every $B \in \mathtt{out}(\mathcal{L}ab_O)$ there exists precisely one $RETRACT(B)$ statement in the discussion (because no $RETRACT(B)$ statement can be repeated). This means the total number of moves in the discussion is $|\mathtt{in}(\mathcal{L}ab_P)| + |\mathtt{out}(\mathcal{L}ab_P)| + |\mathtt{in}(\mathcal{L}ab_O)| + |\mathtt{out}(\mathcal{L}ab_O)|$. From the facts that $\mathtt{in}(\mathcal{L}ab_P) \cap \mathtt{out}(\mathcal{L}ab_P) = \emptyset$ and $\mathtt{in}(\mathcal{L}ab_O) \cap \mathtt{out}(\mathcal{L}ab_O) = \emptyset$, it follows that the total number of moves is $|\mathtt{in}(\mathcal{L}ab_P) \cup \mathtt{out}(\mathcal{L}ab_P)| + |\mathtt{in}(\mathcal{L}ab_O) \cup \mathtt{out}(\mathcal{L}ab_O)|$. From the fact that $\mathcal{L}ab_P = \mathcal{L}ab_O$ (Lemma 8) it then follows that the total number of moves is $2 \cdot |\mathtt{in}(\mathcal{L}ab_P) \cup \mathtt{out}(\mathcal{L}ab_P)|$, or equivalently $2 \cdot |\mathtt{in}(\mathcal{L}ab_O) \cup \mathtt{out}(\mathcal{L}ab_O)|$.

As an aside, it can be observed that following a lowest number strategy does not always yield a shortest discussion. As an example, consider the following argumentation framework, which we refer to as AF_{ex3}.



Here, following a lowest number strategy (based on the grounded labelling) can produce the following discussion for main argument A.

(1) P: $HTB(A)$
(2) O: $CB(B)$
(3) P: $HTB(C)$
(4) O: $CB(E_1)$
(5) P: $HTB(G)$
(6) O: $CONCEDE(G)$
(7) O: $RETRACT(E_1)$
(8) O: $CB(E_2)$
(9) O: $RETRACT(E_2)$
(10) O: $CB(E_3)$
(11) O: $RETRACT(E_3)$
(12) O: $CB(E_4)$
(13) O: $RETRACT(E_4)$
(14) O: $CONCEDE(C)$
(15) O: $RETRACT(B)$
(16) O: $CONCEDE(A)$

However, a shorter discussion that is still won by the proponent would be as follows.

(1) P: $HTB(A)$
(2) O: $CB(B)$
(3) P: $HTB(D)$
(4) O: $CB(F)$
(5) P: $HTB(H)$
(6) O: $CB(I)$
(7) P: $HTB(G)$
(8) O: $CONCEDE(G)$
(9) O: $RETRACT(I)$
(10) O: $CONCEDE(H)$
(11) O: $RETRACT(F)$
(12) O: $CONCEDE(D)$
(13) O: $RETRACT(B)$
(14) O: $CONCEDE(A)$

The former discussion yields a strongly admissible labelling $\mathcal{L}ab_1 = (\{G, C, A\}, \{E_1, E_2, E_3, E_4, B\}, \{I, H, F, D\})$ whereas the latter discussion yields a strongly admissible labelling $\mathcal{L}ab_2 = (\{G, H, D, A\}, \{I, F, B\}, \{E1, ..., En, C\})$, with the size of $\mathcal{L}ab_1$ being bigger than the size of $\mathcal{L}ab_2$.

This example illustrates that in order to have a relatively short discussion we have to carefully chose the strongly admissible labelling that is the basis of the lowest number strategy, as $\mathcal{L}ab_2$ will yield a shorter discussion than chosing $\mathcal{L}ab_1$ or the grounded labelling. We conjecture that an “optimal” strongly admissible labelling is one where the main argument is labelled `in` and where the size is minimal.

Conjecture 1. Let $AF = (Ar, att)$ be an argumentation framework and $A \in Ar$. Let $\mathcal{L}ab$ be a strongly admissible labelling that labels A `in` and that has a minimal size among all strongly admissible labellings that label A `in`. When following a smallest number strategy based on $\mathcal{L}ab$, the resulting discussion for main argument A will have minimal length among all discussions for A that are won by the proponent.

3.4 Efficiency (Computation)

As was observed in Section 3.3, the Grounded Discussion Game is linear in the number of moves needed to show grounded membership. As each move consists of a single argument, it is also linear in the total number of arguments moved, hence the “communication complexity” (total amount of information that needs to be communicated) is also linear.

Apart from the burden of communication, there is also the burden of computation. After all, each move has preconditions, and verifying these is not a trivial task. In the current section, we will therefore examine the computational costs of each step in the discussion. To do so, we assume the presence of a number of datastructures.

The first datastructure, called the *AF datastructure*, represents the argumentation framework $AF = (Ar, att)$. It is essentially an array, with an index position for each argument (so argument A_0, gets position 0, argument A_1 gets position 1, etc). Each array position i is the start of two linked lists: one for the arguments in A_i^- and one for the arguments in A_i^+. For argumentation framework AF_{ex} of Section 2 the associated AF datastructure is depicted in Figure 1.

Given a particular strongly admissible labelling, we assume that the AF datastructure is such that for each `out`-labelled argument A, the first element of its A^- linked list will be an `in` labelled attacker with minimal min-max number (among all `in` labelled attackers of A).

Apart from the AF datastructure, there is a second array, which we will refer to as the *flags and counters datastructure*, which for each argument A contains:

- a flag `HTB`$[A]$, which indicates whether the argument has been played in a HTB move. Initially, this flag is `false`.

```
             +-+ +-+ +-+
             |G| |D| |B|
             +-+ +-+ +-+
              ^   ^   ^

     +-+ +-+ +-+ +-+ +-+ +-+ +-+
     |A| |B| |C| |B| |A| |H| |G|
     +-+ +-+ +-+ +-+ +-+ +-+ +-+
      ^   ^   ^   ^   ^   ^   ^

+---+---+---+---+---+---+---+---+  attacked by
| A | B | C | D | E | F | G | H |
+---+---+---+---+---+---+---+---+  attacks
  v   v   v   v           v   v
 +-+ +-+ +-+ +-+         +-+ +-+
 |B| |C| |D| |E|         |D| |G|
 +-+ +-+ +-+ +-+         +-+ +-+
  v   v                   v
 +-+ +-+                 +-+
 |F| |E|                 |H|
 +-+ +-+                 +-+
```

Fig. 1. The AF datastructure of argumentation framework AF_{ex}.

- a flag CB$[A]$, which indicates whether the argument has been played in a CB move. Initially, this flag is `false`.
- a flag CONC$[A]$, which indicates whether the argument has been conceded (played in a $CONCEDE$ move). Initially, this flag is `false`.
- a flag RETR$[A]$, which indicates whether the argument has been retracted (played in a $RETRACT$ move). Initially, this flag is `false`.
- a flag ATT_CONC$[A]$, which indicates whether an attacker has been conceded. Initially, this flag is `false`.
- a non-negative integer NR_ATT$[A]$, which indicates the total number of attackers. It is initialised at $|A^-|$ and never changes.
- a non-negative integer $NR\_ATT\_RETR[A]$, which indicates the number of attackers that have been retracted. Initially, this is set to 0.

For keeping track of the last unconceded HTB statement, we use a stack of arguments, called the HTB stack. The idea is that each time a HTB statement is moved, we push its argument on this stack, and that each time we need the last unconceded HTB statement, we keep on popping the stack until we find an argument that has not been conceded.

The last two datastructures are sets: the *to be conceded set* and the to be retracted set. These, respectively, keep track of the arguments that need to be conceded (because all its attackers have been retracted, and the argument itself has been used in a HTB move but not yet in a $CONCEDE$ move) or retracted (because it has an attacker that has been conceded, and the argument itself has been used in a CB move but not yet in a $RETRACT$ move).

Each time a discussion move is made, the datastructures are updated (except for the AF datastructure, which is never updated). We distinguish four cases:

- The move is $HTB(A)$. In that case, we first check whether a HTB-CB repeat has occurred. That is, do we have HTB$[A]$ or CB$[A]$ in the flags and counters datastructure? If so, the discussion is terminated. If not, set the HTB$[A]$ flag in the flags and counters datastructure, and push A onto the HTB stack. Finally, we need to check whether a $CONCEDE$ move is due: if NR_ATT_RETR$[A]$ $=$ NR_ATT$[A]$ then add A to the to be conceded set.
- The move is $CB(A)$. In that case, we first check whether a HTB-CB repeat has occurred. That is, do we have HTB$[A]$ or CB$[A]$ in the flags and counters datastructure? If so, the discussion is terminated. If not, set the CB$[A]$ flag in the flags and counters datastructure. Finally, we need to check whether a $RETRACT$ is due: if ATT_CONC$[A]$ then add A to the to be retracted set.

- The move is $CONCEDE(A)$. In that case, first set the $\texttt{CONC}[A]$ flag in the flags and counters datastructure. Then, for each argument B in A^+ (accessed by traversing the A^+ linked list in the AF datastructure):
  - Set the $\texttt{ATT\_CONC}[B]$ flag
  - Check if the $\texttt{CB}[B]$ flag is set. If so, add B to the to be retracted set.
- The move is $RETRACT(A)$. In that case, first set the $\texttt{RETR}[A]$ flag in the flags and counters datastructure. Then, for each argument B in A^+ (accessed by traversing the A^+ linked list in the AF datastructure):
  - Increase the $\texttt{NR\_ATT\_RETR}[B]$ by 1.
  - Check if $\texttt{NR\_ATT\_RETR}[B] = \texttt{NR\_ATT}[B]$. If so, add B to the to be conceded set.

Overall, it can be observed that the task of keeping the datastructures up-to-date after a particular move is at most $O(|Ar|)$.

Using the up-to-date datastructures, the proponent and opponent can then select their moves. We distinguish three possibilities:

1. The to be conceded or to be contracted set is not empty. In that case, the opponent has two possible choices:
   (a) Do a $CONCEDE(A)$ move (where A is in the to be conceded set) and subsequently remove A from the to be conceded set.
   (b) Do a $RETRACT(A)$ move (where A is in the to be retracted set) and subsequently remove A from the to be retracted set.
2. The to be conceded set and the to be retracted set are both empty, and the last move was $CB(A)$. In that case, it is the proponent's turn. He has to respond with a $HTB(B)$ move, where B attacks A. Preferrably, in order to win the discussion, this B should be an `in`-labelled argument with a minimal min-max number among all `in`-labelled attackers of A. The proponent finds this argument by examining the AF datastructure. It is the first argument from the A^- linked list.
3. The to be conceded and to be retracted sets are both empty, and the last move is not a CB move. In that case, it is the opponent's turn. Since there is noting to concede or retract, the next move has to be a CB statement. This means the opponent needs to find the last unconceded HTB statement. For this, keep popping the HTB stack until either:
   (a) We obtain an argument B whose $\texttt{CONC}[B]$ flag is `false`. In that case, traverse the B^- linked list in the AF datastructure and select the first C whose $\texttt{RETR}[C]$ flag is `false`. Move $CB(C)$.
   (b) The stack is empty. In that case, there is no unconceded HTB move, so the discussion is terminated.

Overall, the task of using the datastructures for selecting the next move is at most $O(|Ar|^2)$. So the total cost per move is $O(|Ar|) + O(|Ar|^2) = O(|Ar|^2)$. Since the number of moves in the game is linear w.r.t. the size of the strongly admissible labelling, so at most linear to $|Ar|$, the overall algorithmic complexity is $O(|Ar|^3)$, so polynomial. This is in contrast with for instance the Standard Grounded Game, where even if the complexity of playing an individual argument is $O(1)$, the exponential number of arguments makes the overall complexity exponential.

4 Discussion and Related Work

As was shown in Section 3, the Grounded Discussion Game is based on the concept of strong admissibility. In essence, it constructs a strongly admissible labelling where the main argument is labelled `in` (Theorem 4). Moreover, the presence of a strongly admissible labelling provides the proponent with a winning strategy for the game (Theorem 6). These observations make it possible to compare the Grounded Discussion Game with two previously defined games that are also based on strong admissibility: the Standard Grounded Game [25, 4, 19] and the Grounded Persuasion Game [10].

4.1 The Standard Grounded Game

The Standard Grounded Game (SGG) [25, 4, 19] is one of the earliest dialectical proof procedures for grounded semantics. Each game[6] consists of a sequence $[A_1, \ldots, A_n]$ ($n \geq 1$) of arguments, moved by the proponent and opponent taking turns, with the proponent starting. That is, a move A_i ($i \in \{1 \ldots n\}$) is a proponent move iff i is odd, and an opponent move iff i is even. Each move, except the first one, is an attacker of the previous move. In order to ensure termination even in the presence of cycles, the proponent is not allowed to repeat any of his moves. A game is terminated iff no next move is possible; the player making the last move wins.

As an example, in AF_{ex} $[C, B, A]$ is terminated and won by the proponent (as A has no attackers, the opponent cannot move anymore) whereas $[G, H]$ is terminated and won by the opponent (as the only attacker of H is G, which the proponent is not allowed to repeat). It is sometimes possible for the proponent to win a game even if the main argument is not in the grounded extension. An example would be $[F, B, A]$. This illustrates that in order to show that an argument is in the grounded extension, a single game won by the proponent is not sufficient. Instead, what is needed is a *winning strategy*. This is essentially a tree in which each node is associated with an argument such that (1) each path from the root to a leaf constitutes a terminated discussion won by the proponent, (2) the children of each proponent node (a node corresponding with a proponent move) coincide with all attackers of the associated argument, and (3) each opponent node (a node corresponding with an opponent move) has precisely one child, whose argument attacks the argument of the opponent node.

It has been proved that an argument is in the grounded extension iff the proponent has a winning strategy for it in the SGG [25, 3]. Moreover, it has also been shown that an SGG winning strategy defines a strongly admissible labelling, when labelling each argument of a proponent node `in`, each argument of an opponent node `out` and all remaining arguments `undec` [7].

As an example, in AF_{ex} the winning strategy for argument E would be the tree consisting of the two branches $E - B - A$ and $E - D - C - B - A$, thus proving its membership of the grounded extension by yielding the strongly admissible labelling $(\{A, C, E\}, \{B, D\}, \{F, G, H\})$. As can be observed from this example, a winning strategy of the SGG can contain some redundancy when it comes to multiple occurrences of the same arguments in different branches. In the current example, the redundancy is relatively mild (consisting of just the two arguments A and B) but other cases have been found where the SGG requires a number of moves in the winning strategy that is *exponential* w.r.t. the size of the strongly admissible labelling the winning strategy is defining [7, Figure 2].[7] Hence, one of the advantages of our newly defined GDG compared to the SGG is that we go from an exponential [7, Figure 2] to a linear (Theorem 7) number of moves.[8]

4.2 The Grounded Persuasion Game

One of the main aims of the Grounded Persuasion Game (GPG) [10] was to bring the proof procedures of grounded semantics more in line with Mackenzie-style dialogue theory [16, 17] The game has two participants (P and O) and four types of moves: `claim` (the first move in the discussion, with which P utters the main claim that a particular argument has to be labelled `in`), `why` (with which O asks why a particular argument has to be labelled in a particular way), `because` (with which P explains why a particular argument has to be labelled a particular way) and `concede` (with which O indicates agreement with a particular statement of P). During the game, both P and O keep *commitment stores*, partial labellings (which we will refer to as $\mathcal{P}$ and $\mathcal{O}$) which keep track of which arguments they think are `in` and `out` during the course of the discussion. For P, a commitment is added every time he utters a `claim` or `because`

---

[6] What we call an SGG game is called a "line of dispute" in [19].

[7] A similar remark can be made for other tree-based proof procedures, like [12].

[8] As each move contains a single argument, this means the "communication complexity" (the total number of arguments that needs to be communicated) is also linear. This contrasts with the computational complexity of playing the game, which is polynomial ($O(n^3)$, where n is the number of arguments) due to the fact that selecting the next move can have $O(n^2)$ complexity, as was explained in Section 3.4. This is still less than when applying Standard Grounded Game, whose overall complexity would be exponential (even if each move could be selected in just one step) due to the requirement of a winning strategy, which as we have seen can be exponential in size.

statement. For O, a commitment is added every time he utters a `concede` statement. An *open issue* is an argument where only one player has a commitment. Some of the key rules of the Grounded Persuasion Game are as follows (full details in [10]).

- If O utters a `why` $\texttt{in}(A)$ statement (resp. a `why` $\texttt{out}(A)$ statement) then P has to reply with `because` $\texttt{out}(B_1, \ldots, B_n)$ where $B_1, \ldots, B_n$ are all attackers of A (resp. with `because` $\texttt{in}(B)$ where B is an attacker of A).
- Any `why` statement of O has to be related to the most recently created open issue in the discussion.
- A `because` statement is not allowed to use an argument that is already an open issue.
- Once O has enough evidence to agree with P that a particular argument has to be labelled `in` (because for each of its attackers, O is already committed that the attacker is labelled `out`) or has to be labelled `out` (because it has an attacker of which O is already committed that it is labelled `in`), O has to utter the relevant `concede` statement immediately.

Unlike the SGG, in the GPG it is not necessary to construct a winning strategy to show grounded membership. Instead, an argument A is in the grounded extension iff there exists *at least one game* that starts with P uttering "`claim` $\texttt{in}(A)$" and is won by P [10].[9]

As a general property of the Grounded Persuasion Game, it can be observed that at every stage of the discussion, O's commitment store $\mathcal{O}$ is an admissible labelling [10].[10]

As an example, for argument E in AF_{ex} the discussion could go as follows.

	$\texttt{in}(\mathcal{P})$	$\texttt{out}(\mathcal{P})$	$\texttt{in}(\mathcal{O})$	$\texttt{out}(\mathcal{O})$
(1) P: claim $\texttt{in}(E)$	E			
(2) O: why $\texttt{in}(E)$	E			
(3) P: because $\texttt{out}(B, D)$	E	B,D		
(4) O: why $\texttt{out}(B)$	E	B,D		
(5) P: because $\texttt{in}(A)$	E,A	B,D		
(6) O: concede $\texttt{in}(A)$	E,A	B,D	A	
(7) O: concede $\texttt{out}(B)$	E,A	B,D	A	B
(8) O: why $\texttt{out}(D)$	E,A	B,D	A	B
(9) P: because $\texttt{in}(C)$	E,A,C	B,D	A	B
(10)O: concede $\texttt{in}(C)$	E,A,C	B,D	A,C	B
(11)O: concede $\texttt{out}(D)$	E,A,C	B,D	A,C	B,D
(12)O: concede $\texttt{in}(E)$	E,A,C	B,D	A,C,E	B,D

In the above game, the main claim $\texttt{in}(E)$ is conceded so the proponent wins. As was mentioned above, a "because" statement is not allowed to use an argument that is already an open issue. This is to ensure termination even in the presence of cycles. However, this condition has an undesirable side effect. Consider what happens when, at move (4) of the above discussion, the opponent would have decided to utter "why $\texttt{out}(D)$" instead of "why $\texttt{out}(B)$".

(4′) O: why $\texttt{out}(D)$	E	B,D
(5′) P: because $\texttt{in}(C)$	E,C	B,D
(6′) O: why $\texttt{in}(C)$	E,C	B,D

After move (6′) the proponent cannot reply with "because $\texttt{out}(B)$" as $\texttt{out}(B)$ is an open issue, so the game is terminated (according to the rules of [10]) without the main claim being conceded, meaning the proponent loses. Moreover, there is nothing the proponent could have done different in order to win the game, in spite of E being in the grounded extension. One of the advantages of our currently defined Grounded Discussion Game is that such anomalies cannot occur (Theorem 6). Once the proponent utters $HTB(E)$ he can win the game, regardless of whether the opponent responds with $CB(B)$ or with $CB(D)$.

Another difference between the GPG and our currently defined GDG is related to the player who introduces the counterarguments in the discussion. In the GPG this is always the proponent, who for instance explicitly has to list all the attackers against an argument he is actually trying to defend (like "P: because $\texttt{out}(B, A)$" in the above discussion). However, in natural discussion it would be rare for any participant

---

[9] A discussion is won by P iff at the end of the game O is committed that the argument the discussion started with is labelled `in`.

[10] That is, if one regards all arguments where O does not have any commitments to be labelled `undec`.

to provide counterarguments against his own position, other than by mistake. The GDG, however, is such that in a game won by the proponent, each of the counterarguments uttered against proponent's position is uttered by the opponent.

4.3 Summary and Analysis

Overall, the differences between our approach and the other games are summarised in the following table.

	SGG	GPG	GDG
number of moves needed to show strong admissibility	exp [7]	linear [7]	linear (Th. 7)
supports RETRACT and/or CONCEDE moves	no	yes	yes
both propopent and opponent introduce arguments	yes	no	yes
single succesful game implies grounded membership	no	yes	yes
grounded membership implies $\exists$ winning strategy	yes	no	yes

Apart from the technical considerations mentioned above, the research agenda of developing argument-based discussion games is also relevant because it touches some of the foundations of argumentation theory. Whereas for instance classical logic entailment is based on the notion of *truth*, this notion simply does not exist in abstract argumentation and would be problematic even in instantiated argumentation.[11] But if not truth, then what actually is it that is actually yielded by formal argumentation theory? Our view is that argumentation theory yields what can be defended in rational discussion. As our Grounded Discussion Game is essentially a form of persuasion dialogue [28] we have shown that grounded semantics can be seen as a form of persuasion dialogue. Furthermore, Caminada et al. have for instance showed that (credulous) preferred semantics can be seen as a particular form of Socratic dialogue [6, 8]. Hence, different argumentation semantics correspond to different types of discussion [8], an observation that is not just relevant for philosophical reasons, but also opens up opportunities for argument-based human computer interaction. In further research we hope to report on whether engaging in the Grounded Discussion Game increases people's trust in particular forms of argument-based inference. An implementation, that can serve as the basis for this, is currently under development.

References

1. P. Baroni and M. Giacomin. On principle-based evaluation of extension-based argumentation semantics. *Artificial Intelligence*, 171(10-15):675–700, 2007.
2. A. Bondarenko, P.M. Dung, R.A. Kowalski, and F. Toni. An abstract, argumentation-theoretic approach to default reasoning. *Artificial Intelligence*, 93:63–101, 1997.
3. M.W.A. Caminada. Dialogues and hy-arguments. In *Proceedings of the Tenth International Workshop on Non-monotonic Reasoning*, Whistler, Canada, 2004.
4. M.W.A. Caminada. *For the sake of the Argument. Explorations into argument-based reasoning.* Doctoral dissertation Free University Amsterdam, 2004.
5. M.W.A. Caminada. On the issue of reinstatement in argumentation. In M. Fischer, W. van der Hoek, B. Konev, and A. Lisitsa, editors, *Logics in Artificial Intelligence; 10th European Conference, JELIA 2006*, pages 111–123. Springer, 2006. LNAI 4160.
6. M.W.A. Caminada. A formal account of socratic-style argumentation. *Journal of Applied Logic*, 6(1):109–132, 2008.
7. M.W.A. Caminada. Strong admissibility revisited. In Simon Parsons, Nir Oren, Chris Reed, and Frederico Cerutti, editors, *Computational Models of Argument; Proceedings of COMMA 2014*, pages 197–208. IOS Press, 2014.

---

[11] For instance, if a conclusion is considered justified in ASPIC+ [21], does this imply the conclusion is also *true*?

8. M.W.A. Caminada, W. Dvořák, and S. Vesic. Preferred semantics as socratic discussion. *Journal of Logic and Computation*, 2014. (in print).
9. M.W.A. Caminada and D.M. Gabbay. A logical account of formal argumentation. *Studia Logica*, 93(2-3):109–145, 2009. Special issue: new ideas in argumentation theory.
10. M.W.A. Caminada and M. Podlaszewski. Grounded semantics as persuasion dialogue. In Bart Verheij, Stefan Szeider, and Stefan Woltran, editors, *Computational Models of Argument - Proceedings of COMMA 2012*, pages 478–485, 2012.
11. M.W.A. Caminada and Y. Wu. An argument game of stable semantics. *Logic Journal of IGPL*, 17(1):77–90, 2009.
12. P. M. Dung, P. Mancarella, and F. Toni. Computing ideal sceptical argumentation. *Artificial Intelligence*, 171(10-15):642–674, 2007.
13. P.M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77:321–357, 1995.
14. P.M. Dung, R.A. Kowalski, and F. Toni. Assumption-based argumentation. In Guillermo Simari and Iyad Rahwan, editors, *Argumentation in Artificial Intelligence*, pages 199–218. Springer US, 2009.
15. N. Gorogiannis and A. Hunter. Instantiating abstract argumentation with classical logic arguments: Postulates and properties. *Artificial Intelligence*, 175(9-10):1479–1497, 2011.
16. J. D. Mackenzie. Question-begging in non-cumulative systems. *Journal of Philosophical Logic*, 8:117–133, 1979.
17. J. D. Mackenzie. Four dialogue systems. *Studia Logica*, 51:567–583, 1990.
18. S. Modgil. Reasoning about preferences in argumentation frameworks. *Artificial Intelligence*, 173:901–1040, 2009.
19. S. Modgil and M.W.A. Caminada. Proof theories and algorithms for abstract argumentation frameworks. In I. Rahwan and G.R. Simari, editors, *Argumentation in Artificial Intelligence*, pages 105–129. Springer, 2009.
20. S. Modgil and H. Prakken. A general account of argumentation with preferences. *Artificial Intellligence*, 195:361–397, 2013.
21. S. Modgil and H. Prakken. The ASPIC+ framework for structured argumentation: a tutorial. *Argument & Computation*, 5:31–62, 2014. Special Issue: Tutorials on Structured Argumentation.
22. S. Parsons, M. Wooldridge, and L. Amgoud. Properties and complexity of formal inter-agent dialogues. *Journal of Logic and Computation*, 13(3):347–376, 2003.
23. M. Podlaszewski, Y. Wu, and M. Caminada. An implementation of basic argumentation components. In *The 10th International Conference on Autonomous Agents and Multiagent Systems-Volume 3*, pages 1307–1308, 2011.
24. H. Prakken. Coherence and flexibility in dialogue games for argumentation. *Journal of Logic and Computation*, 15(6):1009–1040, 2005.
25. H. Prakken and G. Sartor. Argument-based extended logic programming with defeasible priorities. *Journal of Applied Non-Classical Logics*, 7:25–75, 1997.
26. C. Schulz. Graphical representation of assumption-based argumentation. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, pages 4204–4205, 2015.
27. C. Schulz and F. Toni. Logic programming in assumption-based argumentation revisited - semantics and graphical representation. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, pages 1569–1575, 2015.
28. D. N. Walton and E. C. W. Krabbe. *Commitment in Dialogue: Basic Concepts of Interpersonal Reasoning*. SUNY Series in Logic and Language. State University of New York Press, Albany, NY, USA, 1995.