Exploiting attack–defense trees to find an optimal set of countermeasures: An online appendix

Barbara Fila, Wojciech Wideł

April 27, 2020

1 Example of the bottom-up evaluation of the satisfiability attribute

The following example illustrates our formalization of a goal achievement in ADTrees.

Example 4. *Let T be the ADTree from Figure 1. Assuming that the attacker executes only the actions* $\mathtt{a}$ *and* $\mathtt{c}$ *and the defender executes only* $\mathtt{d}_1, \mathtt{d}_2$*, and* $\mathtt{d}_3$*, which is modeled by assigning* 1 *to each of these actions and* 0 *to the remaining basic actions, the defender fails to counter the action* $\mathtt{a}$ *(the value computed at the* AND *node countering* $\mathtt{a}$ *is* 0*), and so the attacker achieves the goal of the root node (the value computed at the root node is* 1*). In other words, for the set* $X = \{\mathtt{a}, \mathtt{c}, \mathtt{d}_1, \mathtt{d}_2, \mathtt{d}_3\}$ *the equality* $\mathrm{achieved}_{\mathrm{T}}(\mathrm{root}(T), X) = 1$ *holds.*

2 Proof of Corollary 1

For the proof, we assume that $\mathtt{b} \in B$, since otherwise the statement is obviously true.

Suppose first that $\mathrm{actor}(v) = \mathrm{s}_T$. The value of $\mathrm{achieved}_{\mathrm{T}}(v, B)$ is computed by replacing the 0 assigned to the variable corresponding to the basic action $\mathtt{b}$ in $\mathrm{achieved}_{\mathrm{T}}(v, B \setminus \{\mathtt{b}\})$ into 1. Since $\mathtt{b} \in B$ and $B \subseteq \mathbb{B}^{\mathrm{s}_T}$, the function $\mathrm{achieved}_{\mathrm{T}}(v, \cdot)$ is positive in that variable. Together with the equality $\mathrm{achieved}_{\mathrm{T}}(v, B) = 0$, this implies that $\mathrm{achieved}_{\mathrm{T}}(v, B \setminus \{\mathtt{b}\}) = 0$.

Suppose now that $\mathrm{actor}(v) = \overline{\mathrm{s}}_T$. Observe that Definition 1 and 3 together with the definition of the *satisfiability* attribute domain imply that $\mathrm{achieved}_{\mathrm{T}}(v, \emptyset) = 0$. Together with the fact, the function $\mathrm{achieved}_{\mathrm{T}}(v, \cdot)$ is negative in the variables corresponding to the actions belonging to the set B, this implies that $\mathrm{achieved}_{\mathrm{T}}(v, B \setminus \{\mathtt{b}\}) = 0$. □

3 Lemma 2

Lemma 2. *Let $T = (V, E, \mathrm{L}, \lambda, \mathrm{actor}, \tau)$ be an ADTree, and let $P \subseteq \mathbb{B}^{\mathrm{p}_T}$ and $O \subseteq \mathbb{B}^{\mathrm{o}_T}$ be sets of basic actions of the actors. If $v \in V$ is a node such that*

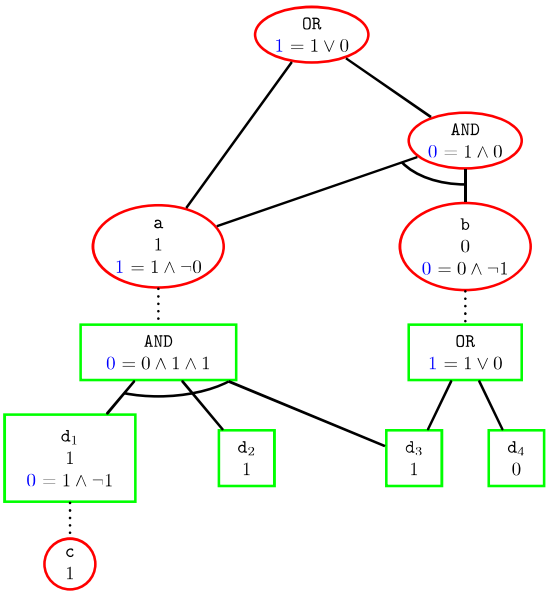

Figure 1: Bottom-up evaluation of the *satisfiability* attribute. Values assigned to the basic actions are given in black, and values computed at the intermediate nodes using the attribute domain's operations in dark blue.

- $\tau(v) = \mathtt{N}$,
- $\text{achieved}_{\mathrm{T}}(\text{root}(T), P \cup O) = 1$, *and*
- *on every path from* v *to* $\text{root}(T)$, *there exists a node* v' *s.t.* $\text{achieved}_{\mathrm{T}}(v', P) = 0$ *and* $\text{achieved}_{\mathrm{T}}(v', O) = 0$,

then $\text{achieved}_{\mathrm{T}}(\text{root}(T), P \cup O \setminus \{\lambda(v)\}) = 1$.

Proof. Let v' be one of the nodes satisfying the last condition of the lemma. Corollary 4 implies that $\text{achieved}_{\mathrm{T}}(v', P \cup O) = 0$. Therefore, when the value of $\text{achieved}_{\mathrm{T}}(\text{root}(T), P \cup O)$ is computed using the bottom-up procedure, the value propagated up to the root from v' is zero. Furthermore, it follows from Corollary 1 that $\text{achieved}_{\mathrm{T}}(v', O \setminus \{\lambda(v)\}) = 0$ and $\text{achieved}_{\mathrm{T}}(v', P \setminus \{\lambda(v)\}) = 0$. Thus, by Corollary 4, $\text{achieved}_{\mathrm{T}}(v', P \cup O \setminus \{\lambda(v)\}) = 0$, i.e., the value propagated from v' remains unchanged after the removal of the basic action $\lambda(v)$ from $P \cup O$. Hence,

$$\begin{aligned}&\text{achieved}_{\mathrm{T}}(\text{root}(T), P \cup O \setminus \{\lambda(v)\}) = \\ &\text{achieved}_{\mathrm{T}}(\text{root}(T), P \cup O) = 1.\end{aligned}$$

□

4 Example of the bottom-up evaluation of the sufficient witnesses attribute

Example 5 generalizes the reasoning from Example 3.

Example 5. *Let T be an ADTree being a path of $2n+1$ alternating non-refined nodes of the proponent and the opponent, with the first node on the path belonging to the proponent, as schematized in Figure 2. The total number of non-empty opponent's strategies in T is 2^n-1, whereas there are only $n-1$ strategies in the result of the bottom-up evaluation of* `SuffWit` *on T. Furthermore, each of them is a unique witness for one of the proponent's strategies: the opponent's strategy $\{\mathtt{d}_1,\ldots,\mathtt{d}_i\}$, with $i \in \{1,\ldots,n\}$, is the unique witness for the proponent's strategy $\{\mathtt{b}_1,\ldots,\mathtt{b}_{i+1}\}$.*

Observe the following: if O is an opponent's strategy belonging to the result of the bottom-up evaluation of `SuffWit` *on T and $\mathtt{d}_i \in O$, with $i \in \{1,\ldots,n\}$, then $\mathtt{d}_j \in O$ for every $j \in \{1,\ldots,i-1\}$. Informally speaking, there are no "gaps" in the obtained opponent's strategies. This is intentional: should the above condition be not satisfied by an opponent's strategy O, say, $O = \{\mathtt{d}_1,\ldots,\mathtt{d}_i,\mathtt{d}_{i+k}\}$, with $i, i+k \in \{1,\ldots,n\}$, $k>1$, then O is a witness for the same proponent's strategies as $\{\mathtt{d}_1,\ldots,\mathtt{d}_i\}$. This example motivates our choice of the operation $\odot$ as the one to be performed in the bottom-up procedure when traversing countermeasures against goals of the opponent.*

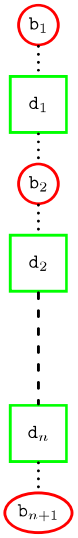

Figure 2: A schema generalizing the tree from Figure 3

5 Proof of Lemma 1

The proof is by induction on the structure of T'. We consider three cases.

Case 1. The node v is not refined and $\bar{v} \notin V'$.
Since the set $\mathbb{B}^{\mathrm{o}_T}_{T'}$ is not empty, it follows that $\mathrm{actor}(v) = \mathrm{o}_T$, $\mathbb{B}^{\mathrm{o}_T}_{T'} = \{\lambda(v)\}$, and $\mathtt{SuffWit}(T, \beta^{\mathtt{SW}}_T, v) = \{\{\lambda(v)\}\}$. Thus, the claim holds.

Case 2. The node v is not refined and $\bar{v} \in V'$.
If $\mathrm{actor}(v) = \mathrm{p}_T$, then $\mathbb{B}^{\mathrm{o}_T}_{T'} = \mathbb{B}^{\mathrm{o}_T}_{T'(\bar{v})}$. Since $\mathbb{B}^{\mathrm{o}_T}_{T'} \neq \emptyset$, it follows that $\mathbb{B}^{\mathrm{o}_T}_{T'(\bar{v})} \neq \emptyset$, and so the subdag $T'(\bar{v})$ of T' rooted at $\bar{v}$ satisfies all of the assumptions of the

lemma. Thus, by the induction hypothesis, we have $\mathbb{B}^{o_T}_{T'(\bar{v})} \in \texttt{SuffWit}(T, \beta^{\texttt{SW}}_T, \bar{v})$. The definition of the attribute domain for `SuffWit` and the operation $\oplus$ defined by formula (2) imply that $\texttt{SuffWit}(T, \beta^{\texttt{SW}}_T, \bar{v}) \subseteq \texttt{SuffWit}(T, \beta^{\texttt{SW}}_T, v)$. Hence, $\mathbb{B}^{o_T}_{T'} \in \texttt{SuffWit}(T, \beta^{\texttt{SW}}_T, v)$.

If $\text{actor}(v) = o_T$, then $\mathbb{B}^{o_T}_{T'} = \mathbb{B}^{o_T}_{T'(\bar{v})} \cup \{\lambda(v)\}$. From the definition of the attribute domain for `SuffWit`, the basic assignment $\beta^{\texttt{SW}}_T$, and the operation $\odot$ defined by formula (4), it follows that both sets $\{\{\lambda(v)\}\}$ and $\texttt{SuffWit}(T, \beta^{\texttt{SW}}_T, \bar{v}) \otimes \{\{\lambda(v)\}\}$ are subsets of $\texttt{SuffWit}(T, \beta^{\texttt{SW}}_T, v)$. Therefore, regardless of whether the set $\mathbb{B}^{o_T}_{T'(\bar{v})}$ is empty or not, we have $\mathbb{B}^{o_T}_{T'} \in \texttt{SuffWit}(T, \beta^{\texttt{SW}}_T, v)$, as required.

Case 3. The node v is refined.
Let k be the size of the (possibly empty) set $\{v' \in \text{ref}_T(v) \cap V' \mid \mathbb{B}^{o_T}_{T'(v')} \neq \emptyset\}$. If $k \neq 0$, we use $v_1, \ldots, v_k$ to denote the elements of this set. Depending on whether or not $\bar{v} \in V'$, we have

$$\mathbb{B}^{o_T}_{T'} = \bigcup_{i=1}^{k} \mathbb{B}^{o_T}_{T'(v_i)}$$

or

$$\mathbb{B}^{o_T}_{T'} = \bigcup_{i=1}^{k} \mathbb{B}^{o_T}_{T'(v_i)} \cup \mathbb{B}^{o_T}_{T'(\bar{v})}.$$

Note that, since $\mathbb{B}^{o_T}_{T'} \neq \emptyset$, this implies that $k \geq 1$ or the set $\mathbb{B}^{o_T}_{T'(\bar{v})}$ is not empty. Observe also that, by the induction hypothesis, $\mathbb{B}^{o_T}_{T'(v_i)} \in \texttt{SuffWit}(T, \beta^{\texttt{SW}}_T, v_i)$, for every $i \in \{1, \ldots, k\}$. If in addition $\bar{v} \in V'$ and $\mathbb{B}^{o_T}_{T'(\bar{v})} \neq \emptyset$, then also $\mathbb{B}^{o_T}_{T'(\bar{v})} \in \texttt{SuffWit}(T, \beta^{\texttt{SW}}_T, \bar{v})$.

We distinguish two subcases, depending on the value of k.

Case 3.1 $k = 0$. The assumption of this case implies that $\text{actor}(v) = \text{p}_T$, $\bar{v} \in V'$, and $\mathbb{B}^{o_T}_{T'} = \mathbb{B}^{o_T}_{T'(\bar{v})} \neq \emptyset$. Similarly to Case 2, now it follows from the definition of the attribute domain for `SuffWit` and the operation defined by formula (2) that $\texttt{SuffWit}(T, \beta^{\texttt{SW}}_T, \bar{v}) \subseteq \texttt{SuffWit}(T, \beta^{\texttt{SW}}_T, v)$. Hence, $\mathbb{B}^{o_T}_{T'} \in \texttt{SuffWit}(T, \beta^{\texttt{SW}}_T, v)$.

Case 3.2 $k > 0$.
In this case, the definition of the attribute domain for `SuffWit` and the operations given by formulæ (1), (4), and (2) imply that the set

$$\bigotimes_{i=1}^{k} \texttt{SuffWit}(T, \beta^{\texttt{SW}}_T, v_i)$$

as well as the set

$$\texttt{SuffWit}(T, \beta^{\texttt{SW}}_T, \bar{v}) \otimes \bigotimes_{i=1}^{k} \texttt{SuffWit}(T, \beta^{\texttt{SW}}_T, v_i),$$

if $\bar{v}$ exists, are subsets of $\texttt{SuffWit}(T, \beta^{\texttt{SW}}_T, v)$. Thus, $\bigcup_{i=1}^{k} \mathbb{B}^{o_T}_{T'(v_i)} \in \texttt{SuffWit}(T, \beta^{\texttt{SW}}_T, v)$, and if $\bar{v}$ exists and the set $\mathbb{B}^{o_T}_{T'(\bar{v})}$ is not empty, then also $\bigcup_{i=1}^{k} \mathbb{B}^{o_T}_{T'(v_i)} \cup \mathbb{B}^{o_T}_{T'(\bar{v})} \in \texttt{SuffWit}(T, \beta^{\texttt{SW}}_T, v)$. This completes the proof of Lemma 1. □

6 Proof of Proposition 3

The proof is by induction on the structure of $T(v)$ – the maximal subdag of T rooted at v.

For the base case, let v be a non-refined node and assume that $\overline{v}$ does not exist. Since the set $\texttt{CounterOpp}(T, \beta_T^O, v)$ is not empty, the definition of the basic assignment β_T^O implies that $\text{actor}(v) = \text{p}_T$ and $P = \{\lambda(v)\}$, or $\text{actor}(v) = \text{o}_T$ and $P = \emptyset$. In the former case, the claim follows immediately. In the latter, we have $\lambda(v) \notin O$, implying that

$$\text{achieved}_\text{T}(v, P \cup O) = \text{achieved}_\text{T}(v, O) = 0,$$

as required.

Case 1. The node v is not refined and $\overline{v}$ exists.

Case 1.1. $\text{actor}(v) = \text{p}_T$.
Under the assumptions of this case, and since the set $\texttt{CounterOpp}(T, \beta_T^O, v)$ is not empty, formula (1), the definition of the `CounterOpp` domain, and the definition of the basic assignment β_T^O imply that $\texttt{CounterOpp}(T, \beta_T^O, \overline{v}) \neq \emptyset$ and $P = \overline{P} \cup \{\lambda(v)\}$, for some set $\overline{P} \in \texttt{CounterOpp}(T, \beta_T^O, \overline{v})$. By the induction hypothesis, the equality $\text{achieved}_\text{T}(\overline{v}, \overline{P} \cup O) = 0$ holds, and so the definition of the *satisfiability* attribute domain implies that $\text{achieved}_\text{T}(v, P \cup O) = 1$.

Case 1.2. $\text{actor}(v) = \text{o}_T$.
In the case when $\lambda(v) \in O$, we have $\texttt{CounterOpp}(T, \beta_T^O, v) = \texttt{CounterOpp}(T, \beta_T^O, \overline{v})$, by formula (3) and the definition of β_T^O. Thus, $P \in \texttt{CounterOpp}(T, \beta_T^O, \overline{v})$, which together with the induction hypothesis implies $\text{achieved}_\text{T}(\overline{v}, P \cup O) = 1$. Now the equality $\text{achieved}_\text{T}(v, P \cup O) = 0$ follows from the definition of the *satisfiability* attribute domain.

If $\lambda(v) \notin O$, then $\beta_T^O(v) = \{\emptyset\}$, and so $\texttt{CounterOpp}(T, \beta_T^O, v) = \{\emptyset\}$, by formula (3). And indeed, since $\lambda(v) \notin O$, the demanded equality

$$\text{achieved}_\text{T}(v, P \cup O) = \text{achieved}_\text{T}(v, O) = 0$$

follows from the definition of the *satisfiability* attribute domain.

If v is a refined node, we let $\text{ref}_T(v) = \{v_1, \ldots, v_k\}$.

Case 2. The node v is refined and $\tau(v) = \texttt{OR}$.

Case 2.1. $\text{actor}(v) = \text{p}_T$.
Depending on whether or not $\overline{v}$ exists, either $P = P_i \cup \overline{P}$ (if $\overline{v}$ does exist) or $P = P_i$ (if $\overline{v}$ does not exist), for some $i \in \{1, \ldots, k\}$, $P_i \in \texttt{CounterOpp}(T, \beta_T^O, v_i)$ and $\overline{P} \in \texttt{CounterOpp}(T, \beta_T^O, \overline{v})$. Thus, by the induction hypothesis, we have $\text{achieved}_\text{T}(\overline{v}, \overline{P} \cup O) = 0$ and $\text{achieved}_\text{T}(v_i, P_i \cup O) = 1$, implying that $\text{achieved}_\text{T}(v, P \cup O) = 1$.

Case 2.2. $\text{actor}(v) = \text{o}_T$.
Again, depending on the existence of $\overline{v}$, and, if it does exist, on whether or not the set $\texttt{CounterOpp}(T, \beta_T^O, \overline{v})$ is empty, either $P = (P_1 \cup \ldots \cup P_k)$, for some $P_i \in \texttt{CounterOpp}(T, \beta_T^O, v_i)$ or $P = \overline{P}$, for some $\overline{P} \in \texttt{CounterOpp}(T, \beta_T^O, \overline{v})$. In the latter case, we have $\text{achieved}_\text{T}(\overline{v}, \overline{P} \cup O) = 1$, by the induction hypothesis, and the equality $\text{achieved}_\text{T}(v, \overline{P} \cup O) = 0$ follows from the definition of the *satisfiability* attribute domain. Suppose now that the former of the two cases occurs. The induction hypothesis implies that $\text{achieved}_\text{T}(v_i, P_i \cup O) = 0$, for

$i \in \{1, \ldots, k\}$. Now, it follows from Corollary 2 that $\text{achieved}_{\text{T}}(v_i, P \cup O) = 0$, for $i \in \{1, \ldots, k\}$. Thus, $\text{achieved}_{\text{T}}(v, P \cup O) = 0$.

Case 3. The node v is refined and $\tau(v) = \texttt{AND}$.

Case 3.1. $\text{actor}(v) = \text{p}_T$.
Depending on the existence of the countermeasure $\overline{v}$, it either holds that $P = (P_1 \cup \ldots \cup P_k) \cup \overline{P}$ or $P = (P_1 \cup \ldots \cup P_k)$, for some $P_i \in \texttt{CounterOpp}(T, \beta_T^O, v_i)$ and $\overline{P} \in \texttt{CounterOpp}(T, \beta_T^O, \overline{v})$. The induction hypothesis implies that $\text{achieved}_{\text{T}}(v_i, P_i \cup O) = 1$, for $i \in \{1, \ldots, k\}$, and $\text{achieved}_{\text{T}}(\overline{v}, \overline{P} \cup O) = 0$. By applying Corollary 2, we get $\text{achieved}_{\text{T}}(v, P \cup O) = 1$.

Case 3.2. $\text{actor}(v) = \text{o}_T$.
If $\overline{v}$ does not exist or $\texttt{CounterOpp}(T, \beta_T^O, \overline{v}) = \emptyset$, then, by formula (3), $P = P_i$, for some $i \in \{1, \ldots, k\}$ and $P_i \in \texttt{CounterOpp}(T, \beta_T^O, v_i)$. Otherwise, it might hold that $P = \overline{P}$, for some $\overline{P} \in \texttt{CounterOpp}(T, \beta_T^O, \overline{v})$. In either case, the demanded equality follows from the induction hypothesis and the definition of the *satisfiability* attribute domain. □

7 Proof of Proposition 4

The proof of Proposition 4 is by induction on the structure of $T(v)$, the maximal subdag of T rooted at v. For the base case, assume that v has no children at all, i.e., that $\text{ref}_T(v) = \emptyset$ and that $\overline{v}$ does not exist. If $\text{actor}(v) = \text{p}_T$, then the only set P satisfying the assumptions of the theorem is $P = \{\lambda(v)\}$. If $\text{actor}(v) = \text{o}_T$, then either no such P exists (if $\lambda(v) \in O$) or $P = \emptyset$ (if $\lambda(v) \notin O$). In either case, the statement holds.

We now proceed with the remaining cases.

Case 1. The node v is not refined and $\overline{v}$ exists.

Case 1.1. $\text{actor}(v) = \text{p}_T$.
Since $\text{achieved}_{\text{T}}(v, P \cup O) = 1$, the assumptions of this case imply that $\text{achieved}_{\text{T}}(\overline{v}, P \cup O) = 0$. From the minimality of P it follows that P can be represented as $\overline{P} \cup \{\lambda(v)\}$, for some minimal set $\overline{P}$ satisfying $\text{achieved}_{\text{T}}(\overline{v}, \overline{P} \cup O) = 0$. By induction hypothesis, we have $\overline{P} \in \texttt{CounterOpp}(T, \beta_T^O, \overline{v})$, and so $P \in \texttt{CounterOpp}(T, \beta_T^O, v)$.

Case 1.2. $\text{actor}(v) = \text{o}_T$.
The proof in this case is analogous to that from the previous one. Nevertheless, we include it for completeness. Since $\text{achieved}_{\text{T}}(v, P \cup O) = 0$, it follows from the definition of the *satisfiability* domain that $\text{achieved}_{\text{T}}(\overline{v}, P \cup O) = 1$. The minimality of P implies that $P = \overline{P}$, for some minimal set $\overline{P}$ satisfying $\text{achieved}_{\text{T}}(\overline{v}, \overline{P} \cup O) = 1$. As $\overline{P} \in \texttt{CounterOpp}(T, \beta_T^O, \overline{v})$, by the induction hypothesis, the definition of the `CounterOpp` attribute domain now implies that $P \in \texttt{CounterOpp}(T, \beta_T^O, v)$, as required.

For a proof of the remaining cases, when v is a refined node, we let $\text{ref}_T(v) = \{v_1, \ldots, v_k\}$ and assume that the node $\overline{v}$ exists. The proof for the cases when $\overline{v}$ does not exist is obtained by skipping the parts related to $\overline{v}$ in what follows.

Case 2. The node v is refined and $\tau(v) = \texttt{OR}$.

Case 2.1. $\text{actor}(v) = \text{p}_T$.
We begin with proving that there exists $i \in \{1, \ldots, k\}$, a minimal set P' for which $\text{achieved}_{\text{T}}(v_i, P' \cup O) = 1$ and a minimal set $\overline{P}$ for which $\text{achieved}_{\text{T}}(\overline{v}, \overline{P} \cup$

$O) = 0$, such that $P = P' \cup \overline{P}$. To obtain such sets P' and $\overline{P}$, proceed iteratively as follows. Set $P' := P$, $\overline{P} := P$. As long as there exists a basic action $\mathtt{b} \in \overline{P}$ such that $\text{achieved}_{\mathrm{T}}(\overline{v}, \overline{P} \cup O \setminus \{\mathtt{b}\}) = 0$, set $\overline{P} := \overline{P} \setminus \{\mathtt{b}\}$. Similarly, as long as there exists a basic action $\mathtt{b} \in P'$ such that $\text{achieved}_{\mathrm{T}}(v_i, P' \setminus \{\mathtt{b}\} \cup O) = 1$, for at least one $i \in \{1, \ldots, k\}$, remove $\mathtt{b}$ from P'. Observe that, by the minimality of P, the actions that were removed from $\overline{P}$ belong to P', and those removed from P' belong to $\overline{P}$. In other words, the equality $P = P' \cup \overline{P}$ indeed holds. Furthermore, the sets P' and $\overline{P}$ are minimal sets satisfying $\text{achieved}_{\mathrm{T}}(\overline{v}, \overline{P} \cup O) = 0$ and $\text{achieved}_{\mathrm{T}}(v_i, P' \cup O) = 1$, for some $i \in \{1, \ldots, k\}$. Thus, by the induction hypothesis, we have that $P' \in \mathtt{CounterOpp}(T, \beta_T^O, v_i)$ and $\overline{P} \in \mathtt{CounterOpp}(T, \beta_T^O, \overline{v})$. Hence, $P \in \mathtt{CounterOpp}(T, \beta_T^O, v)$.

Case 2.2. $\text{actor}(v) = \mathrm{o}_T$.
If $P \in \mathtt{CounterOpp}(T, \beta_T^O, \overline{v})$, then the definition of the `CounterOpp` attribute domain and operation $\oslash$ defined by (3) imply that $P \in \mathtt{CounterOpp}(T, \beta_T^O, v)$. Thus, in this case the claimed statement holds.

Assume now that $P \notin \mathtt{CounterOpp}(T, \beta_T^O, \overline{v})$. Since $\text{achieved}_{\mathrm{T}}(v, P \cup O) = 0$, it follows from the definition of the *satisfiability* attribute domain that the equality $\text{achieved}_{\mathrm{T}}(v_i, P \cup O) = 0$ holds, for every $i \in \{1, \ldots, k\}$. Furthermore, P being a *minimal* set satisfying $\text{achieved}_{\mathrm{T}}(v, P \cup O) = 0$ implies that it can be represented as

$$P = P_1 \cup \ldots \cup P_k \tag{9}$$

for some minimal sets $P_1, \ldots, P_k$ satisfying $\text{achieved}_{\mathrm{T}}(v_i, P_i \cup O) = 0$. To see that this is indeed the case, suppose for a proof by contradiction that it is not. Then, in every representation (9) of P there exists a set P_i satisfying $\text{achieved}_{\mathrm{T}}(v_i, P_i \cup O) = 0$, for some $i \in \{1, \ldots, k\}$, that is not a minimal set having this property. Let $P'_1 \cup \ldots \cup P'_k$ be a representation (9) of P that minimizes the number of such non-minimal sets, and let P'_j be such a non-minimal set. Then, there exists a minimal set $P''_j \subset P'_j$ for which $\text{achieved}_{\mathrm{T}}(v_j, P''_j \cup O) = 0$ holds. The first statement of Corollary 2 implies that

$$\text{achieved}_{\mathrm{T}}(v_i, P'_1 \cup \ldots \cup P''_j \cup \ldots \cup P'_k \cup O) = 0,$$

for every $i \in \{1, \ldots, k\}$. Thus,

$$\text{achieved}_{\mathrm{T}}(v, P'_1 \cup \ldots \cup P''_j \cup \ldots \cup P'_k \cup O) = 0.$$

But from the choice of $P'_1 \cup \ldots \cup P'_k$, it follows that

$$P \supset P'_1 \cup \ldots \cup P''_j \cup \ldots \cup P'_k.$$

This contradicts the minimality of P. Thus, the set P admits the representation (9).

Now it follows from the induction hypothesis that for every $i \in \{1, \ldots, k\}$, the set P_i from (9) belongs to $\mathtt{CounterOpp}(T, \beta_T^O, v_i)$. Together with the definition of the `CounterOpp` attribute domain and operation $\otimes$ defined by (1), this fact implies that $P \in \mathtt{CounterOpp}(T, \beta_T^O, v)$, completing the proof of this case.

Case 3. The node v is refined and $\tau(v) = \mathtt{AND}$.

Case 3.1. $\text{actor}(v) = \mathrm{p}_T$.
The assumptions of this case and the fact that P is a minimal set for which the equality $\text{achieved}_{\mathrm{T}}(v, P \cup O) = 1$ holds imply that P can be represented

as $P = P_1 \cup \ldots \cup P_k \cup \overline{P}$, for some minimal sets $P_1, \ldots, P_k$ and $\overline{P}$ satisfying $\text{achieved}_{\mathrm{T}}(v_i, P_i \cup O) = 1$ and $\text{achieved}_{\mathrm{T}}(\bar{v}, \overline{P} \cup O) = 0$. By the induction hypothesis, we have $P_i \in \texttt{CounterOpp}(T, \beta_T^O, v_i)$, for $i \in \{1, \ldots, k\}$, and $\overline{P} \in \texttt{CounterOpp}(T, \beta_T^O, \bar{v})$. Thus, $P \in \texttt{CounterOpp}(T, \beta_T^O, v)$, by the definition of the `CounterOpp` attribute domain and operation $\otimes$ defined by (1).

Case 3.2. $\text{actor}(v) = \mathrm{o}_T$.
Similarly as in Case 2.1, we assume that $P \notin \texttt{CounterOpp}(T, \beta_T^O, \bar{v})$, since otherwise the claimed statement follows immediately. In this case, the definition of the *satisfiability* domain and the fact that P is a minimal set satisfying $\text{achieved}_{\mathrm{T}}(v, P \cup O) = 0$ imply that there exists $i \in \{1, \ldots, k\}$ for which P is a minimal set satisfying $\text{achieved}_{\mathrm{T}}(v_i, P \cup O) = 0$. Thus, by the induction hypothesis, $P \in \texttt{CounterOpp}(T, \beta_T^O, v_i)$. Now it follows immediately from the definition of the `CounterOpp` attribute domain that $P \in \texttt{CounterOpp}(T, \beta_T^O, v)$. □

8 Examples of extremal trees

The examples below illustrate the three observations from Section III-C.

Example 6. *Let T be the ADTree depicted on Figure 3. Every set of basic actions of the defender that contains a set of the form $\{b_{1i}, b_{2j}, b_{3k}\}$, with $i, j, k \in \{1, 2\}$, is a non-empty defender's strategy in T. There are 3^3 such defender's strategies in T, and each of them belongs to the result of the bottom-up evaluation of the* `SuffWit` *attribute on T.*

A simple proof by induction shows that in the general case, when there are n `OR` *nodes of the defender, there are 3^n non-empty defender's strategies.*

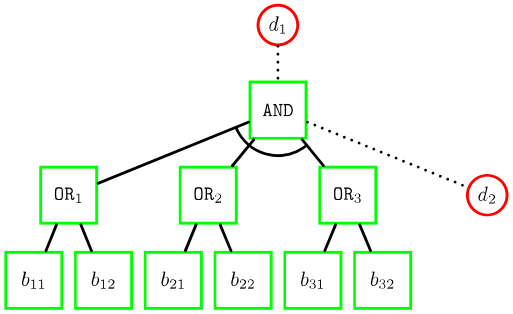

Figure 3: An example of a tree with the number of the opponent's strategies exponential in the number of basic actions

Example 7. *Let T be the ADTree depicted on Figure 4. Each of the sets of the form $\{b_1, b_{1i}, b_{2j}, b_{3k}\}$, with $i, j, k \in \{1, 2\}$, is an attacker's strategy witnessed by the defender's strategy $\{d_1\}$. There are 2^3 such strategies.*

Example 8. *Let T be the ADTree depicted on Figure 3. Each of the sets of the form $\{b_{1i}, b_{2j}, b_{3k}\}$, with $i, j, k \in \{1, 2\}$, is a minimal defender's strategy countering the attacker's strategy $\{d_1\}$. The number of such sets is 2^3.*

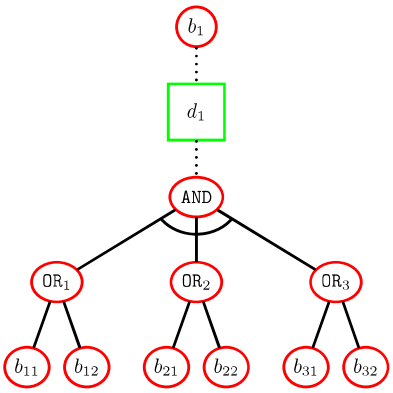

Figure 4: An example of a tree with the number of the proponent's strategies witnessed by a single opponent's strategy exponential in the number of basic actions

9 The OSEAD tool interface

Figure 5 illustrates our tool's interface allowing the user to input relevant parameters and displaying run times of different parts of solving the selected problem: the blue frame shows the time of the defense semantics extraction, and the red one the time spent on solving the actual optimization problem.

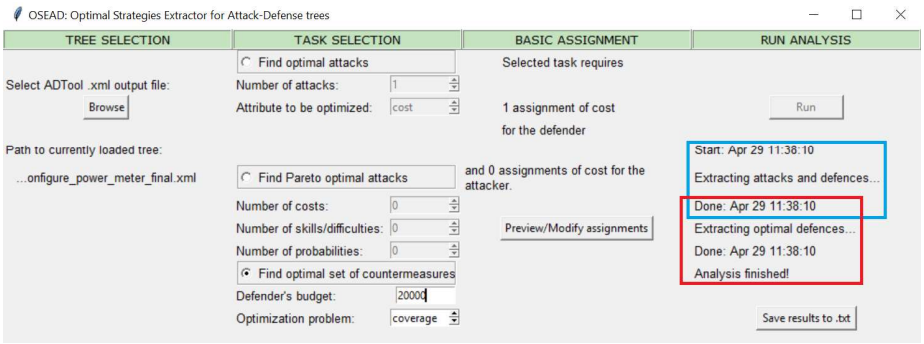

Figure 5: Solving the coverage problem using OSEAD