Learning as Interaction

Nina Gierasimczuk Lena Kurzen
Fernando R. Velázquez-Quesada

Abstract

In formal approaches to inductive learning, the ability to learn is understood as the ability to single out a correct hypothesis from a range of possibilities. Although most of the existing research focuses on the characteristics of the learner, in many paradigms the significance of the teacher's abilities and strategies is in fact undeniable. Motivated by this observation, in this paper we highlight the interactive nature of learning by proposing a game-theoretical and logical approach. We consider learning as a game, and present different levels of cooperativeness between the players. Then, we look at different variants of Sabotage Games as learning scenarios, expressing the conditions for learnability in Sabotage Modal Logic and analyzing their complexity. Our work constitutes the first step towards a unified game-theoretical and logical approach to formal learning theory.

1 Introduction

The objective of this paper is to investigate how logics for interaction in multi-agent systems can be used to reason about strategic abilities and information flow during the *learning* process. Formal learning theory (see e.g. [4]) is concerned with the process of inductive inference: it formalizes the process of inferring general conclusions from partial, inductively given information, as in the case of language learning (inferring grammars from sentences) and scientific inquiry (drawing general conclusions from partial experiments). We can think of this general process as a game between two players: Learner and Teacher. The game starts with a class of possible worlds from which Teacher chooses the actual one, and Learner has to find out which one it is. Teacher provides information about the world in an inductive manner, and whenever Learner receives a piece of information, he picks a conjecture from the initial class, indicating which one he thinks is the case. Several conditions can be defined for the success of the learning process: we can require that Learner arrives at a correct hypothesis (finite identification), or that the sequence of Learner's conjectures converges to a correct hypothesis (identification in the limit) [3].

We restrict ourselves to a high-level analysis of the process described above. Our proposal focuses on some important elements of learnability. First of all, we treat learning as a procedure of singling out one correct hypothesis from some range of possibilities. Moreover, we see this procedure not as a one-move choice, but as a sequence of them, and therefore we allow many steps of update before the conclusion is reached. Those two properties make our notion of learning different from the concept of learning formalized as epistemic update in dynamic epistemic logic (see e.g. [2]), where the word "learning" is often used as a synonym of "getting to know" and is usually represented as a one-step epistemic update. Moreover, in our approach we pay attention to the strategies for teaching, highlighting the fact that restricting the power and knowledge of the learner can be compensated by providing additional insights and intentions for the teacher.

2 Learning as a Sabotage Game

Our work is motivated by the *learning from queries and counterexamples* model [1]. In that paradigm, the goal of Learner is to recognize a given language, which is initially unknown to him. In order to do this, he is allowed to ask Teacher two types of questions: about the membership of a certain string to the unknown language, and about the equivalence of his conjecture to the unknown language. When answering those questions Teacher does not have any freedom — the responses are predetermined by the initial choice of the language. However, a negative answer to the second question is accompanied by a counterexample, which plays the role of a hint for Learner. This is the only point of the procedure in which Teacher has a relative freedom of choice, and in fact, the informativeness of the string given as a counterexample influences the effectiveness of the learning process. We want to focus on this aspect of learning and illustrate how in various scenarios the "profile" of Teacher can be encoded in his strategy. To make the picture of possible strategies in learning scenarios complete, we describe games in which Teacher is either helpful or unhelpful, and we also consider two possible "profiles" of Learner: to be either eager or unwilling.

Let us start by considering a very simple "classroom" situation with one Teacher and one Learner. From our high-level perspective, learning is a step-by-step process through which Learner changes his information state, and the process is successful if he eventually reaches a state representing the goal. The information that Teacher provides can be seen as feedback about Learner's current conjecture, allowing Learner to rule out possible changes of mind because they are inconsistent with the received information. By looking at the Teacher-Learner interaction from this perspective, we can represent the situation as a graph whose vertices represent Learner's possible information states and edges represent transitions between them. During the learning process, Learner can change his information state by moving along the edges and Teacher can cut off edges, thereby preventing Learner from making certain transitions. Some state is associated with the learning goal: whenever the agent reaches it, we say that the learning process has been successful. The correspondence between the learning model from formal learning theory and our proposal, based on the Sabotage Games defined below, is described in Table 1.

It is easy to observe that in learning from queries and counterexamples, giving an informative counterexample results in the absolute removal of some, initially possible, hypotheses. We want to generalize this idea and allow the multiplicity of transitions between two states, so that the removal of one transition does not in principle has to make it impossible to get from one state to another. Obviously, this view is more general, and therefore we analyze the

Table 1: Correspondence with Learning Model

Learning Model	**Sabotage Games**
hypotheses	states
correct hypothesis	goal state
possibility of a mind change from hypothesis *a* to hypothesis *b*	transition from state *a* to *b*
giving a counterexample that eliminates the possibility of a mind change from *a* to *b*	removing a transition between *a* and *b*

above-described paradigm of learning by queries and counterexamples only as a special case.

2.1 Sabotage Games

Our perspective on learning leads naturally to the framework of Sabotage Games [6, 10]. A *Sabotage Game* is played in a directed multi-graph, with two players, *Runner* and *Blocker*, alternatingly moving with Runner being the first. Runner moves by making a single transition from the current vertex; Blocker moves by deleting *any* edge from the graph. We begin by defining the structure in which a sabotage game takes place.

Definition 2.1 (Directed Multi-graph [6]). *A* (directed) multi-graph *is a tuple* $G = (V, E)$ *where* V *is a set of vertices and* $E : V \times V \to \mathbb{N}$ *is a function indicating the number of edges between any two vertices.*

Then the sabotage game is defined as follows.

Definition 2.2 (Sabotage Game [6]). *A* Sabotage Game *(SG) is a tuple* $SG = \langle V, E, v, v_g \rangle$*, where* (V, E) *is a multi-graph and* $v, v_g \in V$*. Vertex* v *represents the position of Runner and* v_g *represents the goal state.*

The game is played as follows: the initial position $SG_0 = \langle V, E_0, v_0, v_g \rangle$ *is given by* $SG_0 = SG$*. Round* $k+1$*, from position* $SG_k = \langle V, E_k, v_k, v_g \rangle$*, consists of Runner choosing some* v_{k+1} *such that* $E(v_k, v_{k+1}) > 0$*, and Blocker answering by choosing some* (v, v') *such that* $E_k(v, v') > 0$*. The new position* $SG_{k+1} = \langle V, E_{k+1}, v_{k+1}, v_g \rangle$ *is such that* $E_{k+1} := E_k(v, v') - 1$ *and, for every* $(u, u') \neq (v, v')$*,* $E_{k+1}(u, u') := E_k(u, u')$*. The game ends if some player cannot make a move or if* $v_k = v_g$*, and Runner wins iff* $v_l = v_g$ *for some round* l*.*

Note that in this definition of the Sabotage Game, Blocker removes an edge between two states v, v' by decreasing the value of $E(v, v')$ by 1. As we will see later, this definition of the game based on the above definition of multi-graphs can lead to some technical problems when transforming such a graph into a Kripke model. Therefore, we will now present an alternative definition. In Theorem 1, we then show that for our purposes the definitions are equivalent with respect to the existence of a winning strategy.

Definition 2.3 (Directed Labelled Multi-graph). *Let* $\Sigma = \{a_1, \dots a_n\}$ *be a finite set of labels. A* (directed) labelled multi-graph *is a tuple* $\mathsf{G}^\Sigma = (V, \mathcal{E})$ *where* V *is a set of vertices and* $\mathcal{E} = (\mathcal{E}_{a_1}, \dots, \mathcal{E}_{a_n})$*, where* $\mathcal{E}_{a_i} \subseteq V \times V$ *for each* $a_i \in \Sigma$*.*

In this definition, labels from Σ are used to represent multiple edges between two vertices. Then, the definition of the game is as follows.

Definition 2.4 (Labelled Sabotage Game). *A* Labelled Sabotage Game *(SG^Σ) is a tuple* $SG^\Sigma = \langle V, \mathcal{E}, v, v_g \rangle$*, where* $(V, \mathcal{E})$ *is a labelled multi-graph and* $v, v_g \in V$*. Vertex* v *represents the position of Runner and* v_g *represents the goal state.*

The game is played as follows: the initial position $SG_0^\Sigma = \langle V, \mathcal{E}^0, v_0, v_g \rangle$ *is given by* $SG_0^\Sigma = SG^\Sigma$*. Round* $k+1$*, from position* $SG_k = \langle V, \mathcal{E}^k, v_k, v_g \rangle$ *with* $\mathcal{E}^k = (\mathcal{E}^k_{a_1}, \dots, \mathcal{E}^k_{a_n})$*, consists of Runner choosing some* v_{k+1} *such that there is some* $a_i \in \Sigma$ *for which* $(v_k, v_{k+1}) \in \mathcal{E}^k_{a_i}$*, and Blocker answering by choosing some* $((v, v'), a_j) \in ((V \times V) \times \Sigma)$ *for which* $(v, v') \in \mathcal{E}^k_{a_j}$*. Then new position* $SG^\Sigma_{k+1} = \langle V, \mathcal{E}^{k+1}, v_{k+1}, v_g \rangle$ *is such that* $\mathcal{E}^{k+1}_{a_j} = \mathcal{E}^k_{a_j} \setminus \{(v, v')\}$ *and* $\mathcal{E}^{k+1}_{a_i} = \mathcal{E}^k_{a_i}$ *for all* $i \neq j$*. The game ends if some player cannot make a move or if* $v_k = v_g$*, and Runner wins iff* $v_l = v_g$ *for some round* l*.*

With this definition of the game, it is easy to see that each time Blocker removes an edge from v to v', it is irrelevant for Runner which is the label of the removed edge; what matters for the existence of a winning strategy for him is the number of edges that are left from v to v'.

Observation 1. *Let* $SG^\Sigma = \langle V, \mathcal{E}, v_0, v_g \rangle$ *and* $SG'^\Sigma = \langle V, \mathcal{E}', v_0, v_g \rangle$ *be two labelled Sabotage Games that differ only in the labels of their edges, that is,*

$$\forall (v, v') \in V \times V : |\{\mathcal{E}_{a_i} \mid (v, v') \in \mathcal{E}_{a_i}\}| = |\{\mathcal{E}'_{a_i} \mid (v, v') \in \mathcal{E}'_{a}\}|.$$

Then Runner has a winning strategy in SG^Σ *iff he has a wining strategy in* SG'^Σ*.*

We will now show that the decision problems of deciding whether Runner has a winning strategy in each of the sabotage games SG and SG^Σ are polynomially equivalent. We start by formalizing the problems.

Definition 2.5 (SABOTAGE Problem). *The* sabotage game problem *is as follows.*

INPUT: *A Sabotage Game* $SG = \langle V, E, v_0, v_g \rangle$*.*

QUESTION: *Does Runner have a winning strategy in SG?*

Definition 2.6 (Σ-SABOTAGE Problem). *The* labelled sabotage game problem *is as follows.*

INPUT: *A Sabotage Game on a labelled multi-graph* $SG^\Sigma = \langle V, \mathcal{E}, v_0, v_g \rangle$*.*

QUESTION: *Does Runner have a winning strategy in* SG^Σ*?*

Theorem 1. *SABOTAGE and* Σ*-SABOTAGE are polynomially equivalent.*

Proof. We show that the problems can be polynomially reduced to each other.

- Given $SG = \langle V, E, v_0, v_g \rangle$, let $m := \max E[V \times V]$. Define $f(SG) := \langle V, \mathcal{E}, v_0, v_g \rangle$ where $\mathcal{E} := (\mathcal{E}_1, \dots \mathcal{E}_m)$ and each $\mathcal{E}_i$ is given by $\mathcal{E}_i := \{(v, v') \mid (v, v') \in V \times V,\ E(v, v') \geq i\}$.

We show that Runner has a winning strategy (w.s.) for SG iff he has one for $f(SG)$. The proof is by induction on $n = \sum_{(v,v')\in V\times V} E(v,v')$, which is the total number of edges in the multi-graph. Note that by definition of f, $n = \sum_{i=1}^{i=n} |\mathcal{E}_i|$.

The base case is straightforward since in both games Runner has a w.s. iff $v = v_g$. For the inductive case, from left to right, Runner has a w.s. in $SG = \langle V, E, v_0, v_g\rangle$ with $\sum_{(v,v')\in V\times V} E(v,v') = n+1$. Then there is some $v_1 \in V$ such that $E(v_0, v_1) > 0$ and Runner has a w.s. for all games $SG' = \langle V, E', v_1, v_g\rangle$ that result from Blocker choosing a pair (v,v') with $E(v,v') > 0$. Now, by definition of f, in $f(SG)$, choosing v_1 is also a legal move for Runner in $f(SG)$. By Observation 1, Runner has a w.s. in the game that results from Blocker choosing any $((v,v'),a)$ such that $(v,v') \in \mathcal{E}_a$ iff he has one in the game resulting from Blocker choosing $((v,v'),E(v,v'))$, which is $f(SG')$. Then we can apply the inductive hypothesis.

From right to left, Runner having a w.s. in $f(SG)$ means that he can choose some v_1 with $(v_0,v_1) \in \mathcal{E}_i$ for some $i \leq m$ such that he has a w.s. in all games $f(SG)''$ that result from Blocker's move. Choosing v_1 is also a legal move of Runner in SG. Suppose that Blocker replies by choosing (v,v'). Let us call the resulting game SG'. By assumption and Observation 1, Runner also has a w.s. in the game $f(SG)'$ which is the result from Blocker choosing $((v,v'),E(v,v'))$. Since $f(SG)' = f(SG')$, we can apply the inductive hypothesis.

- Given $SG^\Sigma = \langle V, \mathcal{E}, v, v_g\rangle$ with $\Sigma = \{a_1, \ldots a_m\}$, let $f'(SG^\Sigma) := \langle V, E, v, v_g\rangle$, where $E(v,v') := |\{\mathcal{E}_{a_i} \mid (v,v') \in \mathcal{E}_{a_i}\}|$.

  Showing that Runner has a w.s. in SG^Σ iff he has one in $f(SG^\Sigma)$ is straightforward and can be done by induction on $n := \sum_{a\in\Sigma} |\mathcal{E}_a|$, i.e. the number of edges in $(V,\mathcal{E})$.

Both f and f' are polynomial. This concludes the proof. □

2.2 Sabotage Learning Games

Based on the Sabotage Games framework, we define Sabotage Learning Games as follows.

Definition 2.7 (Sabotage Learning Game (SLG))**.** *A* Sabotage Learning Game *is a labelled sabotage game played by* Learner *(L, taking the role of Runner) and* Teacher *(T, taking the role of Blocker). We distinguish between three different versions, SLGUE, SLGHU and SLGHE, that differ in the winning conditions (given in Table 2).*

In Table 2, we show winning conditions for different versions of SLG. They correspond to different levels of Teacher's helpfulness and Learner's willingness to learn. We can have an *unhelpful* teacher and an *eager* learner (SLGUE), but there is also the possibility of a *helpful* teacher and an *unwilling* learner (SLGHU). The cooperative case corresponds to a *helpful* teacher and an *eager* learner (SLGHE).

Having provided a formal framework for Teacher-Learner interactions by means of SLGs, we now show how Sabotage Modal Logic can be used for reasoning about players' strategic power in these games.

Table 2: Sabotage Learning Games

Game	Winning Condition
SLGUE	Learner wins iff he reaches the goal state, Teacher wins otherwise.
SLGHU	Teacher wins iff Learner reaches the goal state, Learner wins otherwise.
SLGHE	Both players win iff Learner reaches the goal state. Both lose otherwise.

2.3 Sabotage Modal Logic

Sabotage Modal Logic (SML) has been introduced in [10]. Besides the standard modalities, it also contains "transition-deleting" modalities for reasoning about model change that occurs when a transition is removed. To be more precise, we have formulas of the form $\Diamond\!\!\!-\phi$, expressing that it is possible to delete a pair from the accessibility relation such that ϕ holds in the resulting model at the current state.

Definition 2.8 (Sabotage Modal Language [10]). *Let* PROP *be a countable set of propositional letters and let* Σ *be a finite set. Formulas of the language of Sabotage Modal Logic are given by*

$$\phi ::= p \mid \neg\phi \mid \phi \vee \phi \mid \Diamond_a\phi \mid \Diamond\!\!\!-_a\phi$$

with $p \in$ PROP *and* $a \in \Sigma$*. We write* $\Diamond\phi$ *for the finite disjunction* $\bigvee_{a\in\Sigma} \Diamond_a\phi$ *and* $\Diamond\!\!\!-\phi$ *for the finite disjunction* $\bigvee_{a\in\Sigma} \Diamond\!\!\!-_a\phi$.

Definition 2.9 (Sabotage Model [7]). *Given a countable set of propositional letters* PROP *and a finite set* $\Sigma = \{a_1, \ldots, a_n\}$*, a* Sabotage Model *for the Sabotage Modal Logic is a tuple* $M = \langle W, (R_{a_i})_{a_i\in\Sigma}, Val\rangle$ *where* W *is a set of worlds, each* $R_{a_i} \subseteq W \times W$ *is an accessibility relation and* Val : PROP $\to \mathcal{P}(W)$ *is a propositional valuation function. The pair* M, w *with* $w \in W$ *is called a* Pointed Sabotage Model.

In order to define the way formulas of SML are evaluated in Sabotage Models, we first define the model that results from removing an edge.

Definition 2.10. *Let* $M = \langle W, R_{a_1}, \ldots R_{a_n}, Val\rangle$ *be a Sabotage Model. The model* $M^{a_i}_{(w,v)}$ *that results from removing the edge* $(w, v) \in R_{a_i}$ *is defined as*

$$M^{a_i}_{(w,v)} := \langle W, R_{a_1}, \ldots R_{a_{i-1}}, R_{a_i} \setminus \{(v, v')\}, R_{a_{i+1}}, \ldots R_{a_n}, Val\rangle.$$

Definition 2.11. *Given a sabotage model* $M = \langle W, (R_a)_{a\in\Sigma}, Val\rangle$ *and a world* $w \in W$*, atomic propositions, negations, disjunctions and standard modal formulas are interpreted as usual. For the case of "transition-deleting" formulas, we have*

$$M, w \models \Diamond\!\!\!-_a\phi \quad \text{iff} \quad \exists\, w, v \in W : (w, v) \in R_a \;\&\; M^a_{(w,v)}, w \models \phi.$$

Theorem 2 ([7]). *Model checking of SML is PSPACE-complete.*

2.4 Sabotage Learning Games in Sabotage Modal Logic

The Sabotage Modal Logic is useful for reasoning about the players' strategic power in sabotage learning games, since for any given Sabotage Learning Game we can construct a Pointed Sabotage Model $M(G)$ in a straightforward way.

Definition 2.12. *Let $SG^{\Sigma} = \langle V, \mathcal{E}, v_0, v_g \rangle$ be a Sabotage game with $\mathcal{E} = (\mathcal{E}_a)_{a\in\Sigma}$; we define a Pointed Sabotage Model $M(SG^{\Sigma}), v_0$ over the set of atomic propositions* $\texttt{PROP} := \{goal\}$ *as*

$$M(SG^{\Sigma}) := \langle V, \mathcal{E}, Val \rangle,$$

where $Val(goal) := \{v_g\}$.

For each of the winning conditions in Table 2, we can define a formula of SML that is true in a given Pointed Sabotage Model if and only if the corresponding player has a winning strategy in the game represented by the pointed model.

Let us first look at the game SLGUE, in which the learner tries to reach the goal state and the teacher tries to prevent him from doing so. This is the standard sabotage game of [10]. Inductively, we define

$$\gamma_0^{UE} := goal, \qquad \gamma_{n+1}^{UE} := goal \vee \Diamond \boxminus \gamma_n^{UE}.$$

Then we get the following result, which is a variation of Theorem 7 of [7], rephrased for labelled Sabotage Games. We go into it with some detail to show how our "labelled" definition avoids a technical issue present in the original proof.

Theorem 3. *Learner has a winning strategy in SLGUE in the game $SG^{\Sigma} = \langle V, \mathcal{E}^0, v_0, v_g \rangle$ iff $M(SG^{\Sigma}), v_0 \models \gamma_n^{UE}$, for $n := \sum_{a\in\Sigma} |\mathcal{E}_a^0|$ (the number of edges in $(V, \mathcal{E}^0)$).*

Proof. The proof is by induction on n.

Base case

($\Rightarrow$) L having a winning strategy implies that v_0 is the goal state. Thus $M(SG^{\Sigma}), v_0 \models goal$. Hence, $M(SG^{\Sigma}), v_0 \models \gamma_0^{UE}$.

($\Leftarrow$) $M(SG^{\Sigma}), v_0 \models \gamma_0^{UE}$ means that $M(SG^{\Sigma}), v_0 \models goal$, thus v_0 is the goal state. Hence, L can win immediately.

Inductive case

($\Rightarrow$) Let $\sum_{i=1}^{i=|\Sigma|} |\mathcal{E}_{a_i}^0| = n + 1$. Assume that L has a w.s. Then there are two possibilities. (1) v_0 is the goal state. Then $M(SG^{\Sigma}), v_0 \models goal$. Thus, $M(SG^{\Sigma}), v_0 \models \gamma_{n+1}^{UE}$. (2) v_0 is not the goal state and then there is some $v_1 \in V$ such that $(v_0, v_1) \in \mathcal{E}_{a_i}^0$ for some $a_i \in \Sigma$ and no matter what pair $((v, v'), a_j) \in (V \times V) \times \Sigma$ with $(v, v') \in \mathcal{E}_{a_j}^0$ T chooses, L has a w.s. in the resulting game $SG'^{\Sigma}\langle V, \mathcal{E}^1, v_1, v_g \rangle$, with $\mathcal{E}^1 = (\mathcal{E}_{a_1}^0, \ldots \mathcal{E}_{a_{j-1}}^0, \mathcal{E}_{a_j}^0 \setminus \{v, v'\}, \mathcal{E}_{a_{j+1}}^0, \ldots \mathcal{E}_{a_{|\Sigma|}}^0)$. Now, $\sum_{i=1}^{i=|\Sigma|} |\mathcal{E}_{a_i}^1| = n$ and thus by inductive hypothesis, $M(SG'^{\Sigma}), v_1 \models \gamma_n^{UE}$. This implies that $M(SG^{\Sigma}), v_0 \models \Diamond \boxminus \gamma_n$. Thus, $M(SG^{\Sigma}), v_0 \models \gamma_{n+1}$. They key observation is that the model that results from removing an edge from $M(SG^{\Sigma})$ is always a model that results from transforming a labelled sabotage game into a model. With the original definition of a sabotage game, this is not the case: if in the model an edge between v and v' with label k has been removed, the resulting model is not the image of a multi-graph if $k < E(v, v')$. Another way to look at is the following: the multiple edges of the original multi-graph can be seen as implicitly labelled by numbers, and the existence of a edge labelled with k implies the existence of edges labelled with $1, \ldots, k-1$. This property is not preserved when when T removes an arbitrary edge from the model $M(SG)$.

($\Leftarrow$) $M(SG^\Sigma), v_0 \models goal \vee \Diamond \boxminus \gamma_n$ implies that v_0 is the goal state (so L wins immediately) or else there is v_1 accessible from v_0 such that $M(SG^\Sigma), v_1 \models \boxminus \gamma_n$, that is, $M(SG^\Sigma)^{a_i}_{(v,v')}, v_1 \models \boxminus \gamma_n$ for any $((v, v'), a_i) \in (V \times V) \times \Sigma$. By inductive hypothesis, this gives L a w.s. at v_1 in a game that results from removing any edge from the multi-graph of SG^Σ, and hence a w.s. at v_0 in the game SG^Σ. □

Next, consider the game SLGHU, in which the teacher tries to force the learner to reach the goal state. Inductively, define

$$\gamma_0^{HU} := goal, \qquad \gamma_{n+1}^{HU} := goal \vee (\Diamond \top \wedge \Box \diamondminus \gamma_n^{HU}).$$

Now, we can show that this formula corresponds to the existence of a winning strategy for Teacher. Note that in order to win, Teacher has to make sure that Learner does not get stuck before he has reached the goal state. This is why we need the conjunct $\Diamond \top$ in the formula.

Theorem 4. *Teacher has a winning strategy in SLGHU in the game* $SG^\Sigma = \langle V, \mathcal{E}^0, v_0, v_g \rangle$ *iff* $M(SG^\Sigma), v_0 \models \gamma_n^{HU}$, *for* $n := \sum_{a \in \Sigma} |\mathcal{E}_a^0|$.

Proof. The argument is similar to the proof of Theorem 3. □

Let us now consider the third version of SLG in which Teacher and Learner win together if and only if Learner reaches a goal state. The corresponding formula is defined as follows

$$\gamma_0^{HE} := goal, \qquad \gamma_{n+1}^{HE} := goal \vee \Diamond \diamondminus \gamma_n^{HE}.$$

Theorem 5. *Teacher and Learner have a joint winning strategy in SLGHE in the game* $SG^\Sigma = \langle V, \mathcal{E}^0, v_0, v_g \rangle$ *iff* $M(SG^\Sigma), v_0 \models \gamma_n^{HE}$, *for* $n := \sum_{a \in \Sigma} |\mathcal{E}_a^0|$.

Proof. Note that L and T have a joint w.s. iff there is a path from v_0 to v_g. From left to right this is obvious; from right to left, if there is such path, then there is also one without cycles. Then, there is a joint w.s. that follows the path and at each step removes the edge that has just been used. The Theorem follows by observing that γ_n^{HE} express the existence of such a w.s. □

The previous results are summarized in Table 3.

2.5 Complexity of Sabotage Learning Games

Intuitively, some versions of the Sabotage Learning Game are simpler than others. In cases with a helpful teacher and an eager learner, the learning process should be easier than in cases with an unhelpful teacher or a unwilling learner. This is indeed reflected in the computational complexity of deciding in a given game whether the winning condition is satisfied or not.

We have shown that our three winning conditions (Table 3) can be expressed in SML, and Theorem 2 (proved in [7]) tells us that model checking of SML is

Table 3: Winning Conditions for SLG in SML

Game	Winning Condition in SML	Winner
SLGUE	$\gamma_0^{UE} := goal,\ \gamma_{n+1}^{UE} := goal \vee \Diamond\boxminus\gamma_n^{UE}$	Learner
SLGHU	$\gamma_0^{HU} := goal,\ \gamma_{n+1}^{HU} := goal \vee (\Diamond\top \wedge (\Box\diamondminus\gamma_n^{HU}))$	Teacher
SLGHE	$\gamma_0^{HE} := goal,\ \gamma_{n+1}^{HE} := goal \vee \Diamond\diamondminus\gamma_n^{HE}$	Both

PSPACE-complete. This gives us PSPACE upper bounds for the complexity of the problems of deciding whether each one our winning conditions is satisfied in a given game. For two of the winning conditions (SLGUE and SLGHE), we can give more precise complexity results and also give tight lower bounds.

For SLGUE – the standard sabotage game – PSPACE-hardness is shown by reduction from QBF Problem[7].

Theorem 6 ([7])**.** *SLGUE is PSPACE-complete.*

As mentioned above, for SLGHU we obtain a PSPACE upper bound.

Theorem 7. *Solving SLGHU is in PSPACE.*

Proof. Follows from Theorem 2 and Theorem 4. □

It remains to be shown whether SLGHU is also PSPACE-hard. Note that, on the one hand, SLGHU and SLGUE are very similar since the winning positions for Teacher in SLGHU are exactly the same as the ones for Learner in SLGUE (i.e. the positions in which Learner is in a goal state). However, since the powers of the players are essentially different, due to the different nature of their moves (Teacher moves globally, whereas Learner moves locally), a reduction from SLGUE to SLGHU is not straightforward.

Let us now look at SLGHE. This game is of a different nature than the two previous ones since it is cooperative in the sense that one player wins iff the other one does. In this case, a winning strategy is a joint strategy for both players, and does not need to take into account all possible moves of the opponent. This suggest that this version should be less complex than SLGUE and SLGHU.

The following result shows that at least for the comparison of SLGUE and SLGHE, this is indeed the case: in case of an eager learner, learning with a helpful teacher is easier than learning with an unhelpful one. This follows from the fact that the winning condition of SLGHE is satisfied iff the goal vertex is reachable from the initial vertex. Thus, determining whether Teacher and Learner can win SLGHE is equivalent to solving the REACHABILITY (st-CONNECTIVITY) problem, which is known to be nondeterministic logarithmic space (NL)-complete [8].

Theorem 8. *SLGHE is NL-complete.*

Proof. Polynomial equivalence of *SLGHE* and REACHABILITY follows from the argument given in the proof of Theorem 5. □

Table 4: Complexity Results for Sabotage Learning Games

Game	Winning Condition	Complexity
SLGUE	Learner wins iff he reaches the goal state, Teacher wins otherwise	PSPACE-complete.
SLGHU	Teacher wins iff Learner reaches the goal state, Learner wins otherwise.	PSPACE
SLGHE	Both players win iff Learner reaches the goal state. Both loose otherwise.	NL-complete

Table 4 summarizes the complexity results for the different versions of SLG.

The complexity results agree with our intuition when comparing the cooperative version of the sabotage game (SLGHE) with the non-cooperative ones (SLGUE and SLGHU): The easiest way to learn is indeed when Teacher is helpful and Learner — eager.

3 Refined view on teaching: learning algorithms

The view on learning presented above is very general. To give a more refined view, let us go back to the *queries and counterexamples* paradigm (see [1]). In this approach, Learner is an algorithm that embodies a winning strategy in the game of learning (the learning procedure succeeds on *all* possible *true* data). Teacher can significantly influence the learning process by giving counterexamples, and the time needed for learning depends on her choices. Therefore, the game of teaching in such a setting can be formalized in extensive form as presented in Figure 1.

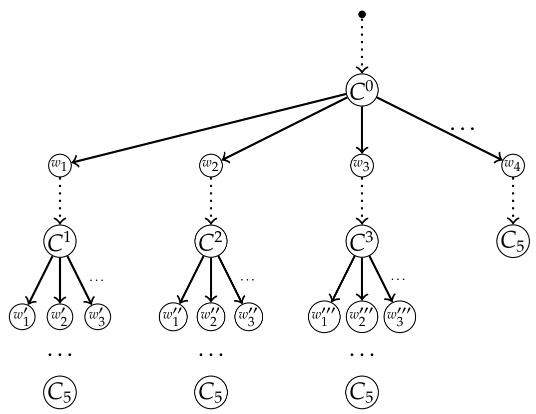

Figure 1: The tree of the teaching game: dotted lines are Learner's moves, which are determined by his algorithm; solid lines are Teacher's moves; w_i are counterexamples given by Teacher; C_i are conjectures made by Learner; C_5 is the correct hypothesis.

There are a number of game-theoretical issues that arise when viewing the run of the learning algorithm as a game. We can for example consider the epistemic status of the players, introduce imperfect information and analyze payoff characteristics. Concerning the payoff characteristics, different classes of teachers such as (un)helpful teachers, we can define corresponding preference relations or payoffs: the *helpful* teacher strictly prefers all shortest paths in the game tree, i.e. the paths in which the learner learns the fastest. The *unhelpful* teacher strictly prefers all the longest paths in the game tree, i.e. the paths in which the learner learns slowly.

We also provide a choice for Learner in this game. Firstly, we can allow that at each step the learner can choose from one or more procedures which are part of one algorithm. Secondly, in the beginning Learner can decide with which of the available algorithms he is going to proceed. Moreover, we can consider also another possibility that involves extending the traditional inductive inference paradigm. Usually, learnability of a class is interpreted as the existence of Learner that learns every element from the class independently of the behavior of Teacher — if we introduce the possibility of non-learnability to the game, we can view learning algorithms as winning strategies for an eager Learner in the learning game. With the possibility of non-learnability, there are also paths in the game tree in which the learner never makes a correct conjecture. In this framework, a *helpful* teacher would also prefer all (shortest) paths ending in a position in which the learner makes a correct conjecture over all the other paths. An *unhelpful* teacher then prefers all the paths in which the learner does not learn over those in which he does learn.

4 Conclusions and further work

We have provided a game theoretical approach to learning that takes into account different levels of cooperativeness between the learner and the teacher. Based on *Sabotage Games*, we have defined *Sabotage Learning Games* with three variations of the winning condition, each of them representing different levels of cooperativeness between Teacher and Learner. Then, we have shown how *Sabotage Modal Logic* can be used to reason about these games and, in particular, we have identified formulas of the language that characterize each of the three winning conditions, providing also complexity results for each one of them.

From the game-theoretical perspective, Sabotage Learning Games can be extended to more general scenarios by relaxing the strict alternation. As we have mentioned before, there is a difference in the "nature" of moves of the payers in this game. Learner's moves can be seen as internal ones while Teacher's moves can be interpreted externally. Due to this asymmetry, each of Learner's moves does not in principle need to be followed by a teacher's move (e.g. the learner can perform several changes of his information state before the teacher can actually make a restriction). An interesting extension of this work could deal with such different possibilities, providing corresponding formulas expressing the winning conditions and exploring their computational complexity.

From the perspective of Formal Learning Theory, several relevant extensions can be done. We have described the learning process as *changes in information states*, without going further into their epistemic and/or doxastic interpretation.

A deeper analysis can give us insights about how the learning process is related to different notions of dynamics of information, such as belief revision or dynamic epistemic logic.

In the introduction we described the concepts of *finite identification* and *identification the limit*. Our work on SLGs is closer to the first one, as we understand learning as the ability to reach an appropriate information state, without taking into account what will happen after such a state has been reached. In particular, we are not concerned with the stability of the resulting belief. *Identification in the limit* extends *finite identification* by looking beyond reachability in order to describe "ongoing behaviour". Fixed-point logics, like the propositional μ-calculus [9, 5], can provide us with tools to express this notion of learnability. In this case, epistemic and doxastic interpretations of learning would involve notions of stable belief and a kind of operational, non-introspective knowledge as a result of the process.

References

[1] Dana Angluin. Learning regular sets from queries and counterexamples. *Information and Computation*, 75(2):87–106, 1987.

[2] H. van Ditmarsch, W. van der Hoek, and B. Kooi. *Dynamic Epistemic Logic*. Springer Netherlands, 2007.

[3] E.M. Gold. Language identification in the limit. *Information and Control*, 10:447–474, 1967.

[4] Sanjay Jain, Daniel Osherson, James S. Royer, and Arun Sharma. *Systems that Learn*. MIT Press, Chicago, 1999.

[5] Dexter Kozen. Results on the propositional mu-calculus. *Theoretical Computer Science*, 27:333–354, 1983.

[6] Christof Löding and Philipp Rohde. Solving the sabotage game is PSPACE-hard. In *Proceedings of the 28th International Symposium on Mathematical Foundations of Computer Science, MFCS '03*, volume 2474 of *LNCS*, pages 531–540. Springer, 2003.

[7] Christof Löding and Philipp Rohde. Solving the sabotage game is PSPACE-hard. Technical report, Aachener Informatik Berichte, Rwth Aachen, 2003.

[8] Christos M. Papadimitriou. *Computational complexity*. Addison-Wesley, MA, 1994.

[9] Dana Scott and W. de Bakker. A theory of programs. Unpublished manuscript, IBM, Vienna, 1969.

[10] Johan van Benthem. An essay on sabotage and obstruction. In *Mechanizing Mathematical Reasoning, Essays in Honor of Jörg H. Siekmann on the Occasion of His 60th Birthday*, pages 268–276, 2005.