Winning by Numbers: Connecting Strong Admissibility to Optimal Play in Argumentation*

Shawn Bowers[1], Martin Caminada[2], and Bertram Ludäscher[3]

[1] Gonzaga University bowers@gonzaga.edu
[2] Cardiff University CaminadaM@cardiff.ac.uk
[3] University of Illinois, Urbana-Champaign ludaesch@illinois.edu

Abstract. Strongly admissible labelings and min-max numberings offer well-founded explanations in formal argumentation. We establish a precise correspondence between min-max numberings and remoteness functions from combinatorial game theory, showing that min-max numbers characterize optimal play length, i.e., where players seek the fastest win or longest delay of loss. Our game–argumentation duality strengthens the theoretical and computational foundations for cross-fertilization between argumentation and game theory: game-theoretic provenance explanations apply to argumentation frameworks; pure strategy-based provenance aligns with strongly admissible labelings; and a linear-time algorithm for computing remoteness is sufficient to compute grounded labelings and min-max numbers.

Keywords: Formal Argumentation · Strongly Admissible Labelings · Provenance · Combinatorial Game Theory

1 Introduction

Formal argumentation is a key approach to reasoning with uncertainty. Strong admissibility [1,7] plays a central role for grounded semantics, much like admissibility does for preferred semantics, particularly in proof procedures. To show an argument is in a preferred extension, it suffices to show it is in an admissible set, without constructing the full extension. Similarly, to show that an argument is in the grounded extension it suffices to show it is in a strongly admissible set [7]. This strongly admissible set can then be presented directly or used as the basis for an interactive explanation as a discussion game [6].

Strong admissibility has been defined in several different but equivalent ways [1,7,2]. We focus on its labeling-based form [7], where min-max numberings are central to defining and characterizing strong admissibility. In this paper, we deepen the connection between strong admissibility, min-max numberings, and optimal play in classical game theory, to further clarify the role of min-max numberings via connections to solving and explaining games.

* Supported by the Joint Research and Innovation Seed Grants Program between University of Illinois and Cardiff University under *XAI-CA: Explainable AI via Computational Argumentation.*

Contributions. We establish a formal and precise correspondence between min-max numberings and remoteness functions [21] in combinatorial game theory, showing that min-max numbers are related to optimal play. Using this connection, we apply existing game-based provenance explanations [5,4] to argumentation frameworks. We also develop a new class of provenance based on pure strategies that align with strongly admissible labelings. Our results strengthen the connection between argumentation and game theory, providing a foundation for cross-fertilization between the two fields.

Outline. Section 2 recalls basic definitions in formal argumentation. Section 3 reviews relevant game theory concepts and develops provenance-based approaches for explaining games. Section 4 presents our duality results linking games and argumentation. Section 5 summarizes our contributions and suggests future work.

2 Preliminaries: AF Labelings and Min-Max Numbers

This section briefly recalls key concepts from formal argumentation. We assume finite argumentation frameworks (AFs) and games throughout the paper.

Definition 1 ([13]). An *argumentation framework* $F = (A, R)$ consists of a finite set of entities, called *arguments*, and a binary relation $R \subseteq A \times A$. An edge $(x, y) \in R$ means that x *attacks* y.

A labeling $\mathcal{L}ab : A \to \{\texttt{in}, \texttt{out}, \texttt{undec}\}$ maps arguments to their status under a given semantics where `in` is *accepted*, `out` is *rejected*, and `undec` is *undecided*.

Definition 2 ([7]). $\mathcal{L}ab$ is an *admissible labeling* of F iff for each $x \in A$:

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

$\mathcal{L}ab$ is a *complete labeling* of F iff it is an admissible labeling and for each $x \in A$:

- if $\mathcal{L}ab(x) = \texttt{undec}$ there is a y that attacks x such that $\mathcal{L}ab(y) = \texttt{undec}$, and for each y that attacks x where $\mathcal{L}ab(y) \neq \texttt{undec}$ it holds that $\mathcal{L}ab(y) = \texttt{out}$.

We use $\texttt{in}(\mathcal{L}ab)$ for $\{x \in A \mid \mathcal{L}ab(x) = \texttt{in}\}$, $\texttt{out}(\mathcal{L}ab)$ for $\{x \in A \mid \mathcal{L}ab(x) = \texttt{out}\}$ and $\texttt{undec}(\mathcal{L}ab)$ for $\{x \in A \mid \mathcal{L}ab(x) = \texttt{undec}\}$. We can define partial orders on labelings (similar to subsets of extensions).

Definition 3 ([12]). Let $\mathcal{L}ab$ and $\mathcal{L}ab'$ be labelings of $F = (A, R)$: $\mathcal{L}ab \sqsubseteq \mathcal{L}ab'$ iff $\texttt{in}(\mathcal{L}ab) \subseteq \texttt{in}(\mathcal{L}ab')$ and $\texttt{out}(\mathcal{L}ab) \subseteq \texttt{out}(\mathcal{L}ab')$.

The grounded labeling can be defined as the ($\sqsubseteq$) smallest complete labeling.

Definition 4 ([7]). Let $\mathcal{L}ab$ be a complete labeling of $F = (A, R)$. $\mathcal{L}ab$ is the *grounded labeling* iff $\mathcal{L}ab$ is the (unique) smallest (w.r.t. $\sqsubseteq$) complete labeling.

Strongly admissible labelings can be defined using min-max numberings [7].

Definition 5 ([7]). Let $\mathcal{L}ab$ be an admissible labeling of $F = (A, R)$. A *min-max numbering* is a total function $\mathcal{MM}_{\mathcal{L}ab} : \mathtt{in}(\mathcal{L}ab) \cup \mathtt{out}(\mathcal{L}ab) \to \mathbb{N} \cup \{\infty\}$ such that for each $x \in \mathtt{in}(\mathcal{L}ab) \cup \mathtt{out}(\mathcal{L}ab)$:

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

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

Min-max numbers can be used to define strongly admissible labelings as follows.

Definition 6 ([7]). A *strongly admissible labeling* $\mathcal{L}ab$ is an admissible labeling whose $\mathcal{MM}_{\mathcal{L}ab}$ yields natural numbers only (no argument is numbered ∞).

3 Combinatorial Games: Remoteness and Optimal Play

We recall basic notions and results from combinatorial game theory [17,21,19,20]. A fundamental question addressed is: Who wins under optimal play? We show that solved games represent their own *provenance*, i.e., subgraphs that *explain* objective position values and the length of optimal play.

3.1 Playing Games, Winning Strategies, and Solving Games

Games. A *game* is a finite digraph $G = (V, E)$ consisting of *positions* V and *moves* $E \subseteq V \times V$. To play the game from a starting position $x_0 \in V$, players I and II take turns moving a pebble along the available edges E.

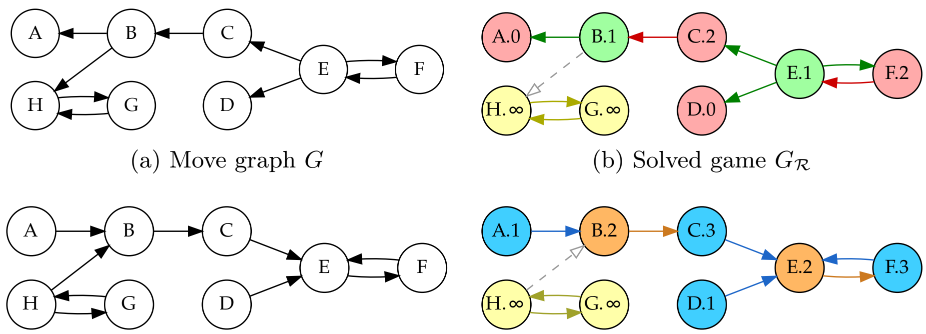
Plays. A *play* π starting at $x_0 \in V$ is a (finite or infinite) sequence of moves:

$$x_0 \xrightarrow{\mathrm{I}} x_1 \xrightarrow{\mathrm{II}} x_2 \xrightarrow{\mathrm{I}} x_3 \xrightarrow{\mathrm{II}} \cdots \qquad (\pi)$$

Player I starts. The *length* $|\pi|$ of a play is the length of the sequence. A play π is *complete* if it either ends after $|\pi| = k$ moves in a *terminal position* (a sink of G), or if $|\pi| = \infty$. The latter means π is a *draw* and the players are forever repeating moves (G is finite, so must have cycles). The player moving to a terminal node *wins*, so the opponent cannot move and *loses*. Players may play optimally, "good enough", or even blunder (e.g., turning a win into a draw or loss). To determine the objective *value* of a position, i.e., under optimal play, we need strategies.

Strategies. A (pure) *strategy* for $G = (V, E)$ is a function $S : V \to V$ such that $(x, S(x)) \in E$. S can be partial (e.g., for terminal positions). For strategy S_{I}, in position x, Player I chooses $S_{\mathrm{I}}(x)$ as the next position if it's I's turn (otherwise II moves according to S_{II}). Any pair S_{I}, S_{II} of strategy functions for I and II defines a unique play $\pi_{S_{\mathrm{I}},S_{\mathrm{II}}}$ from a starting position $x_0 \in V$:

$$x_0 \xrightarrow{\mathrm{I}} \underbrace{S_{\mathrm{I}}(x_0)}_{x_1} \xrightarrow{\mathrm{II}} \underbrace{S_{\mathrm{II}} \circ S_{\mathrm{I}}(x_0)}_{x_2} \xrightarrow{\mathrm{I}} \underbrace{S_{\mathrm{I}} \circ S_{\mathrm{II}} \circ S_{\mathrm{I}}(x_0)}_{x_3} \xrightarrow{\mathrm{II}} \cdots \qquad (\pi_{S_{\mathrm{I}},S_{\mathrm{II}}})$$



(a) Move graph G (b) Solved game $G_{\mathcal{R}}$

(c) Argumentation framework F (d) Grounded $\mathcal{L}ab$ of F with $\mathcal{MM}_{\mathcal{L}ab}$

Fig. 1: (a) Game $G = (V, E)$ and (b) $\mathcal{R}$-labeled solution $G_{\mathcal{R}}$. Node labels "$x.k$" mean $\mathcal{R}(x) = k$ and *optimal play* π_x from x has length $|\pi_x| = k$. $\mathcal{R}$'s parity determines val_G: x is `won` (odd/green), `lost` (even/red), or `drawn` (∞/yellow). (c) AF F is the dual of G. (d) The grounded labeling of F with min-max numbers $\mathcal{MM}_{\mathcal{L}ab}$ and $\mathcal{L}ab$: $x \in Ar$ is either `in` (blue), `out` (orange), or `undec` (yellow).

Position Values. Position $x_0 \in V$ is `won` in $\leq k$ moves if there exists a strategy S_{I} for Player I such that for all strategies S_{II} of II there is an odd number $j < k$ and $S_{\text{I}} \circ (S_{\text{II}} \circ S_{\text{I}})^{\frac{j-1}{2}}(x_0)$ exists, but is not defined for S_{II}. In other words, II cannot move. Such an S_{I} is a *winning strategy*. Conversely, x_0 is `won` for II in $\leq k$ moves if there is a strategy S_{II} such that for all strategies S_{I} there is an even number $j < k$ and $(S_{\text{II}} \circ S_{\text{I}})^{\frac{j}{2}}(x)$ exists, but is not defined for S_{I}: I cannot move!

Note that the objective *value* $\mathsf{val}_G(x_0)$ of position x_0 is *not* determined by an individual play π. Instead, the value of x_0 is `won` (`lost`) if Player I (II) can *force* a win, starting from x_0, no matter how the opponent moves. If neither player can force a win, then x_0 is `drawn` and optimal play is infinite (repeating moves).

Solved Games. Fig. 1b shows the values $\mathsf{val}_G : V \to \{\texttt{won}, \texttt{lost}, \texttt{drawn}\}$ for all $x \in V$ using node colors, i.e., it shows a *solved game*. It is well known that the position values of a solved game satisfy the following two rules:[4]

– $\mathsf{val}_G(x) \coloneqq \texttt{lost}$ if $\forall y$: $(x, y) \in E$ implies $\mathsf{val}_G(y) = \texttt{won}$. $(R_\forall)$
– $\mathsf{val}_G(x) \coloneqq \texttt{won}$ if $\exists y$: $(x, y) \in E$ such that $\mathsf{val}_G(y) = \texttt{lost}$. $(R_\exists)$

3.2 Winning by Numbers: Smith's Remoteness Function $\mathcal{R}$

A classic approach to solve games uses a *remoteness* function due to Steinhaus and Smith [21]. The remoteness $\mathcal{R}$ not only yields position values, but does so by defining for each $x \in V$ the length of optimal play from x.

Let $E^+(x) = \{y \mid (x, y) \in E\}$ denote the *followers* of x in $G = (V, E)$.

[4] Indeed, one way to compute the solution is by iterating these rules, e.g., see [5].

Definition 7 ([21]). The *remoteness* $\mathcal{R} : V \to \mathbb{N} \cup \{\infty\}$ is defined as:

$$\mathcal{R}(x) = \begin{cases} 0 & \text{if } x \text{ has } \textit{no} \text{ followers,} \\ 1 + \min\{\mathcal{R}(y) \mid y \in E^+(x), \mathcal{R}(y) \text{ is } \textit{even}\} & \ldots \text{ has an } \textit{even} \text{ follower,} \\ 1 + \max\{\mathcal{R}(y) \mid y \in E^+(x), \mathcal{R}(y) \text{ is } \textit{odd}\} & \ldots \text{ has } \textit{only odd} \text{ followers,} \\ \infty & \ldots \text{ otherwise.} \end{cases}$$

It is well-known that the parity of $\mathcal{R}$ determines the objective value of a position:

Theorem 2 ($\mathcal{R} \to \mathsf{val}_G$ [21]). For $G = (V, E)$, position $x \in V$ is `won`, `lost`, or `drawn` if and only if $\mathcal{R}(x)$ is *odd*, *even*, or ∞, respectively.

This means "*remoteness is all you need*", i.e., $\mathcal{R}$ yields two connected insights: how long an optimal play from x will last and whether x is won, lost, or drawn.

Remoteness Algorithm. Definition 7 suggests a simple algorithm[5] to compute $\mathcal{R}$, which then can be used to solve for the values of a finite game G and identify optimal play in G: Label all terminal positions x with $\mathcal{R} = 0$. Then label all predecessors y of these x with $\mathcal{R} = 1$. Now delete all such numbered positions x and y from G and repeat after increasing $\mathcal{R}$ by 2, i.e., in the next round, $\mathcal{R}(x)$ will be 2 and 3 (instead of 0 and 1), etc. Repeat until there are no more terminal nodes. The remaining nodes receive $\mathcal{R} = \infty$.

In Figure 2, succ and pred return the *successors* E^+ and *predecessors* E^- of positions, respectively. Lines 2–6 initialize: $\mathcal{R}$-values to ∞; $\mathsf{N}_{\mathsf{succ}}$ to successor counts; T to the terminal nodes; del to false for each node; and the remoteness counter k to 0. Lines 7–20 repeat while there are terminal nodes x to process: in each round, these receive $\mathcal{R} = k$ (meaning *lost in* k), and their predecessors y get $\mathcal{R} = k + 1$ (i.e., *won in* $k + 1$), after which these nodes are deleted. Lines 13–16 compute the new terminal nodes after deletions; k is incremented by 2, and the loop starts over. It is easy to see that $\mathcal{R}$ can be computed in linear time:

Theorem 3. Smith's remoteness function $\mathcal{R}$ can be computed in $\mathcal{O}(|V| + |E|)$.

Proof. Consider the algorithm in Figure 2. *Initialization:* Lines 2, 4, 5 are $\mathcal{O}(|V|)$ and Line 3 is $\mathcal{O}(|E|)$. *Main loop:* Each $x \in V$ can occur in T at most once, then it is deleted; so the loop in Line 7 executes at most $\mathcal{O}(|V|)$ times. *Predecessor processing* (Lines 11–17): When $x \in T$ is processed, each predecessor y corresponds to an edge $(y, x) \in E$, yielding $\mathcal{O}(|E|)$ *total* (i.e., over all loop iterations) for Lines 11, 12, 17. *Successor count updates* (Lines 13–16): For each y, we examine each of its predecessors z and the edge $(z, y) \in E$. Each of these is processed once (and then deleted with y). Lines 14–16 are $\mathcal{O}(1)$ per edge, so no edge is visited more than once in the main loop, resulting in a total cost of $\mathcal{O}(|V| + |E|)$.

Since on connected graphs $|E| \geq |V| - 1$, we have:

Corollary 4. On connected graphs, $\mathcal{R}$ can be computed in $\mathcal{O}(|E|)$.[6]

[5] The authors of [3] attribute the method to von Neumann and Morgenstern [17]

[6] Fraenkel [14] sketches essentially the same algorithm, claiming it is $\mathcal{O}(|E|)$.

Algorithm: Computing `Remoteness` $\mathcal{R}$

Input: Finite game graph $G = (V, E)$
Result: Remoteness function $\mathcal{R} : V \to \mathbb{N} \cup \{\infty\}$

```
begin
    R(x) := ∞ for each x ∈ V                         // drawn unless proven otherwise
    N_succ[x] := |succ(x)| for each x ∈ V             // count successors
    T := {x ∈ V | N_succ[x] = 0}                      // initial terminal nodes
    del[x] := false for each x ∈ V                    // nothing deleted yet
    k := 0                                            // initial remoteness
    while T ≠ ∅ do                                    // until no more terminals
        T_next := ∅                                   // prepare next terminal set
        for x ∈ T do                                  // for each current terminal
            R(x) := k                                 // even R(x) ⇒ x is lost
            for y ∈ pred(x) and ¬del[y] do            // for each predecessor
                R(y) := k + 1                         // odd R(y) ⇒ y is won
                for z ∈ pred(y) and ¬del[z] do        // update predecessor counts:
                    N_succ[z] := N_succ[z] − 1        // ... z loses successor y
                    if N_succ[z] = 0 then             // ... z becomes terminal
                        T_next := T_next ∪ {z}        // ... add to next iteration
                del[y] := true                        // remove y from graph
            del[x] := true                            // remove x from graph
        T := T_next                                   // ready for new terminal set
        k := k + 2                                    // ready for the next two levels
```

Fig. 2: Computing SMITH's remoteness function $\mathcal{R}$ [21] for finite games.

Example 1 ($\mathcal{R} \to \mathsf{val}_G$)**.** Consider the game G in Fig. 1a and its $\mathcal{R}$-labeled, colored solution $G_{\mathcal{R}}$ in Fig. 1b. Positions $\{\mathsf{A}, \mathsf{D}\}$ are terminal ($\mathcal{R} = 0$) and thus immediately lost (red). Positions $\{\mathsf{B}, \mathsf{E}\}$ are predecessors of $\{\mathsf{A}, \mathsf{D}\}$, so they are won (green) with $\mathcal{R} = 1$. After removing these four nodes, $\{\mathsf{C}, \mathsf{F}\}$ become the new terminal (lost) nodes, receiving $\mathcal{R} = 2$. After these have been removed, no more new terminal nodes are created and the algorithm terminates. H and G haven't been reached, so they are drawn (yellow), having infinite remoteness ($\mathcal{R} = \infty$).

Optimal Play. The $\mathcal{R}$-numbers of a solved game $G_{\mathcal{R}}$ allow to find optimal plays and winning strategies easily. Similar to how node colors indicate position values, *edge colors* (Fig. 1b) indicate which moves are *winning* (green), *delaying* a loss (red), or *drawing* (yellow). Another edge type are *blunders* (grey, dashed), e.g., $\mathsf{B} \to \mathsf{H}$: While $\mathsf{B} \to \mathsf{A}$ is a winning move,[7] the move to H blunders the win from B and gives the opponent a *draw* (via an infinite play $\mathsf{H} \rightleftharpoons \mathsf{G}$.) The optimal "countdown play" from E.1 is to D.0; the "count-up" move to C.2 is still winning, but requires a longer play.

[7] $\mathsf{B} \to \mathsf{A}$ is also optimal because it counts down: $\mathcal{R}(\mathsf{A}) = \mathcal{R}(\mathsf{B}) - 1$.

Proposition 1 (Optimal Moves). All non-terminal positions x in $G_{\mathcal{R}}$ have at least one optimal (i.e., *countdown*) move to y, i.e., where $\mathcal{R}(y) = \mathcal{R}(x) - 1$. For drawn x, i.e., $\mathcal{R}(x) = \infty$, some y also has $\mathcal{R}(y) = \infty$ (keeping the draw).

Consider a game G and its $\mathcal{R}$-annotated solution $G_{\mathcal{R}}$. Using the latter, an *optimal play* π from any position $x \in V$ is found simply by following countdown moves.

Definition 8 ($\mathcal{R} \rightarrow$ Optimal Strategies [21]). Given a solved game $G_{\mathcal{R}}$, the strategy $S\colon V \rightarrow V$ is *optimal* if $S(x) = y$ implies $(x, y) \in E$ and $\mathcal{R}(y) = \mathcal{R}(x) - 1$.

If both players follow optimal strategies, they win in the fewest moves possible, delay inevitable defeat as long as possible, and avoid losing from drawn positions. Starting from x, this means that $\mathcal{R}(x)$ bounds the length of optimal play. Winning strategies (and winning moves) don't have to be optimal: e.g., in Fig. 1b, the move $\mathsf{E} \rightarrow \mathsf{C}$ is winning but not optimal.

3.3 Provenance: Explaining Position Values through Subgraphs

The *provenance* $\mathcal{P}(x)$ of $x \in V$ is a subgraph of G that explains x's value (`won`, `lost`, or `drawn`) and possibly its remoteness $\mathcal{R}(x)$. Informally, $\mathcal{P}(x)$ is a subgraph rooted at x that contains some or all of the complete plays from x that are relevant for establishing x's value. We define different types of provenance: *potential*, *actual*, *primary*, and *pure*. Each type provides more specific (i.e., usually smaller) subgraphs that justify x's value (or remoteness).

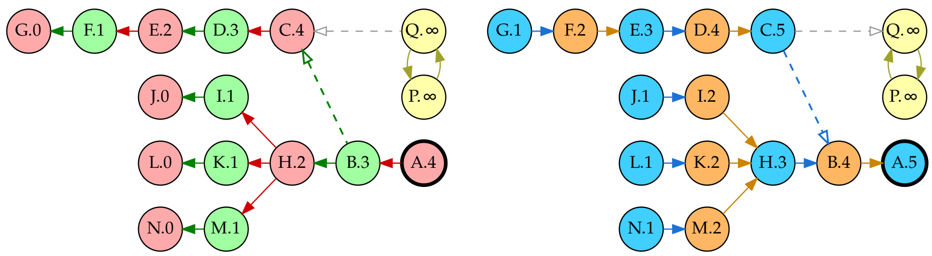
Definition 9 (Potential Provenance). The *potential provenance* $\mathcal{P}_{\mathsf{pt}}(x)$ of a node $x \in V$ is the subgraph of nodes and edges reachable from x in $G = (V, E)$.

$\mathcal{P}_{\mathsf{pt}}(x)$ might overestimate but never underestimate the subgraph needed to justify the value of x. If x is `won`, there exists a move to y that is `lost` for the opponent. However, x may also have moves that are *blunders*, i.e., to some y which is `won` or `drawn` for the opponent. Similarly, if x is `drawn`, it may have a follower y that blunders the draw and allows the opponent to win. *Actual provenance* $\mathcal{P}_{\mathsf{ac}}$ eliminates all blunders, i.e., contains only moves that can be used to determine position values. To this end, we first define edge types.

Definition 10 (Edge Types). Given $G_{\mathcal{R}} = (V, E)$ and position values val_G, the *edge types* $\tau : V \times V \rightarrow \{\texttt{won}, \texttt{lost}, \texttt{drawn}, \texttt{blunder}\}$ are defined by:

$$\tau(x, y) := \begin{cases} \texttt{won} & \text{if } \mathsf{val}_G(x) = \texttt{won} \text{ and } \mathsf{val}_G(y) = \texttt{lost} \\ \texttt{lost} & \text{if } \mathsf{val}_G(x) = \texttt{lost} \text{ and } \mathsf{val}_G(y) = \texttt{won} \\ \texttt{drawn} & \text{if } \mathsf{val}_G(x) = \texttt{drawn} \text{ and } \mathsf{val}_G(y) = \texttt{drawn} \\ \texttt{blunder} & \text{otherwise.} \end{cases}$$

Definition 11 (Actual Provenance). $\mathcal{P}_{\mathsf{ac}}(x)$, the *actual provenance* of x, is the subgraph of G reachable from x by following `won`, `lost`, and `drawn` edges.



(a) A solved game (b) Grounded labeling of the dual AF

Fig. 3: (a) Optimal play *vs.* minimal-size explanation: The primary provenance $\mathcal{P}_{\mathsf{pr}}(\mathsf{A})$ excludes the dashed edge $\mathsf{B}{\rightarrow}\mathsf{C}$ and the subgraph $\mathsf{C} \rightsquigarrow \mathsf{G}$ as only countdown paths are in $\mathcal{P}_{\mathsf{pr}}$. The suboptimal $\mathsf{B}{\rightarrow}\mathsf{C} \rightsquigarrow \mathsf{G}$, however, is a minimal-size explanation of $\mathsf{val}(\mathsf{A})$. (b) The AF-dual of (a): $\{\mathsf{A}, \mathsf{C}, \mathsf{E}, \mathsf{G}\}$ is a *minimal strongly admissible set* witnessing $\mathcal{L}ab(\mathsf{A})$: it uses the subgraph $\mathsf{A}{\leftarrow}\mathsf{B}{\leftarrow}\mathsf{C} \leftsquigarrow \mathsf{G}$.

Since all blunders are removed, the remaining moves in $\mathcal{P}_{\mathsf{ac}}(x)$ are "good enough" to achieve the best outcome $\mathsf{val}(x)$, independent of what the opponent does. However, by playing suboptimal moves, a *mixed* strategy[8] may be required to win: In Fig. 1b, although the move $\mathsf{E} \rightarrow \mathsf{F}$ *is* winning, it requires the player to break out of the cycle *eventually* and move from E to either C or D to force a win. This is only possible with a mixed strategy, which allows a player to take one of many possible moves from a given position, but not with a pure strategy.

One solution to this problem is to exclude suboptimal winning moves from consideration. This idea gives rise to the notion of *primary provenance*. In the resulting subgraphs pure strategies are sufficient to explain position values.

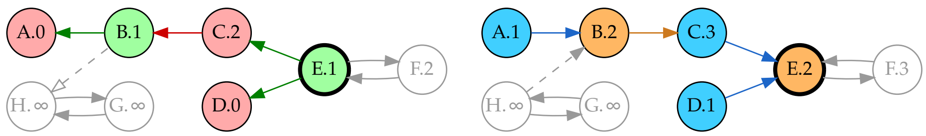
Definition 12 (Primary Provenance). $\mathcal{P}_{\mathsf{pr}}(x)$ is the subset of $\mathcal{P}_{\mathsf{ac}}(x)$ that excludes suboptimal (i.e., non-countdown) winning moves.

Example 2. In Fig. 1b $\mathcal{P}_{\mathsf{pr}}(\mathsf{E})$ excludes both the "detour" through F, which was included in $\mathcal{P}_{\mathsf{ac}}(\mathsf{E})$, and the suboptimal path $\mathsf{E}{\rightarrow}\mathsf{C} \rightsquigarrow A$.

$\mathcal{P}_{\mathsf{pr}}$ only follows optimal winning moves, but includes suboptimal (not maximal) delaying moves. The rationale for this choice is that *all* followers of a losing position x must be explored to establish x as `lost`. Primary provenance avoids "detours" (e.g., through F above) and bases explanations on pure strategies. Finally, like actual provenance, $\mathcal{P}_{\mathsf{pr}}$ is easily computed using $\mathcal{R}$.

Optimal *vs.* Minimal Explanations. If we are interested in minimal-size explanations, by design, $\mathcal{P}_{\mathsf{pr}}$ may exclude smaller explanations (subgraphs) that are not countdown-optimal: e.g., in Fig. 3a, the primary provenance $\mathcal{P}_{\mathsf{pr}}(\mathsf{A})$ excludes the dashed edge $\mathsf{B}{\rightarrow}\mathsf{C}$ (it's not a countdown edge), and thus the subgraph $\mathsf{C} \rightsquigarrow \mathsf{G}$, as only countdown paths are in $\mathcal{P}_{\mathsf{pr}}$. However, the subgraph $\mathsf{B}{\rightarrow}\mathsf{C} \rightsquigarrow \mathsf{G}$ is

[8] In that case, S is a *relation* and not necessarily a function.



(a) The pure provenance of E (b) AF strongly-admissible set from E

Fig. 4: (a) The pure provenance $\mathcal{P}_{\mathsf{pu}}(\mathsf{E})$ of E (highlighted) does not include F, since E→F is not selected by any pure winning strategies (otherwise, it would result in infinite play from E), however, both the optimal move E→D (as in primary provenance) and the suboptimal move E→C (unlike in primary provenance) are included. (b) The corresponding explanation of E in the dual AF, where $\{\mathsf{A}, \mathsf{C}\}$, $\{\mathsf{D}\}$, and $\{\mathsf{A}, \mathsf{C}, \mathsf{D}\}$ are corresponding strongly admissible sets.

a size-minimal explanation of $\mathsf{val}(\mathsf{A})$. Thus, $\mathcal{P}_{\mathsf{pr}}$ can be too selective to include all minimal explanations. The actual provenance $\mathcal{P}_{\mathsf{ac}}(\mathsf{A})$ does include the size-minimal subgraph ($\mathsf{B} \rightsquigarrow \mathsf{G}$), but unfortunately also includes the unfounded loop that primary provenance was meant to eliminate. What is needed is a new form of provenance that lies between actual and primary provenance.

Definition 13 (Pure-Strategy Provenance). $\mathcal{P}_{\mathsf{pu}}(x)$ is the subset of $\mathcal{P}_{\mathsf{ac}}(x)$ that excludes all y that cannot be reached from x via a pure (winning) strategy.

Example 3. Fig. 4a depicts the pure provenance $\mathcal{P}_{\mathsf{pu}}(\mathsf{E})$ of E, which includes the subgraphs $\mathsf{E} \rightarrow \mathsf{C} \rightsquigarrow A$ and $\mathsf{E} \rightarrow \mathsf{D}$. Unlike with actual provenance, F is not included in $\mathcal{P}_{\mathsf{pu}}(\mathsf{F})$: no pure (winning) strategy can include E→F as it would result in infinite play from E (where a pure strategy allows only one move from a given position). The suboptimal move E→C is included in $\mathcal{P}_{\mathsf{pu}}(\mathsf{E})$ (and its associated subgraph) unlike with $\mathcal{P}_{\mathsf{pr}}(\mathsf{E})$.

Proposition 2. Let $G_{\mathcal{R}}$ be a solved game. For all positions $x \in V$:

$$\mathcal{P}_{\mathsf{pt}}(x) \supseteq \mathcal{P}_{\mathsf{ac}}(x) \supseteq \mathcal{P}_{\mathsf{pu}}(x) \supseteq \mathcal{P}_{\mathsf{pr}}(x)$$

This hierarchy allows users to employ the most suitable notion of provenance for their use cases. The potential provenance is easy to compute since it reduces to a simple reachability query. Similarly, actual and primary provenance are easily computed via $\mathcal{R}$ and regular path queries [5]. The pure provenance $\mathcal{P}_{\mathsf{pu}}(x)$, on the other hand, cannot be computed based on $\mathcal{R}$ alone.

4 Game–Argumentation Duality

It has been shown that grounded labelings of argumentation frameworks and solutions of games (computed via the well-founded semantics [15]) directly correspond to one another [4]. We revisit and expand this *Game–AF duality* here, as it allows us to transfer notions and results from one community to another.

4.1 Argumentation Frameworks as Combinatorial Games

To view an argumentation framework F as a game G_F (G for short), we reverse its attack edges, i.e., use the *attacked-by* relation.

Definition 14 (Dual Game). Let $F = (A, R)$ be an AF. The *dual game* $G = (A, R^{-1})$ of F has the same nodes, but *reversed* edges, i.e., the moves of G are the *attacked-by* relation: $R^{-1} = \{(y, x) \mid (x, y) \in R\}$.

Example 4. The game in Fig. 1a and the AF in Fig. 1c are dual to each other. They only differ in the interpretation of nodes (*positions vs. arguments*) and edges (*moves vs. attacks*). The duality carries over to the solved game $G_{\mathcal{R}}$ in Fig. 1b and its dual, the grounded labeling $\mathcal{MM}_{\mathcal{L}ab}$ in Fig. 1d: Positions that are `won` (green), `lost` (red), and `drawn` (yellow) correspond to arguments that are `out` (orange), `in` (blue), and `undec` (yellow), respectively. Positions have a *remoteness* $\mathcal{R}$, while arguments have similar *min-max numbers* from $\mathcal{MM}_{\mathcal{L}ab}$.

A Skeptic's Argumentation Game (SAG [4]). Consider argument E in Fig. 1c and 1d. To show that $x = \mathsf{E}$ is defeated (`out`), it suffices to find an attacker $y \in \{\mathsf{C}, \mathsf{D}, \mathsf{F}\}$ that is accepted (`in`). As it turns out (see below), this is equivalent to moving from x to a follower $y \in \{\mathsf{C}, \mathsf{D}, \mathsf{F}\}$ which is `lost`. More generally, if a player makes the move $x \to y$ in G, the intent is to demonstrate that x is `won` by selecting a y that is `lost` for the opponent. If, however, all moves from x end in a position that is `won` by the opponent, then x itself is `lost`. In the dual AF, this means that to show that x is `out`, one must find an attacker y that is `in`. If, however, all attackers y of x are `out`, then x itself is `in`. The first duality between G and F, illustrated by Fig. 1, is captured by the following theorem.

Theorem 5 (Duality $\mathcal{L}ab \cong \mathsf{val}$). Let $F = (A, R)$ be an AF, $\mathcal{L}ab$ its grounded labeling, and $G_{\mathcal{R}} = (A, R^{-1})$ the solved dual game. For all $x \in A$:

$$\mathcal{L}ab(x) = \texttt{in}/\texttt{out}/\texttt{undec} \text{ iff } \mathsf{val}_G(x) = \texttt{lost}/\texttt{won}/\texttt{drawn}, \text{ respectively.}$$

Proof. It is well-known [13] that the following rules, under the well-founded semantics (WFS) [15], compute the grounded solutions of AFs.

$$\begin{aligned} \texttt{out}(x) &\leftarrow \texttt{attacks}(y, x),\ \texttt{in}(y). \\ \texttt{in}(x) &\leftarrow \neg\, \texttt{out}(x). \end{aligned} \qquad (P_{AF})$$

The following are equivalent under the reversed "*attacked-by*" direction of edges and thus also compute the grounded solutions.

$$\begin{aligned} \texttt{out}(x) &\leftarrow \texttt{attackedBy}(x, y),\ \texttt{in}(y). \\ \texttt{in}(x) &\leftarrow \neg\, \texttt{out}(x). \end{aligned} \qquad (P_{AF}^{-1})$$

It is also well-known that the WFS of the following program solves games [15].

$$\begin{aligned} \texttt{won}(x) &\leftarrow \texttt{move}(x, y),\ \texttt{lost}(y). \\ \texttt{lost}(x) &\leftarrow \neg\, \texttt{won}(x). \end{aligned} \qquad (P_G)$$

Since P_{AF}^{-1} and P_G are the same program (up to renaming), they have the same well-founded models (up to renaming/interpretation). Note that if x is `out` in the grounded labeling, a skeptic making the claim that x is defeated has a winning strategy, hence we call this the *Skeptics Argumentation Game* (SAG) [4].

Example 4 (*Continued*). Consider again the solved game in Fig. 1b and the grounded AF labeling in Fig. 1d. As in Theorem 5, each `won` (green) position in the game is `out` (orange) in the AF, each `lost` position is `in` (blue), and each `drawn` position (yellow) is `undec` (also yellow).

4.2 Remoteness *vs.* Min-Max Numbers

An argumentation framework F and its dual G each have an associated numbering, i.e., *min-max numbers* $\mathcal{MM}_{\mathcal{L}ab}$ for the grounded labeling of F and remoteness $\mathcal{R}$ for G, respectively. Fig. 1 shows that these two numberings differ by 1. Another difference is that $\mathcal{R}$-values are derived directly from G, while min-max numbers are defined for (strongly) admissible labelings.

Theorem 6 (Duality $\mathcal{MM} \cong \mathcal{R}+1$, Grounded $\mathcal{L}ab$). Let $F = (A, R)$ be an AF, $\mathcal{L}ab$ be the grounded labeling of F, $\mathcal{MM}_{\mathcal{L}ab}$ its min-max numbering, and $G_{\mathcal{R}} = (A, R^{-1})$ be the solved dual of F. For each $x \in A$:

- If $\mathcal{L}ab(x) \in \{\texttt{in}, \texttt{out}\}$ then $\mathcal{MM}_{\mathcal{L}ab}(x) = \mathcal{R}(x) + 1$;
- If $\mathcal{L}ab(x) = \texttt{undec}$ then $\mathcal{MM}_{\mathcal{L}ab}(x) = \bot$ (undefined) and $\mathcal{R}(x) = \infty$.

Proof. By induction using Def. 5 and 7.

Base Case: If x is unattacked, $\mathcal{L}ab(x) = \texttt{in}$, $\mathcal{MM}_{\mathcal{L}ab}(x) = 1 + \max(\emptyset) = 1$, and $\mathcal{R}(x) = 0$ (since x is terminal in G), thus $\mathcal{MM}_{\mathcal{L}ab}(x) = \mathcal{R}(x) + 1$.

Rejected Case: Suppose $\mathcal{L}ab(x)$ is `out` and x has `in`-labeled attackers $y_1, \ldots, y_n$, then $\mathcal{MM}_{\mathcal{L}ab}(x) = \min(\{\mathcal{MM}_{\mathcal{L}ab}(y_1), \ldots, \mathcal{MM}_{\mathcal{L}ab}(y_n)\}) + 1$. Assume $\mathcal{MM}_{\mathcal{L}ab}(y_i) = \mathcal{R}(y_i) + 1$ for $1 \leq i \leq n$. Because $\mathcal{L}ab(y_i)$ is `in`, $\mathsf{val}(y_i)$ is `lost` and $\mathcal{R}(y_i)$ is even, thus $\mathcal{R}(x) = 1 + \min(\{\mathcal{R}(y_1), \ldots, \mathcal{R}(y_n)\})$. If $\mathcal{MM}_{\mathcal{L}ab}(y_k)$ has the smallest number of $y_1, \ldots, y_n$, then $\mathcal{MM}_{\mathcal{L}ab}(x) = \mathcal{MM}_{\mathcal{L}ab}(y_k) + 1$, $\mathcal{R}(x) = \mathcal{R}(y_k) + 1$, and since $\mathcal{MM}_{\mathcal{L}ab}(y_k) = \mathcal{R}(y_k) + 1$, $\mathcal{MM}_{\mathcal{L}ab}(x) = \mathcal{R}(x) + 1$.

Accepted Case: Suppose $\mathcal{L}ab(x)$ is `in` with attackers $y_1, \ldots, y_n$, which must be `out`, and $\mathcal{MM}_{\mathcal{L}ab}(x) = \max(\mathcal{MM}_{\mathcal{L}ab}(y_1), \ldots, \mathcal{MM}_{\mathcal{L}ab}(y_n)\} + 1$. Assume $\mathcal{MM}_{\mathcal{L}ab}(y_i) = \mathcal{R}(y_i) + 1$ for $1 \leq i \leq n$. Because $\mathcal{L}ab(y_i)$ is `out`, $\mathsf{val}(y_i)$ is `won` and $\mathcal{R}(y_i)$ is odd, thus $\mathcal{R}(x) = 1 + \max(\{R(y_1), \ldots, \mathcal{R}(y_n)\})$. If $\mathcal{MM}_{\mathcal{L}ab}(y_k)$ has the largest min-max number of $y_1, \ldots, y_n$, then $\mathcal{MM}_{\mathcal{L}ab}(x) = \mathcal{MM}_{\mathcal{L}ab}(y_k)+1$, $\mathcal{R}(x) = \mathcal{R}(y_k)+1$, and since $\mathcal{MM}_{\mathcal{L}ab}(y_k) = \mathcal{R}(y_k) + 1$, $\mathcal{MM}_{\mathcal{L}ab}(x) = \mathcal{R}(x) + 1$.

Undecided Case: If $\mathcal{L}ab(x)$ is `undec`, then by definition $\mathcal{MM}_{\mathcal{L}ab}(x) = \bot$, and since $\mathsf{val}(x)$ is `drawn`, $\mathcal{R}(x) = \infty$.

Example 4 (*Continued*). As shown in the solved game in Fig. 1b and the grounded solution of the dual AF in Fig. 1d, remoteness and min-max numbers differ by 1 when the values are natural numbers. This "off-by-1" nature of remoteness and min-max numbers follows from Theorem 6.

The $\mathcal{MM} \cong \mathcal{R}+1$ correspondence also extends to admissible labelings. Let $G|_W = (W, E \cap (W \times W))$ denote the restriction of G to a set of nodes $W \subseteq V$.

Theorem 7 (Duality $\mathcal{MM} \cong \mathcal{R}$, Admissible $\mathcal{L}ab$). For $F = (A, R)$, its admissible labeling $\mathcal{L}ab$ with $W = \mathtt{in}(\mathcal{L}ab) \cup \mathtt{out}(\mathcal{L}ab)$, $\mathcal{MM}_{\mathcal{L}ab}$ the min-max numbering, $G = (A, R^{-1})$, and $\mathcal{R}_{G|_W}$ the remoteness function on $G|_W$. Then for all $x \in A$:

- If $\mathcal{MM}_{\mathcal{L}ab}(x) \neq \bot$ then $\mathcal{MM}_{\mathcal{L}ab}(x) = \mathcal{R}_{G|_W}(x) + 1$.

Proof. $\mathcal{MM}_{\mathcal{L}ab}$ is a unique numbering of F that only examines arguments labeled `in` or `out` by $\mathcal{L}ab$: for any x whose $\mathcal{L}ab(x)$ is `undec`, $\mathcal{MM}_{\mathcal{L}ab}(x) = \bot$. It follows that $\mathcal{MM}_{\mathcal{L}ab}$ returns the same numbers for $F|_W$ as for F. From Theorem 6, when x is `in` or `out` in the grounded labeling of $F|_W$, $\mathcal{MM}_{\mathcal{L}ab}(x) = \mathcal{R}_{G|_W}(x)+1$. For those arguments x that are labeled `undec` in the grounded labeling of $F|_W$, $R_{G|_W}(x) = \infty$. Thus, it is enough to show that these same arguments have $\mathcal{MM}_{\mathcal{L}ab}(x) = \infty$. Note that such an x must have at least one move to a `drawn` position (`undec` attacker) and no moves to `lost` positions (`in` arguments) in $G|_W$ ($F|_W$, resp.). There are two cases to consider for such an argument x, which we show by contradiction: (1) If $\mathcal{L}ab(x)$ is `out` and $\mathcal{MM}_{\mathcal{L}ab}(x) \neq \infty$, x must have an `in`-labeled attacker y such that $\mathcal{MM}_{\mathcal{L}ab}(y) \neq \infty$. However, such a y implies $\mathcal{R}_{G|_W}(y) \neq \infty$ which means y cannot be `drawn`. (2) If $\mathcal{L}ab(x)$ is `in` and $\mathcal{MM}_{\mathcal{L}ab}(x) \neq \infty$, then all attackers y must have $\mathcal{MM}_{\mathcal{L}ab}(y) \neq \infty$. This means each such y cannot be `drawn` since $R_{G|_W}(y) \neq \infty$, and so $R_{G|_W}(x) \neq \infty$ implying x cannot be `drawn`.

The extension to admissible labelings is a direct consequence of the fact that, like the remoteness function, $\mathcal{MM}_{\mathcal{L}ab}$ computes the grounded solution of an AF restricted to the `in`/`out`-labeled arguments of $\mathcal{L}ab$.

Corollary 8 (Parity of $\mathcal{MM}$). Let $F = (A, R)$ be an AF, $\mathcal{L}ab_1$ an admissible labeling of F with $W = \mathtt{in}(\mathcal{L}ab_1) \cup \mathtt{out}(\mathcal{L}ab_1)$, $\mathcal{MM}_{\mathcal{L}ab_1}$ its min-max numbering, and $\mathcal{L}ab_2$ the grounded labeling of $F|_W$. For each $x \in A$:

- $\mathcal{MM}_{\mathcal{L}ab_1}(x)$ is *odd*/*even*/∞ iff $\mathcal{L}ab_2(x) = \mathtt{in}/\mathtt{out}/\mathtt{undec}$, respectively.

Given the connection between min-max numberings and remoteness, min-max numbers can be viewed as lengths given by optimal play. The following is immediate from Theorem 7.

Corollary 9 ($\mathcal{MM}$ *vs.* Optimal Play). Let $F = (A, R)$ be an AF, $\mathcal{L}ab$ be an admissible labeling of F with $W = \mathcal{L}ab(\mathtt{in}) \cup \mathcal{L}ab(\mathtt{out})$, and $\mathcal{MM}_{\mathcal{L}ab}$ its min-max numbering. If $\mathcal{MM}_{\mathcal{L}ab}(x) = n$, then the length of optimal play from x in the dual game $G|_W$ is $n-1$, for all $x \in A$.

As a consequence of Theorems 2–6, the grounded labeling $\mathcal{L}ab$ and its min-max numbering $\mathcal{MM}_{\mathcal{L}ab}$ can be computed in linear time:

Corollary 10 (Computing Grounded $\mathcal{L}ab$). Let $F = (A, R)$ be an AF. The grounded labeling $\mathcal{L}ab$ of F can be computed in $\mathcal{O}(|A| + |R|)$.

Corollary 11 (Computing $\mathcal{MM}_{\mathcal{L}ab}$). Let $F = (A, R)$ be an AF and $\mathcal{L}ab$ its grounded labeling. $\mathcal{MM}_{\mathcal{L}ab}$ can be computed in $\mathcal{O}(|A| + |R|)$.

4.3 Strong Admissibility and Games

Close connections exist between strongly admissible labelings (as a form of explanation) and game provenance for explaining position values. As an example, min-max numberings can be used to check if a labeling is strongly admissible (Definition 6), and in a similar way, remoteness can be used to check if a subgraph of G corresponds to an admissible labeling.

Definition 15 (Admissible Subgraph). Let G be a game graph and val_G be a (potentially partial) `won-lost` labeling that satisfies the rules $R_\forall$ and $R_\exists$ (Section 3.1). G' is an *admissible subgraph* of G if it is an induced subgraph containing exactly the positions labeled as `won` or `lost` in val_G.

The following is immediate from Definition 6 and the duality of $\mathcal{MM}$ and $\mathcal{R}$.

Corollary 12 (Strongly Admissible Subgraphs). Let G' be an admissible subgraph of G. Then G' is a *strongly admissible subgraph* of G if its remoteness only yields natural numbers for all positions in G'.

Additionally, pure provenance of a `won` or `lost` position in a game represents a strongly admissible labeling of the dual AF. This follows because only position values with natural numbers are used to construct pure provenance.

Corollary 13 (Pure Provenance *vs.* Strong Admissibility). The pure provenance $\mathcal{P}_{\mathsf{pu}}(x)$ of a position in G is a strongly admissible subgraph of G.

4.4 Applying Game Provenance to Argumentation Frameworks

Game provenance can be directly applied to AFs based on the Game–AF duality.

Definition 16 (AF Potential Provenance). The *potential provenance* $\mathcal{P}_{\mathsf{pt}}(x)$ of argument x is the subgraph of arguments and attacks that reach x in F.

In games, the provenance of a node x is determined by what can be reached (via moves) from x, while in AFs (with edges reversed), x's provenance depends on the arguments that can reach it (i.e., attack x directly or indirectly). As in games, the potential provenance $\mathcal{P}_{\mathsf{pt}}(x)$ is an overestimate of the actual provenance (it includes attacks that correspond to blunders in SAG). The following defines the edge types of AFs for actual provenance.

Definition 17 (AF Edge Types). Let $F = (A, R)$ and $\mathcal{L}ab$ be its grounded labeling. The *edge types* $\tau : A \times A \to \{\mathtt{out}, \mathtt{in}, \mathtt{undec}, \mathtt{blunder}\}$ are defined by:

$$\tau(x,y) := \begin{cases} \mathtt{out} & \text{if } \mathcal{L}ab(x) = \mathtt{out} \text{ and } \mathcal{L}ab(y) = \mathtt{in} \\ \mathtt{in} & \text{if } \mathcal{L}ab(x) = \mathtt{in} \text{ and } \mathcal{L}ab(y) = \mathtt{out} \\ \mathtt{undec} & \text{if } \mathcal{L}ab(x) = \mathtt{undec} \text{ and } \mathcal{L}ab(y) = \mathtt{undec} \\ \mathtt{blunder} & \text{otherwise.} \end{cases}$$

Actual provenance for AFs is then defined as:

Definition 18 (AF Actual Provenance). $\mathcal{P}_{\mathsf{ac}}(x)$, the *actual provenance* of x, is the subgraph of F that reaches x by following `in`, `out`, and `undec` edges.

As in games, the actual provenance of an AF discards blunder attacks, but may include suboptimal attacks according to $\mathcal{MM}_{\mathcal{L}ab}$. The primary provenance of an AF removes suboptimal attacks:

Definition 19 (AF Primary Provenance). $\mathcal{P}_{\mathsf{pr}}(x)$ is the subset of $\mathcal{P}_{\mathsf{ac}}(x)$ that excludes `in` attacks $(x, y) \in R$ where $\mathcal{MM}_{\mathcal{L}ab}(x) \neq \mathcal{MM}_{\mathcal{L}ab}(y) - 1$.

Fig. 3b highlights the suboptimal attack $\mathsf{C}{\to}\mathsf{B}$ within the actual provenance of A. Like games, the smaller explanations provided by primary provenance may not include all well-founded explanations of an argument, unlike in pure provenance:

Definition 20 (AF Pure Provenance). $\mathcal{P}_{\mathsf{pu}}(x)$ is the subset of $\mathcal{P}_{\mathsf{ac}}(x)$ that excludes arguments y that cannot reach x via a pure (winning) strategy in SAG.

Fig. 4b gives the pure provenance of E, which discards the unfounded attack from F. Finally, from Corollary 13, the pure provenance $\mathcal{P}_{\mathsf{pu}}(x)$ of argument x is a strongly admissible set of F, which also provides the well-founded justification for the grounded label of x.

5 Conclusion

We established formal connections between min-max numberings in abstract argumentation and optimal play in combinatorial games. By linking min-max numbers to SMITH's remoteness function, provenance-based explanations can be directly applied to AFs. We also showed that pure strategy-based explanations provide a new class of provenance that bridges optimal and minimal approaches. Finally, we obtained new insights into min-max numberings via remoteness, including that parity determines argument labeling status and enables efficient computation of grounded labelings for admissible AF subgraphs.

Connections between game theory and argumentation have been studied extensively. Dung's seminal paper on argumentation frameworks [13] drew on n-player cooperative games from [17], while [16] uses similar game-theoretic concepts for defining argument strength. Two-player combinatorial games can be viewed as instances of n-person games in [18] where notions of independence and dominance apply. However, existing two-player dialog games for argumentation [7,10] operate on already-labeled AFs under specific semantics like strongly admissible and stable extensions, rather than establishing a direct correspondence between unlabeled frameworks and games that we develop here.

In future work, we aim to further explore the connections between games and argumentation. Since checking whether a strongly admissible labeling is minimal is co-NP-complete [8] for a given `in`-labeled argument, we conjecture that constructing minimal provenance explanations in games faces similar computational challenges. This contrasts with our remoteness-based provenance explanations, which can be computed efficiently. Building on approximation techniques [11,9], we will investigate tractable methods for computing approximately minimal explanations while preserving the theoretical guarantees of our duality framework.

References

1. Baroni, P., Giacomin, M.: On principle-based evaluation of extension-based argumentation semantics. Artificial Intelligence **171**(10-15), 675–700 (2007)
2. Baumann, R., Linsbichler, T., Woltran, S.: Verifiability of argumentation semantics. In: COMMA. pp. 83–94 (2016)
3. Boros, E., Gurvich, V., Makino, K., Vyalyi, M.: Computing remoteness functions of Moore, Wythoff, and Euclid's games. International Journal of Game Theory **53**(4), 1315–1333 (Dec 2024)
4. Bowers, S., Xia, Y., Ludäscher, B.: The skeptic's argumentation game or: well-founded explanations for mere mortals. In: SAFA'24 (2024)
5. Bowers, S., Xia, Y., Ludäscher, B.: On the structure of game provenance and its applications. In: Theory and Practice of Provenance (TaPP) at EuroS&PW (2024)
6. Caminada, M.: Argumentation semantics as formal discussion. In: Handbook of Formal Argumentation, vol. 1, pp. 487–518. College Publications (2018)
7. Caminada, M., Dunne, P.: Strong admissibility revisited: theory and applications. Argument & Computation **10**, 277–300 (2019)
8. Caminada, M., Dunne, P.: Minimal strong admissibility: a complexity analysis. In: COMMA. pp. 135–146 (2020)
9. Caminada, M., Harikrishnan, S.: An evaluation of algorithms for strong admissibility. In: SAFA. pp. 69–82 (2024)
10. Caminada, M., Wu, Y.: An argument game for stable semantics. Logic Journal of the IGPL **17**(1), 77–90 (01 2009)
11. Caminada, M., Harikrishnan, S.: Tractable algorithms for strong admissibility. Argument & Computation **16**(2), 212–236 (2025)
12. Caminada, M., Pigozzi, G.: On judgment aggregation in abstract argumentation. Autonomous Agents and Multi-Agent Systems **22**(1), 64–102 (2011)
13. Dung, P.: 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. Fraenkel, A.S.: Combinatorial game theory foundations applied to digraph kernels. Electron. J. Comb. **4**(2) (1997)
15. Gelder, A.V., Ross, K.A., Schlipf, J.S.: The well-founded semantics for general logic programs. J. ACM **38**(3), 620–650 (1991)
16. Matt, P.A., Toni, F.: A game-theoretic measure of argument strength for abstract argumentation. In: Logics in Artificial Intelligence. pp. 285–297 (2008)
17. von Neumann, J., Morgenstern, O.: Theory of games and economic behavior. Princeton University Press (1944)
18. Roth, A.E.: Subsolutions and the supercore of cooperative games. Math. Oper. Res. **1**(1), 43–49 (1976)
19. Roth, A.E.: Two-person games on graphs. Journal of Combinatorial Theory, Series B **24**(2), 238–241 (1978)
20. Siegel, A.: Combinatorial game theory, Graduate Studies in Mathematics, vol. 146. American Mathematical Society (2013)
21. Smith, C.A.: Graphs and composite games. Journal of Combinatorial Theory **1**(1), 51–81 (1966)