On Angels and Demons: Strategic (De)Construction of Dynamic Models

Davide Catta
LIPN, CNRS, Université Sorbonne Paris Nord
Villetaneuse, France
catta@lipn.univ-paris13.fr

Rustam Galimullin
University of Bergen
Bergen, Norway
rustam.galimullin@uib.no

Munyque Mittelmann
LIPN, CNRS, Université Sorbonne Paris Nord
Villetaneuse, France
mittelmann@lipn.univ-paris13.fr

ABSTRACT

In recent years, there has been growing interest in logics that formalise strategic reasoning about agents capable of modifying the structure of a given model. This line of research has been motivated by applications where a modelled system evolves over time, such as communication networks, security protocols, and multi-agent planning. In this paper, we introduce three logics for reasoning about strategies that modify the topology of weighted graphs. In *Strategic Deconstruction Logic*, a destructive agent (the demon) removes edges up to a certain cost. In *Strategic Construction Logic*, a constructive agent (the angel) adds edges within a cost bound. Finally, *Strategic Update Logic* combines both agents, who may cooperate or compete. We study the expressive power of these logics and the complexity of their model checking problems.

KEYWORDS

Strategic Reasoning; Model Change; Expressivity; Model Checking; Modal Logic

ACM Reference Format:
Davide Catta, Rustam Galimullin, and Munyque Mittelmann. 2026. On Angels and Demons: Strategic (De)Construction of Dynamic Models. In *Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), Paphos, Cyprus, May 25 – 29, 2026*, IFAAMAS, 10 pages. https://doi.org/10.65109/XEZZ1033

1 INTRODUCTION

The growing adoption and reliance on autonomous systems in safety-critical applications, such as autonomous vehicles [22] and cybersecurity systems [21], call for reliable verification methods. Model checking [35] is one of the de facto standard approaches to verification of such temporal properties of a given system as safety and reachability. In the context of Multi-Agent Systems (MAS), this approach was extended to capture the interaction and strategic behaviour of autonomous agents. Properties of a system that one would like to verify are usually expressed in temporal or strategic logics, like the Linear Temporal Logic (LTL) [68], Computation Tree Logic (CTL) [34], and Alternating-time Temporal Logic (ATL) [11].

In model checking, a given system is typically represented using a *static* model (e.g., a labelled state-transition model or a concurrent game structure), which describes a fixed set of configurations, or states, of the system and how it transitions between them. Such models are incapable of capturing scenarios in which the system's structure may dynamically change, including modifications caused by agents' actions or the removal of vulnerable components. The assumption of static models limits the applicability of model-checking approaches, as many real-life applications are inherently *dynamic*. One example would be the addition of a new route in a metro system, which may cause bottlenecks at interchange stations. Another example is the implementation of defence strategies to prevent cyberattacks exploiting system vulnerabilities [29].

Motivated by these limitations, there has been a growing interest in logic-based approaches to the specification and verification of dynamic systems in recent years. A notable line of research draws inspiration from the sabotage game [73], which is a reachability game on graphs where one player, called the demon, can delete edges to obstruct the other player, the traveller, from reaching her goal. *Obstruction Logic* (OL) [29] was recently proposed to analyse sabotage-like games on weighted graphs, in which the demon can temporarily disable edges whose weights do not exceed a specified cost. OL, however, only captures a particular type of graph modifications, where removed edges are immediately restored after the traveller's move. Returning to the cybersecurity example, while it can represent the existence of defensive measures that briefly block an attacker's access to a sensitive module, it fails to capture measures such as removing vulnerable execution paths.

Our contribution. We propose three novel logics for reasoning about strategic permanent change of weighted graphs. The first one, *Strategic Deconstruction Logic* (SDL), has a similar flavour to OL: a destructive agent, or the demon, permanently removes edges up to a certain cost. In the cybersecurity setting, permanent removal of edges allows blocking vulnerable paths long before the attacker reaches them, which is impossible in OL. The second logic, *Strategic Construction Logic* (SCL), considers, instead, a constructive agent, or the angel, that can add new edges within a cost bound. Finally, *Strategic Update Logic* (SUL) combines both agents acting concurrently *à la* ATL. In SUL, the angel and demon can cooperate towards the same goal, and their behaviour may even include joint strategies to mimic OL-like strategies. Additionally, SUL captures situations in which they are adversarial, thereby enhancing the strategic dimension of the problem. The main advantage of the proposed logics is that they enable the modelling and verification of dynamic systems in which access control and defence mechanisms can be employed during the execution of the system.

For the new logics, we first demonstrate that each one of them is strictly more expressive than CTL. Second, we show that SDL and SCL are, expressivity-wise, incomparable, and that SUL subsumes

Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), C. Amato, L. Dennis, V. Mascardi, J. Thangarajah (eds.), May 25 – 29, 2026, Paphos, Cyprus. © 2026 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). https://doi.org/10.65109/XEZZ1033

both of them. Next, we study and discuss their nuanced relation with OL. We also investigate the model checking problem for the logics and show that for SDL, SCL, and a fragment of SUL, the problem is PSPACE-complete, whereas for the full SUL it is in EXPSPACE.

2 LOGICS OF ANGELS AND DEMONS

We introduce logics for games played on weighted directed graphs with a serial transition relation. The games involve players who can modify the graph and *the traveller* who moves along the edges of the graph. In the first case, the modifying player is called *the demon*, and she can remove a subset of edges from the graph, possibly empty, up to a certain cost. Then the traveller makes a move in the modified graph. In the second case, the modifying player is called *the angel*[1], and she is able to add edges to the graph up to a certain cost. Finally, we consider the scenario where both the demon and the angel are present and they make concurrent choices of which edges to add and to remove. In all of the logics, we are able to reason about strategies of the modifying players reaching a temporal goal regardless of the moves of the traveller.

2.1 Models

Let $\mathtt{At} = \{p, q, r \ldots\}$ be a countable set of *atoms*.

Definition 1 (Model). A *model* is a tuple $\mathfrak{M} = (S, \to, \mathcal{V}, C)$, where:

- S is a non-empty set of states.
- $\to \subseteq S \times S$ is a serial binary relation over S. We will write $s \to s'$ for $(s, s') \in \to$.
- $\mathcal{V} : \mathtt{At} \to 2^S$ is a valuation function specifying which atoms hold in which states.
- $C : S \times S \to \mathbb{N}^+$ is a cost function assigning to *any* pair of states a positive natural number. Intuitively, this number represents the cost of removing or adding an edge between two given states.

Given a model $\mathfrak{M}$, its set of states will be denoted by $S^{\mathfrak{M}}$, and its set of edges by $\xrightarrow{\mathfrak{M}}$. A *pointed model* is a pair $(\mathfrak{M}, s)$, where $\mathfrak{M}$ is a model and $s \in S^{\mathfrak{M}}$. We write $\mathfrak{M} \setminus A$ for $(S, \to \setminus A, \mathcal{V}, C)$, where $A \subseteq \xrightarrow{\mathfrak{M}}$. And we write $\mathfrak{M} \cup A$ for $(S, \to \cup A, \mathcal{V}, C)$, where $A \subseteq S^{\mathfrak{M}} \times S^{\mathfrak{M}}$.

Given a model $\mathfrak{M}$, we let $True(s) = \{p \in \mathtt{At} | s \in \mathcal{V}(p)\}$ be the set of atoms true in state s. Then, we define the *size* of $\mathfrak{M}$ as $|\mathfrak{M}| = |S| + |\to| + \sum_{s \in S} |True(s)| + \sum_{s,t \in S} C(s,t)$, where integers are encoded in binary. Finally, we call $\mathfrak{M}$ *finite*, if $|\mathfrak{M}|$ is finite.

Remark 1. We use a single cost for adding or removing an edge. Although this could be relaxed without affecting our results, we keep this simpler assumption.

Definition 2 ((De)Construction and Updates). Given two pointed models $(\mathfrak{M}, s)$ and $(\mathfrak{M}', s')$, we say that $(\mathfrak{M}', s')$ is *deconstruction accessible* from $(\mathfrak{M}, s)$ with cost at most n, denoted by $(\mathfrak{M}, s) \overset{n}{\rightsquigarrow} (\mathfrak{M}', s')$, iff $\mathfrak{M}' = \mathfrak{M} \setminus A$ for some $A \subset \xrightarrow{\mathfrak{M}}$, $s \xrightarrow{\mathfrak{M}'} s'$, and $C(A) \leqslant n$, where $C(A)$ is $\sum_{x \in A} C(x)$. We will call $\mathfrak{M}'$ an *n-submodel* of $\mathfrak{M}$.

We say that $(\mathfrak{M}', s')$ is *construction accessible* from $(\mathfrak{M}, s)$ with cost at most n, denoted $(\mathfrak{M}, s) \overset{n}{\triangleright} (\mathfrak{M}', s')$, if and only if $\mathfrak{M}' = \mathfrak{M} \cup A$ for some $A \subseteq ((S^{\mathfrak{M}} \times S^{\mathfrak{M}}) \setminus \to^{\mathfrak{M}})$, $C(A) \leqslant n$, and $s \xrightarrow{\mathfrak{M}'} s'$. We will also call $\mathfrak{M}'$ an *n-supermodel* of $\mathfrak{M}$.

Given a model $\mathfrak{M}$, its n-supermodel $\mathfrak{M}_1$ obtained by adding the set of edges A, and m-submodel $\mathfrak{M}_2$ obtained by removing the set of edges B, we will denote by $\mathfrak{M}_1 \star \mathfrak{M}_2 = (\mathfrak{M} \setminus B) \cup A$ the resulting model after edges B were removed and edges A were added. We will call $\mathfrak{M}_1 \star \mathfrak{M}_2$ an *n-m-update*. Note that $\mathfrak{M}_1 \star \mathfrak{M}_2$ is well-defined as the sets of edges A and B are disjoint.

For two pointed models $(\mathfrak{M}, s)$ and $(\mathfrak{M}', s')$, we say that $(\mathfrak{M}', s')$ is *update accessible* from $(\mathfrak{M}, s)$ with costs n and m, denoted by $(\mathfrak{M}, s) \overset{n,m}{\Rightarrow} (\mathfrak{M}', s')$, iff $\mathfrak{M}'$ is an n-m-update of $\mathfrak{M}$ and $s \xrightarrow{\mathfrak{M}'} s'$.

The agent making the move $s \xrightarrow{\mathfrak{M}'} s'$ in any of the three contexts of deconstruction, construction, or update accessibility, is called the *traveller*.

Intuitively, a model $\mathfrak{M}'$ is deconstruction accessible (i.e., a submodel) with cost at most n from $\mathfrak{M}$ if we can obtain $\mathfrak{M}'$ by removing edges from $\mathfrak{M}$ with the total cost of up to n. The n-supermodel $\mathfrak{M}'$ is obtained from $\mathfrak{M}$ by adding a *new* set of edges that are not already present in $\mathfrak{M}$ and whose total cost does not exceed n. Observe that since all the notions of accessibility above are defined between two (pointed) models, all the operations preserve seriality.

2.2 Strategic Deconstruction Logic

We call an agent who is capable of removing edges *the demon*. As opposed to sabotage games [73], we are interested not merely in one-step actions of the demon, but rather in her *strategies* that ensure some property against all moves of the traveller.

Definition 3 (Demonic Strategy). Let π be a (countably) infinite sequence of pointed models. Then π is a *decreasing model path with cost n* iff for every $i \geqslant 0$ we have that $\pi(i) \overset{n}{\rightsquigarrow} \pi(i+1)$. Note that such a sequence is indeed infinite, as we can always take a 0-submodel of the given model, i.e., keep the current model intact. We will denote by $\pi^{\mathfrak{M}}$ the corresponding sequence of (non-pointed) models in π.

A *demonic strategy* is a function $\mathfrak{S}$ that, given as an input a pointed model $(\mathfrak{M}, s)$, outputs a set of edges $A \subset \xrightarrow{\mathfrak{M}}$.

A decreasing model path π is *compatible* with a demonic strategy $\mathfrak{S}$ iff for all $i \in \mathbb{N}$, we have that $\mathfrak{S}(\pi(i)) = A$ implies $\pi^{\mathfrak{M}}(i+1) = \pi^{\mathfrak{M}}(i) \setminus A$. We let $Out(\mathfrak{S}, (\mathfrak{M}, s))$ denote the set of paths that are compatible with $\mathfrak{S}$ and whose first component is $(\mathfrak{M}, s)$. A demonic strategy $\mathfrak{S}$ has cost n (in this case, it will be called *n-strategy* $\mathfrak{S}$) iff each model path that is compatible with the strategy has cost n.

Definition 4 (Strategic Deconstruction Logic). State (φ) and path (ψ) formulae are defined by mutual recursion:

$$\varphi := p \mid \neg\varphi \mid (\varphi \wedge \varphi) \mid \langle \text{⛤}^n \rangle \psi \qquad \psi := \mathsf{X}\,\varphi \mid \varphi\,\mathsf{U}\,\varphi \mid \varphi\,\mathsf{R}\,\varphi$$

where $p \in \mathtt{At}$ and $n \in \mathbb{N}$. Formulae of *Strategic Deconstruction Logic* (SDL) are all and only the state formulae. Constructs $\langle \text{⛤}^n \rangle \psi$ are read as 'there is a demonic n-strategy such that for all moves of the

[1]The name of the edge-removing player, the demon, comes from the literature on sabotage games [15]. Hence, as the counterpart to the demon, we call the edge-adding agent the angel. To alleviate the connotations coming with such names, we could have called the demon the *remover*, or *saboteur*, and the angel the *constructor*. Indeed, in the setting of adding and removing edges, the demon can be viewed as benevolent if she removes edges to bad or undesirable states, and the angel can be considered malevolent if she adds edges to undesirable states. Perhaps, to play on the metaphor, we could view actions of celestial beings as giving or taking away options, and it is up to a mortal, the traveller, to make her own choices.

traveller, ψ holds.' Temporal modalities $\mathsf{X}\,\varphi$ are read as 'φ is true in the neXt step', modalities $\varphi\,\mathsf{U}\,\psi$ mean 'φ holds Until ψ is true', and modalities $\varphi\,\mathsf{R}\,\psi$ are read as 'truth of φ Releases the requirement for the truth of ψ'.

We define the operator for *sometime* as $\mathsf{F}\varphi := \top\,\mathsf{U}\,\varphi$, and *always* as $\mathsf{G}\varphi := \bot\,\mathsf{R}\,\varphi$. The dual of the demonic operator is defined as $[\circledast^n]\psi := \neg\langle\circledast^n\rangle\neg\psi$, and is read as 'for all demonic n-strategies, there is a move of the traveller such that ψ holds.' Other propositional connectives, like implication, are defined as usual, and the conventions for removing parentheses hold. We will sometimes call n in strategic demonic operators $\langle\circledast^n\rangle\psi$ the *resource bound*.

Given a formula φ of SDL, its *size*, denoted by $|\varphi|$, is the number of symbols in φ with integers encoded in binary.

Definition 5 (SDL Semantics). Let $(\mathfrak{M}, s)$ be a pointed model and φ be a formula of SDL. We define the *satisfaction relation* $(\mathfrak{M}, s) \models \varphi$ by the induction on φ omitting Boolean cases for brevity:

$(\mathfrak{M}, s) \models \langle\circledast^n\rangle\psi$ iff there is an n-strategy $\mathfrak{S}$ s.t. for all $\pi \in Out(\mathfrak{S}, (\mathfrak{M}, s))$ we have $\pi \models \psi$

Given a path formula ψ and a path π, the satisfaction relation $\pi \models \psi$ is defined by the induction on ψ:

$\pi \models \mathsf{X}\,\varphi_1$ iff $\pi(1) \models \varphi_1$

$\pi \models \varphi_1\,\mathsf{U}\,\varphi_2$ iff there is $j \in \mathbb{N}$ such that $\pi(j) \models \varphi_2$ and $\pi(i) \models \varphi_1$ for each $0 \leqslant i < j$

$\pi \models \varphi_1\,\mathsf{R}\,\varphi_2$ iff either $\pi(j) \models \varphi_2$ for each $j \in \mathbb{N}$ or there is $k \in \mathbb{N}$ such that $\pi(k) \models \varphi_1$ and $\pi(i) \models \varphi_2$ for all $0 \leqslant i \leqslant k$

We can define the standard modal box and diamond modalities using demonic strategies with 0 resources as follows: $\Box\varphi := \langle\circledast^0\rangle\mathsf{X}\,\varphi$ and $\Diamond\varphi := [\circledast^0]\mathsf{X}\,\varphi$. It is easy to verify that the box and diamond have exactly the intended semantics, as the only strategy the demon can play keeps the model intact.

Example 1. (Access control) Let model $\mathfrak{M}_1$ in Figure 1 depict a computational system managed by a security engineer. The system user can be seen as the traveller, whereas the security engineer controls her access (i.e., the demon). Also, let *error*, *admin*, and *server* be atoms. State s_0 represents an authentication stage and leads to state s_1 if the user enters an incorrect password and to s_2 otherwise. Node s_1 is a failure state and prevents the user from accessing any other state of the system, including going back to s_0. States s_2 and s_3 are system modules within the server, with s_3 representing an administrative module. The valuation is as follows: $\mathcal{V}(s_0) = \emptyset$, $\mathcal{V}(s_1) = \{error\}$, $\mathcal{V}(s_2) = \{server\}$, and $\mathcal{V}(s_3) = \{server, admin\}$.

Assume that the user is trusted to access the system, but not the admin module. In this case, the security engineer has a strategy with cost 2 that removes the transitions from s_2 to s_3 and s_0 to s_3 and prevents global access to the admin state from other states. The order of removing transitions depends on the current state. The resulting model $\mathfrak{M}_2$ after two steps of the game is shown in Figure 1. We can thus see that $(\mathfrak{M}_1, s) \models \langle\circledast^2\rangle\mathsf{G}\,\neg\,admin$, for any $s \neq s_3$.

2.3 Strategic Construction Logic

Now we introduce Strategic Construction Logic that, in some sense, is dual to SDL. Whilst in SDL the demon can remove edges, in Strategic Construction Logic *the angel* can strategically add edges.

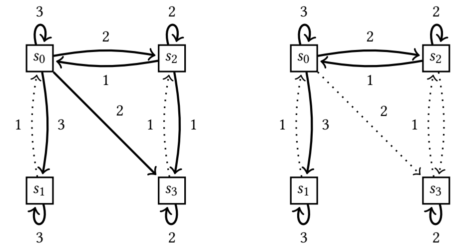

Figure 1: Models $\mathfrak{M}_1$ (left) and $\mathfrak{M}_2$ (right). Arrows with solid lines represent live transitions in the system, whereas dotted lines depict possible new transitions within cost 3.

Definition 6 (Angelic Strategy). Given a countably infinite sequence π of pointed models, π is an *increasing model path with cost* n iff for every $i \geqslant 0$ we have that $\pi(i) \overset{n}{\triangleright} \pi(i+1)$. As in Definition 3, we write $\pi^{\mathfrak{M}}$ to denote the corresponding sequence of models.

An *angelic strategy* is a function $\mathcal{S}$ that, given a pointed model $(\mathfrak{M}, s)$, returns a set of edges $A \subseteq ((S^{\mathfrak{M}} \times S^{\mathfrak{M}}) \setminus \rightarrow^{\mathfrak{M}})$.

An increasing model path π is compatible with $\mathcal{S}$ iff for all $i \in \mathbb{N}$, we have that $\mathcal{S}(\pi(i)) = A$ implies $\pi^{\mathfrak{M}}(i+1) = \pi^{\mathfrak{M}}(i) \cup A$. An angelic strategy $\mathcal{S}$ will be called an *n-strategy* iff every increasing model path that is compatible with the strategy has cost n. We let $Out(\mathcal{S}, (\mathfrak{M}, s))$ denote the set of increasing model paths starting at $(\mathfrak{M}, s)$ that are compatible with $\mathcal{S}$.

Definition 7 (Strategic Construction Logic). Formulae of *Strategic Construction Logic* (SCL) are defined similarly to those of Strategic Deconstruction Logic in Definition 4, with the difference that state formulae of the form $\langle\circledast^n\rangle\psi$ are replaced by state formulae for strategic angelic operators $\langle \propto\!\bigcirc\!\propto^n\rangle\psi$. Constructs $\langle \propto\!\bigcirc\!\propto^n\rangle\psi$ are read as 'there is an angelic n-strategy such that for all moves of the traveller, ψ holds'. The dual of the operator is defined as $[\propto\!\bigcirc\!\propto^n]\psi := \neg\langle \propto\!\bigcirc\!\propto^n\rangle\neg\psi$.

The semantics of SCL is defined similarly to the semantics of SDL (Definition 5), except for the case of the strategic operator.

Definition 8 (SCL Semantics). Let $(\mathfrak{M}, s)$ be a pointed model.

$(\mathfrak{M}, s) \models \langle \propto\!\bigcirc\!\propto^n\rangle\psi$ iff there is an n-strategy $\mathcal{S}$ s.t. for all $\pi \in Out(\mathcal{S}, (\mathfrak{M}, s))$ we have $\pi \models \psi$

Similarly to SDL, we can define the standard modal box and diamond in SCL as $\Box\varphi := \langle \propto\!\bigcirc\!\propto^0\rangle\mathsf{X}\,\varphi$ and $\Diamond\varphi := [\propto\!\bigcirc\!\propto^0]\mathsf{X}\,\varphi$.

Example 2. (Access control, cont.) Let us consider model $\mathfrak{M}_1$ from Example 1 again. Once the user enters the failure state s_1 by, e.g., typing a wrong password, she remains permanently blocked there. Assume now that we want to allow the user to exit the failure state in one step. Adding a transition from s_1 to s_0, which has a cost of 1, would allow the user to retry entering the server module. The resulting model $\mathfrak{M}_3$ after applying this angelic strategy is shown in Figure 2. Notice that the angel does not ensure the traveller will move out of the error state, i.e., $(\mathfrak{M}_3, s_1) \not\models \langle \propto\!\bigcirc\!\propto^1\rangle\mathsf{X}\,\neg\,error$. This is because angelic strategies can create new possibilities for the traveller, but they cannot force her to take them. However, we can easily check that $(\mathfrak{M}_1, s) \models \langle \propto\!\bigcirc\!\propto^1\rangle\mathsf{G}(error \rightarrow \Diamond\,\neg\,error)$ holds for any state s, that is, the angel has a strategy to ensure that every time the traveller enters a failure state, she can leave it in one step.

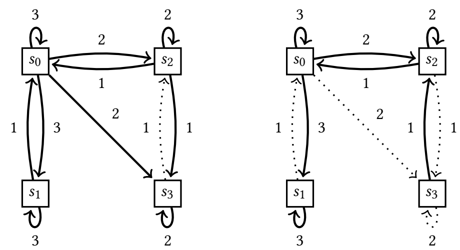

Figure 2: Model $\mathfrak{M}_3$ (left) and $\mathfrak{M}_4$ (right), obtained from changing $\mathfrak{M}_1$ and $\mathfrak{M}_2$ (Figure 1), respectively.

Indeed, it is enough for the angel to restore the transition from s_1 and s_0, resulting in model $\mathfrak{M}_3$, to satisfy $\mathsf{G}(error \rightarrow \Diamond \neg error)$.

2.4 Strategic Update Logic

Having defined separate logics for the demon and the angel, we now combine the two, allowing them to cooperate or compete with one another. In this, we take inspiration from logics for MAS (like alternating-time temporal logic (ATL) [11] and coalition logic (CL) [66]), and assume that the angel and the demon execute their actions *concurrently*. Our new modalities, which are inspired by those of ATL and CL, $\langle\!\![C]\!\!\rangle\varphi$, mean 'there is a joint strategy of agents in coalition C such that no matter what agents outside of the coalition do, φ holds'. In our case, agents with strategies are the angel and the demon.

Definition 9 (Update Model Paths)**.** For a countably infinite sequence π of pointed models, we say that π is an *update model path with costs n and m* iff for all $i \in \mathbb{N}$ we have that $\pi(i) \overset{n,m}{\Rightarrow} \pi(i+1)$. Given an angelic n-strategy $\mathcal{S}$ and a demonic m-strategy $\mathfrak{S}$, an update model path π with costs n and m is compatible with the strategies iff for all $i \in \mathbb{N}$, we have $\pi^{\mathfrak{M}}(i+1) = (\pi^{\mathfrak{M}}(i) \setminus \mathfrak{S}(\pi(i))) \cup \mathcal{S}(\pi(i))$. Finally, $Out(\mathcal{S}, \mathfrak{S}, (\mathfrak{M}, s))$ is the set of update model paths starting at $(\mathfrak{M}, s)$ and compatible with strategies $\mathcal{S}$ and $\mathfrak{S}$.

Definition 10 (Strategic Update Logic)**.** *Strategic Update Logic* (SUL) is defined similarly to SDL and SCL, with the difference that modalities for strategic demonic or angelic operators are substituted with formulas $\langle\!\![C]\!\!\rangle^{n,m}\psi$, where $n, m \geqslant 0$, and $C \subseteq \{\text{⛧}, \text{∝◯∝}\}$. In particular, the formula can have four variations: 1) $C = \{\text{⛧}\}$, and we will write $\langle\!\![\text{⛧}^n]\!\!\rangle^m\psi$ meaning that 'there is a demonic n-strategy such that for all angelic m-strategies and all moves of the traveller, ψ holds'; 2) $C = \{\text{∝◯∝}\}$, and we will write $\langle\!\![\text{∝◯∝}^n]\!\!\rangle^m\psi$ with the meaning as in the previous case with demonic and angelic strategies swapped; 3) $C = \{\text{⛧}, \text{∝◯∝}\}$, which we will write as $\langle\!\![\text{∝◯∝}^n, \text{⛧}^m]\!\!\rangle\psi$ meaning 'there is an angelic n-strategy and a demonic m-strategy such that for all moves of the traveller, ψ holds'; and, finally, 4) $C = \emptyset$, denoted as $\langle\!\![\emptyset]\!\!\rangle^{n,m}\psi$, with the meaning as in the previous case with existential quantifiers swapped for universal ones. As usual, we will denote the dual of the combined strategic operator as $[\langle C \rangle]^{n,m}\psi := \neg\langle\!\![C]\!\!\rangle^{n,m}\neg\psi$.

Definition 11 (SUL Semantics)**.** Let $(\mathfrak{M}, s)$ be a pointed model. We present only cases of $\langle\!\![\text{∝◯∝}^n]\!\!\rangle^m\psi$ and $\langle\!\![\text{∝◯∝}^n, \text{⛧}^m]\!\!\rangle\psi$, and the semantics for $\langle\!\![\text{⛧}^n]\!\!\rangle^m$ and $\langle\!\![\emptyset]\!\!\rangle^{n,m}\psi$ are defined analogously by swapping demonic and angelic strategies, as well as quantifiers.

$(\mathfrak{M}, s) \models \langle\!\![\text{∝◯∝}^n]\!\!\rangle^m\psi$ iff there is an n-strategy $\mathcal{S}$ s.t. for all m-strategies $\mathfrak{S}$ and for all $\pi \in Out(\mathcal{S}, \mathfrak{S}, (\mathfrak{M}, s))$ we have that $\pi \models \psi$

$(\mathfrak{M}, s) \models \langle\!\![\text{∝◯∝}^n, \text{⛧}^m]\!\!\rangle\psi$ iff there is an n-strategy $\mathcal{S}$ and there is an m-strategy $\mathfrak{S}$ s.t. for all $\pi \in Out(\mathcal{S}, \mathfrak{S}, (\mathfrak{M}, s))$ we have that $\pi \models \psi$

Example 3. (Access control, cont.) Cooperation between the angel and demon allows for a dynamic and synchronised access control to the system. Let us consider model $\mathfrak{M}_2$ in Figure 1. The security engineer, the demon, on her own can not make the user leave the admin module, i.e., $(\mathfrak{M}_2, s_3) \not\models \langle\!\![\text{⛧}^2]\!\!\rangle^2\mathsf{F}\neg admin$ as the angel can play the strategy of not modifying the model. However, this goal can be achieved when both the demon and the angel cooperate. Particularly, in the first step, the angel creates transition $s_3 \longrightarrow s_2$, and the demon chooses a 0-submodel (i.e., does nothing), and in the next step of the game the demon removes the self-loop at s_3. This strategy results in model $\mathfrak{M}_4$ in Figure 2. Thus, we have that $(\mathfrak{M}_2, s_3) \models \langle\!\![\text{∝◯∝}^2, \text{⛧}^2]\!\!\rangle\mathsf{F}\neg admin$, as the only thing the user can do once the model is updated to $(\mathfrak{M}_4, s_3)$ is to move to state s_2.

With SUL we can also capture adversarial interactions. For instance, the angel can collaborate with a malicious attacker in $(\mathfrak{M}_2, s_1)$ to enable access to unauthorized states, creating vulnerabilities in the system. For example, $(\mathfrak{M}_2, s_1) \not\models \langle\!\![\text{⛧}^2]\!\!\rangle^2\mathsf{G}\neg admin$. Indeed, to prevent the attacker aided by the angel from reaching state s_3 from s_1, the demon can remove the transition $s_0 \longrightarrow s_2$, while at the same time the angel builds a bridge $s_1 \longrightarrow s_0$. Now, if the attacker is in the state s_0, the angel can, for example, add a bridge $s_0 \longrightarrow s_3$ and no matter what the demon does at the same time, the attacker will get access to the admin state. However, the security engineer can prevent this attack if the angel has fewer resources. We can easily check that $(\mathfrak{M}_2, s_1) \models \langle\!\![\text{⛧}^2]\!\!\rangle^1\mathsf{G}\neg admin$. Indeed, the demon can remove the transition from s_0 to s_2, and no matter what the angel does with 1 resource, she cannot restore a path to state s_3.

3 EXPRESSIVITY

In this section, we compare our new logics to each other as well as to established logics in the literature.

Definition 12 (Expressivity)**.** Let $\mathcal{L}_1$ and $\mathcal{L}_2$ be two languages, and let $\varphi \in \mathcal{L}_1$ and $\psi \in \mathcal{L}_2$. We say that φ and ψ are *equivalent*, when for all models $(\mathfrak{M}, s)$: $(\mathfrak{M}, s) \models \varphi$ if and only if $(\mathfrak{M}, s) \models \psi$.

If for every $\varphi \in \mathcal{L}_1$ there is an equivalent $\psi \in \mathcal{L}_2$, we write $\mathcal{L}_1 \preccurlyeq \mathcal{L}_2$ and say that $\mathcal{L}_2$ is *at least as expressive as* $\mathcal{L}_1$. We write $\mathcal{L}_1 \prec \mathcal{L}_2$ iff $\mathcal{L}_1 \preccurlyeq \mathcal{L}_2$ and $\mathcal{L}_2 \not\preccurlyeq \mathcal{L}_1$, and we say that $\mathcal{L}_2$ is *strictly more expressive than* $\mathcal{L}_1$. Finally, if $\mathcal{L}_1 \not\preccurlyeq \mathcal{L}_2$ and $\mathcal{L}_2 \not\preccurlyeq \mathcal{L}_1$, we say that $\mathcal{L}_1$ and $\mathcal{L}_2$ are *incomparable* and write $\mathcal{L}_1 \not\approx \mathcal{L}_2$.

Computation Tree Logic. We start by showing that all of SDL, SCL, and SUL are strictly more expressive than the classic *Computation Tree Logic* (CTL) [34]. For the proof that our logics are at least as expressive s CTL, we argue, similarly to the argument for OL [29], that CTL is a fragment of all of SDL, SCL, and SUL. To show that there are properties that our logics can express while CTL cannot, we use the model changes.

Theorem 1. CTL $\prec$ SDL, CTL $\prec$ SCL, and CTL $\prec$ SUL.

Proof. To see that CTL is a fragment of our logics it is enough to define a truth-preserving translation function t from CTL to our logics. We omit Boolean cases as they are immediate. Now, to deal with CTL path quantifiers, we can employ demonic and angelic strategies over 0 resources. In particular, $t(\mathsf{AX}\varphi) = \heartsuit \mathsf{X}\, t(\varphi)$ and $t(\mathsf{A}(\varphi \,\mathsf{U}\, \psi)) = \heartsuit(t(\varphi) \,\mathsf{U}\, t(\psi))$, where $\heartsuit \in \{\langle \circledast^0 \rangle, \langle \circ\!\bigcirc\!\circ^0 \rangle, \langle\!\![\circ\!\bigcirc\!\circ^0, \circledast^0]\!\!\rangle\}$ depending on the logic in question. It is immediate, by the definition of the semantics, that such a recursive translation is truth-preserving.

Now, for each of SDL, SCL, and SUL, we show that there are models that they can distinguish and CTL cannot. First, we start with SDL. Consider an SDL formula $\langle \circledast^1 \rangle \mathsf{X}\, p$, and assume for contradiction that there is an equivalent formula φ of CTL. Then, consider models $\mathfrak{M}_1$ and $\mathfrak{M}_2$ depicted in Figure 3.

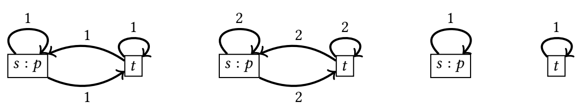

Figure 3: Models $\mathfrak{M}_1$ (left), $\mathfrak{M}_2$ (middle), and $\mathfrak{M}_3$ (right). Atom p is true in states s. Cost of all possible edges in $\mathfrak{M}_3$ is 1.

The two models are isomorphic with the only difference that the cost of every edge in $\mathfrak{M}_1$ is 1, and the cost of every edge in $\mathfrak{M}_2$ is 2. Since in CTL we do not have access to the cost of edges, $(\mathfrak{M}_1, s) \models \varphi$ if and only if $(\mathfrak{M}_2, s) \models \varphi$.

It is also easy to verify that $(\mathfrak{M}_1, s) \models \langle \circledast^1 \rangle \mathsf{X}\, p$. Indeed, the demon can remove the edge $s \to t$, and thus the traveller will never reach state t, where p is false. Also, it is immediate that $(\mathfrak{M}_2, s) \not\models \langle \circledast^1 \rangle \mathsf{X}\, \neg p$ because the cost of every edge in the model is 2, and there is nothing the demon can do to modify the model.

We now turn to SCL and SUL, and consider the following two formulas: $\langle \circ\!\bigcirc\!\circ^1 \rangle \mathsf{X}\, [\circ\!\bigcirc\!\circ^0] \mathsf{X}\, \neg p$ of SCL and $\langle\!\![\circ\!\bigcirc\!\circ^1]\!\!\rangle^0 \mathsf{X}\, [\langle \circ\!\bigcirc\!\circ^0, \circledast^0 \rangle] \mathsf{X}\, \neg p$ of SUL. These formulas mean that the angel can add an arrow such that in the new updated model, the traveller can reach a $\neg p$-state. Assume, for the sake of contradiction, that there is some equivalent formula φ of CTL for each of them. Take model $\mathfrak{M}_3$ in Figure 3 and model $\mathfrak{M}_4$, which is like $\mathfrak{M}_3$ but contains the single state s.

Model $\mathfrak{M}_4$ is just a single state s with a reflexive arrow. Model $\mathfrak{M}_3$ is the state s as well as the state t, both having only reflexive edges. It is immediate that $(\mathfrak{M}_3, s) \models \varphi$ if and only if $(\mathfrak{M}_4, s) \models \varphi$.

At the same time, we have that $(\mathfrak{M}_4, s) \not\models \langle \circ\!\bigcirc\!\circ^1 \rangle \mathsf{X}\, [\circ\!\bigcirc\!\circ^0] \mathsf{X}\, \neg p$ and $(\mathfrak{M}_4, s) \not\models \langle\!\![\circ\!\bigcirc\!\circ^1]\!\!\rangle^0 \mathsf{X}\, [\langle \circ\!\bigcirc\!\circ^0, \circledast^0 \rangle] \mathsf{X}\, \neg p$ as there is simply no state in model $\mathfrak{M}_4$ satisfying $\neg p$. On the other hand, $(\mathfrak{M}_3, s) \models \langle \circ\!\bigcirc\!\circ^1 \rangle \mathsf{X}\, [\circ\!\bigcirc\!\circ^0] \mathsf{X}\, \neg p$, as the angel can add the edge $s \to t$ to make the $\neg p$-state t accessible for the traveller. In particular, after the angel adds the edge $s \to t$, we check all next-time paths the traveller can take. She can either stay in s or move to t. In both cases, $[\circ\!\bigcirc\!\circ^0] \mathsf{X}\, \neg p$ is satisfied, as with 0 resources the angel cannot modify the model anymore, and there a next-time path for the traveller to reach the $\neg p$-state t. A similar reasoning can be used to see that $(\mathfrak{M}_3, s) \models \langle\!\![\circ\!\bigcirc\!\circ^1]\!\!\rangle^0 \mathsf{X}\, [\langle \circ\!\bigcirc\!\circ^0, \circledast^0 \rangle] \mathsf{X}\, \neg p$. □

Angels and Demons. We show that once pitched against one another, the new logics of angels and demons are indeed different. In particular, we prove that SDL and SCL are incomparable and that SUL is strictly more expressive than either one.

Theorem 2. SDL $\not\approx$ SCL, SDL $\prec$ SUL, and SCL $\prec$ SUL.

Proof. We can reuse the arguments from the proof of Theorem 1. To see that SDL $\not\preccurlyeq$ SCL, we recall that models $\mathfrak{M}_1$ and $\mathfrak{M}_2$ are distinguishable by an SDL formula. That no SCL formula can distinguish the two models follows from the fact that the models differ only in the costs of their edges, and since the relations for both models are universal, the angel cannot modify the model.

To show that SCL $\not\preccurlyeq$ SDL, we recall models $\mathfrak{M}_3$ and $\mathfrak{M}_4$ from the proof of Theorem 1 that are distinguishable by an SCL formula. That no SDL formula can distinguish $(\mathfrak{M}_3, s)$ and $(\mathfrak{M}_4, s)$ is immediate by the semantics of the demonic operator (the demon cannot remove any further edges). Hence, SDL $\not\approx$ SCL.

Finally, observe that both SDL and SCL are fragments of SUL via a translation $t(\langle \circledast^n \rangle \varphi) = \langle\!\![\circ\!\bigcirc\!\circ^0, \circledast^n]\!\!\rangle t(\varphi)$ and $t(\langle \circ\!\bigcirc\!\circ^n \rangle \varphi) = \langle\!\![\circ\!\bigcirc\!\circ^n, \circledast^0]\!\!\rangle t(\varphi)$, and therefore SDL $\preccurlyeq$ SUL and SCL $\preccurlyeq$ SUL. Hence, we also have that SCL $\not\preccurlyeq$ SDL implies SUL $\not\preccurlyeq$ SDL, and SDL $\not\preccurlyeq$ SCL implies SUL $\not\preccurlyeq$ SCL. Thereby, we conclude that SDL $\prec$ SUL, and SCL $\prec$ SUL. □

Relation to Obstruction Logic. The semantics of the strategic operators $\langle \dagger^n \rangle \psi$ and $[\dagger^n]\psi$ of *Obstruction Logic* (OL) [29] is similar to the semantics of the SDL modalities $\langle \circledast^n \rangle \psi$ and $[\circledast^n]\psi$ with the crucial difference that after the demon disables some edges and the traveller makes a move, the edges in OL are restored. Hence, in OL, the changes in the given model are *not permanent*. Notice that in the proof of SDL $\not\preccurlyeq$ SCL in Theorem 2, we used only next-time temporal modalities in the SDL formula. Hence, we can use the same argument and the same pair of models, $\mathfrak{M}_1$ and $\mathfrak{M}_2$, to show that OL $\not\preccurlyeq$ SCL. For the other direction, i.e., SCL $\not\preccurlyeq$ OL, the argument is similar to the one for SCL $\not\preccurlyeq$ SDL that uses models $\mathfrak{M}_3$ and $\mathfrak{M}_4$. The same argument implies that SUL $\not\preccurlyeq$ OL.

Theorem 3. OL $\not\approx$ SCL and SUL $\not\preccurlyeq$ OL.

The relation between OL and SDL is more nuanced. In the proof of SDL $\not\preccurlyeq$ OL, we use the fact that the model change in SDL is permanent and hence we can 'remember' the removed edges, which is impossible in OL, where edges are restored before each new action of the demon.

Theorem 4. SDL $\not\preccurlyeq$ OL.

Proof. We provide the idea behind the proof, and the full proof can be found in [27]. Consider an SDL formula $\langle \circledast^1 \rangle \mathsf{F} p$, and assume, for the sake of contradiction, that there is an equivalent formula φ of OL. Since formulas of OL are finite, and each OL modality has a finite resource bound, we can assume that n is the greatest $n \in \mathbb{N}$ appearing in φ. Now consider models $\mathfrak{M}_{n+2}$ and $\mathfrak{M}_{n+3}$ in Figure 4. The models are quite similar, with the only difference being that $\mathfrak{M}_{n+2}$ has one less of t-states. For both models, p is true only in state t_1.

To show that $(\mathfrak{M}_{n+2}, s_1) \models \langle \circledast^1 \rangle \mathsf{F} p$ and $(\mathfrak{M}_{n+3}, s_1) \not\models \langle \circledast^1 \rangle \mathsf{F} p$, we argue that, while the traveller moves along the s-states, at each step in $\mathfrak{M}_{n+2}$ the demon can remove one of the $s_{n+1} \to t_i$ edges

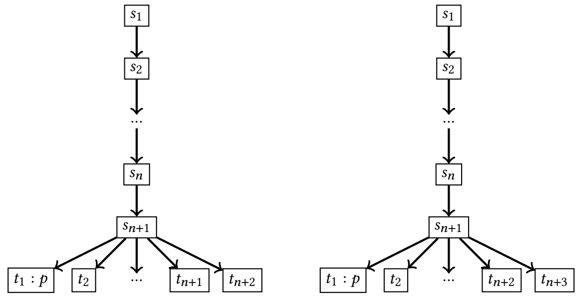

Figure 4: Models $\mathfrak{M}_{n+2}$ (left) and $\mathfrak{M}_{n+3}$ (right) with reflexive arrows for t-states omitted. The cost of all edges is 1.

with $i > 1$. Such a gradual removal results in the situation, when in state s_{n+1} the traveller has only one choice, namely to enter state t_1 thus satisfying $\mathsf{F}p$. In $\mathfrak{M}_{n+3}$, we have one additional t-state, and therefore the demon does not have enough steps to ensure that only the p-state t_1 is available. Hence, the traveller can enter a t-state that does not satisfy p and violate $\mathsf{F}p$. Then we argue that φ cannot distinguish the models, as in OL edges are restored after each game step, and hence the demon should make a move in state s_{n+1} in one of the models that is not replicable in the other model. This is impossible, since the resource bound n is too low. □

The Expressivity Landscape. An overview of the expressivity results is presented in Figure 5.

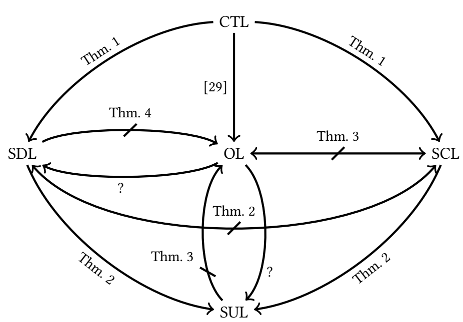

Figure 5: The expressivity results. An arrow from $\mathcal{L}_1$ to $\mathcal{L}_2$ means $\mathcal{L}_1 \prec \mathcal{L}_2$. A strike-out arrow from $\mathcal{L}_1$ to $\mathcal{L}_2$ depicts $\mathcal{L}_1 \not\preccurlyeq \mathcal{L}_2$. Open problems are denoted with question marks.

We leave the questions of whether OL $\not\preccurlyeq$ SDL and OL $\not\preccurlyeq$ SUL for future work and conjecture that it is indeed the case. In conclusion, we would also like to point out an interesting fact about the relationship between OL and SUL. Given $(\mathfrak{M}, s)$ and $\langle \dagger^n \rangle \varphi$ of OL, there is a formula $\langle\!\![\infty^n, \circledast^n]\!\!\rangle \varphi$ of SUL such that $(\mathfrak{M}, s) \models \langle \dagger^n \rangle \varphi$ implies $(\mathfrak{M}, s) \models \langle\!\![\infty^n, \circledast^n]\!\!\rangle \varphi$. This means that for each demonic strategy in OL, there is a joint demonic and angelic strategy in SUL over the same resource bounds reaching the same goal. Intuitively, to capture OL strategies, the angel and the demon can cooperate where the demon removes edges, and the angel restores all or a subset of them in the next turn.

4 MODEL CHECKING

In this section, we study the model checking problem for SDL, SCL, and SUL. In particular, we show that for the first two logics, the problem is PSPACE-complete, and for SUL, the problem is in EXPSPACE. We also mention that the problem is PSPACE-complete for the next-time fragment of SUL.

Definition 13 (Model Checking)**.** Given a pointed model $(\mathfrak{M}, s)$ and a formula φ, *the model checking problem* consists in computing whether $(\mathfrak{M}, s) \models \varphi$.

Whenever necessary, we assume that φ is in *negation normal form* (NNF), meaning that in φ, negations only appear in front of atoms. It is straightforward to show that each formula φ of SDL, SCL, and SUL, can be equivalently rewritten into φ' in NNF using the propositional equivalences, duals (like $[\circledast^n]\psi$ in the case of SDL), and temporal equivalences $\neg(\mathsf{X}\,\varphi) \leftrightarrow \mathsf{X}\,\neg\varphi$, $\neg(\varphi_1 \,\mathsf{U}\, \varphi_2) \leftrightarrow \neg\varphi_1 \,\mathsf{R}\, \neg\varphi_2$, and $\neg(\varphi_1 \,\mathsf{R}\, \varphi_2) \leftrightarrow \neg\varphi_1 \,\mathsf{U}\, \neg\varphi_2$. The size of the formula φ' in NNF is at most linear in the size of the original φ.

Theorem 5. Model checking SDL is PSPACE-complete.

Proof. We start with the lower bound and use a reduction from the PSPACE-complete quantified Boolean formula (QBF) problem. Given an instance of QBF $\Psi := Q_1 p_1 \dots Q_n p_n \psi(p_1, \dots, p_n)$, where $Q_i \in \{\exists, \forall\}$, the problem consists in determining whether Ψ is true. W.l.o.g., we assume that there are no free variables in Ψ and that each variable is used for quantification only once.

For a given instance of QBF Ψ, we construct a model $(\mathfrak{M}^\Psi, s)$ and a formula φ of SDL s.t. $(\mathfrak{M}^\Psi, s) \models \varphi$ iff Ψ is true. Starting with the model, consider $\mathfrak{M}^\Psi = (S, \rightarrow, \mathcal{V}, C)$, where $S = \{s, s_1, \dots, s_{2n}\}$, $s \rightarrow s$, $s \rightarrow s_i$ and $s_i \rightarrow s$ for all $s_i \in S$, $\mathcal{V}(p_i^1) = \{s_i\}$ and $\mathcal{V}(p_i^0) = \{s_{n+i}\}$ for $1 \leqslant i \leqslant n$, and $C(s, s) = 2$ and 1 for any other pair of states. Intuitively, for each p_i in Ψ, we have two states with corresponding atoms p_i^1 and p_i^0, modelling whether p_i was set to *true* or *false*, resp. The model $\mathfrak{M}^\Psi$, corresponding to a QBF instance Ψ with two variables p_1 and p_2, is depicted in Figure 6.

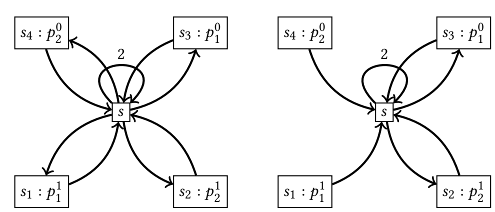

Figure 6: Model $\mathfrak{M}^\Psi$ (left) and one of its updates (right). Cost of all edges is 1 apart from the $s \rightarrow s$ edge that has cost 2.

Now, we will construct the corresponding formula φ of SDL. First, recall that n is the number of atoms in the QBF instance Ψ. Then, we define formula $chosen_k$ that will capture the fact that truth values of only the first k atoms out of n in Ψ were set:

$$chosen_k = \bigwedge_{1 \leqslant i \leqslant k} (\Diamond\Diamond p_i^0 \leftrightarrow \neg\Diamond\Diamond p_i^1) \wedge \bigwedge_{k < i \leqslant n} (\Diamond\Diamond p_i^0 \wedge \Diamond\Diamond p_i^1).$$

Recall that $\Diamond\varphi := [\circledast^0]\mathsf{X}\,\varphi$. Intuitively, $chosen_k$ holds if the demon has cut transitions to some of the first k atoms p_i^0 and p_i^1 in such a way that if the transition to p_i^0 is removed, then there must remain a transition to p_i^1. This simulates the unambiguous choice of the truth value of atom p_i in Ψ. All other atoms beyond the first k should still be accessible. We have double diamonds in the formula because each action of the demon is followed immediately by a move of the traveller. And if the traveller goes to one of the accessible s_l states, we should still be able to verify the accessibility of some p_i^j for $j \in \{0, 1\}$ in two steps passing the state s along the way.

Now, we are ready to tackle the construction of φ.

$$\varphi_0 := \psi(\Diamond\Diamond p_1^1, ..., \Diamond\Diamond p_n^1)$$

$$\varphi_k := \begin{cases} [\circledast^1]\mathsf{X}\,(chosen_k \to \varphi_{k-1}) & \text{if } Q_k = \forall \\ \langle\circledast^1\rangle\mathsf{X}\,(chosen_k \wedge \varphi_{k-1}) & \text{if } Q_k = \exists \end{cases}$$

$$\varphi := \varphi_n.$$

What is left to show is that $\Psi := Q_1 p_1 ... Q_n p_n \psi(p_1, ..., p_n)$ is true if and only if $(\mathfrak{M}^\Psi, s) \models \varphi$. First, observe that we consider demonic strategies with costs of up to 1, i.e., at each step the demon can remove up to one edge from s to some s_i[2]. Removing a transition to p_i^0 means that the truth value of p_i is set to *true*. For an example, see the model on the right in Figure 6, where the value of p_1 was set to *false*, and p_2 was set to *true*. Guards $chosen_k$ ensure that truth-values of propositions are chosen unambiguously. Therefore, together with the guards, constructs $[\circledast^1]\mathsf{X}$ and $\langle\circledast^1\rangle\mathsf{X}$ in the clause φ_k of the translation emulate quantifiers $\forall$ and $\exists$. Once the truth values of all atoms p_i were thus set, the evaluation of the QBF corresponds to the reachability of the corresponding atoms. In particular, we set p_i to *true* in an evaluation of Ψ if and only if p_i^1 is reachable in the corresponding submodel.

To show that the model checking problem for SDL is in PSPACE, we present an alternating Algorithm 1. Let $(\mathfrak{M}, s)$ be a finite model, and φ be a formula of SDL. Without loss of generality, we assume that formula φ is in NNF. In the algorithm, we show only cases $\langle\circledast^n\rangle\mathsf{X}\,\psi$ and $\langle\circledast^n\rangle\psi_1 \cup \psi_2$ for brevity, and the full algorithm can be found in [27].

Algorithm 1 An algorithm for model checking SDL

procedure MC$((\mathfrak{M}, s), \varphi)$
  case $\varphi = \langle\circledast^n\rangle\mathsf{X}\,\psi$
    existentially choose n-submodel $\mathfrak{M}'$
    universally choose s' such that $s \xrightarrow{\mathfrak{M}'} s'$
    return MC$((\mathfrak{M}', s'), \psi)$
  case $\varphi = \langle\circledast^n\rangle\psi_1 \cup \psi_2$
    $X \leftarrow (\mathfrak{M}, s)$
    $i \leftarrow 0$
    while not MC(X, ψ_2) and $i \leqslant brDepth$ **do**
      if not MC(X, ψ_1) **then**
        return *false*
      existentially choose n-submodel $\mathfrak{M}'$ of X
      universally choose s' such that $s \xrightarrow{\mathfrak{M}'} s'$
      $X \leftarrow (\mathfrak{M}', s')$
      $i \leftarrow i + 1$
    end while
    if $i > brDepth$ **then**
      return *false*
    else
      return *true*
end procedure

[2]Recall that by the definition of demonic strategies, the demon cannot remove an edge if it is the only outgoing edge from a given state.

In the algorithm, we explore the computational tree in a depth-first manner using universal and existential choices. The algorithm follows the definition of the semantics closely, and the correctness can be shown by induction on φ. The termination follows from the fact that each case breaks down a subformula into simpler subformulas and the finite depth of the tree.

The depth of each branch, *brDepth*, is at most $O(|\to| \cdot |S|)$, i.e., the demon has at most $|\to|$ ways to consecutively modify the model (in the worst case, removing edges one by one, plus an option to not modify the model), and each such n-submodel has at most $|S|$ states where the traveller can transition to. In other words, for each n-submodel, it is enough to check up to $|S|$ states in the submodel. Thus, the total number of *unique* game positions on a given branch, and hence the depth of each branch in the tree, is bounded by $O(|\to| \cdot |S|) \leqslant O(|\mathfrak{M}|^2)$ and can be explored by our alternating algorithm in polynomial time. Since there are at most $|\varphi|$ subformulas to consider, the total running time of the algorithm is bounded by $O(|\varphi| \cdot |\mathfrak{M}|^2)$. From the fact that APTIME (i.e., alternating polynomial time) = PSPACE [32], we conclude that the model checking problem for SDL is PSPACE-complete. □

A knowledgeable reader may point out that while model checking SDL is PSPACE-complete, model checking its conceptual predecessor, OL, can be done in polynomial time [29]. This is due to the significant difference in the semantics of the two logics. In OL, edges are restored after the traveller has made a move, and in SDL, edge removal is permanent, i.e., we have to keep in memory a polynomial number of modified models to verify the strategic abilities of the demon. However, note that our PSPACE-completeness result is in line with the model checking results for logics with quantification over *permanent* model change (see, e.g., [37, 44, 51, 62, 77]).

Model checking SCL is also PSPACE-complete by a relatively similar argument that we omit for brevity, and a proof sketch can be found in [27].

Theorem 6. Model checking SCL is PSPACE-complete.

Finally, we look into the complexity of the model checking problem for SUL and show that it is in EXPSPACE. However, we also note that the next-time fragment of SUL is still PSPACE-complete.

Theorem 7. Model checking SUL is in EXPSPACE, and model checking the next-time fragment of SUL is PSPACE-complete.

Proof. To show that model checking SUL is in EXPSPACE, we can provide an alternating algorithm similar to Algorithm 1. Here, we present a general idea, and the algorithm with the full argument can be found in [27].

The main difference from Algorithm 1 is that for cases involving $\langle\!\langle C \rangle\!\rangle^{n,m}$ we have to consider n-m-updates depending on C. Each n-m-update is of size bounded by $O(|\mathfrak{M}|^2)$, and can be computed in polynomial time. For the next-time fragment of SUL, we need to check at most $|\varphi|$ such updates, and hence the total running time is in APTIME = PSPACE. The lower bound follows from the fact that SUL subsumes both SDL and SCL.

For the full language, similarly to Algorithm 1, we construct the computational tree with depth of branches bounded by *brDepth*.

However, this time, *brDepth* is exponential, since the demon and the angel together can, in the worst case, force an exponential number of submodels of the fully connected graph of size $|S|^2$, i.e., *brDepth* is bounded by $O(2^{|\mathfrak{M}|^2})$ and hence the algorithm runs in exponential time. Since AEXPTIME (i.e., alternating exponential time) = EXPSPACE [32], model checking SUL is in EXPSPACE. $\square$

5 RELATED WORK

Obstruction Logics. The initial inspiration for our work came from the research on *Obstruction Logic* (OL) [29], where one can reason about the demon's strategies to deactivate some edges so that the target property holds regardless of the traveller's moves. After each step of such a game, all edges are restored, which is different from our SDL setting, where we assume that edges are deactivated once and for all. Extensions of OL include obstruction ATL [30], timed OL [58], and coalition OL [31]. All of these extensions feature only temporary edge deactivations. Compared to the body of research on OL, we have also introduced a formalism for reasoning about strategic *activation* of edges, SCL, and also the interplay between the demon and the angel in SUL for scenarios of cooperation and competition.

Sabotage-like Logics. The research on OL itself was motivated by sabotage games [73], where the demon can deactivate one arbitrary, and the corresponding *Sabotage Modal Logic* (SML) [15, 62] and its generalisation to subsets of edges [28]. A related approach considers definable edge removal [59]. In the same vein, there have also been logics for reasoning about adding an edge to a model, swapping two edges, copying and removing, etc (see, e.g., [12–14]). Compared to all these approaches, we consider weighted graphs, as well as extended strategies of the angel and the demon, as opposed to the next-time outcomes in the cited works.

Dynamic Epistemic Logic. Adding and removing arrows in particular, and model updates in general, are bread-and-butter in *Dynamic Epistemic Logics* (DEL) [76], which are built on epistemic models capturing knowledge of agents, and where updates of such models correspond to various information-changing events. Related to edge removal, one can mention arrow updates [56, 77], as well as various modes of agents sharing their knowledge with each other [3, 17, 51]. One can also think of adding arrows as an introspection effort of a given agent [44]. Moreover, there has been some work on incorporating strategic reasoning into DEL, in particular as reachability games over epistemic models [63, 64], concurrent public communication [2, 45], and alternating-time DEL [37]. Even though we can refer to DEL for some intuitions regarding model updates, needless to say, our setting is different from the one of agents' knowledge and learning.

Dynamic Logics for Social Networks. Inspired by DEL, there has been a considerable amount of research on dynamics in social networks (SNs) (see [67] for an overview). In the setting of SNs, adding or removing edges has been used, for example, to model changes in friendship [16, 52], (un)following other agents [78], gaining knowledge in SNs [33], visibility of posts on SNs [50], balance in a network [75], and sellers' strategies in diffusion auctions [49].

Strategic Reasoning. In the realm of the strategic reasoning, we see model modifications, apart from the case of OL, primarily in *normative reasoning*, where, given a MAS, a social law (or a norm), divides the set of actions of a given agent in a given state into desirable, or allowed, and undesirable, or prohibited. This is usually done by removing some of the transitions in a model (see, e.g., [1, 4, 7, 10, 24, 48, 55]). A general approach to modifications of strategic multi-agent models has recently been proposed in [47]. Adding and removing edges was also interpreted as granting or revoking abilities to\from agents [46]. One can mention the research on module checking [54, 57], where some transitions in an execution tree are cut to model various behaviours of the environment. Some types of model dynamics are prominent in separation logics (see, e.g., [18, 38, 39, 69]) that allow reasoning about the execution of computer programs. To the best of our knowledge, none of these approaches reason about extended temporal goals of agents that are able to modify the topology of the underlying model.

Logics with Resources. In the context of epistemic logics, resources and costs were used to tackle the logical omniscience problem (see, e.g., [5, 6, 42, 43]), as constraints on dynamic epistemic actions [36, 41, 72], and for epistemic planning [20, 23], to name a few use cases. In the context of strategic reasoning, there is a plethora of research on resource-bounded agents in the settings of CL and ATL (for example, see [8, 9, 19, 25, 26, 40, 65]). Observe that in the latter case, models are *static* as opposed to our *dynamic* models, and hence the settings are very different.

Games on Graphs. Finally, sabotage games are not the only games on graphs that involve a traveller and that were analysed with the tools of modal logic (see [74] for an overview). Such analyses include the logics for hide and seek [61, 71], cops and robbers [60], and the poison games [53, 79].

6 DISCUSSION

We have presented three novel logics for reasoning about strategies of agents that are able to modify a given model. In SDL, we have the demon who plays an edge-removing strategy such that for all paths taken by the traveller, a target condition holds. Similarly, to reason about edge-adding strategies of the angel, we have proposed SCL. Finally, to express the interaction between the demon and the angel in an ATL-like fashion, we introduced SUL. For all logics, we studied their expressivity and provided model-checking algorithms.

Since we have just scratched the surface of reasoning about strategic model changes by agents, there is a plethora of future work. First, we would like to study the satisfiability problems of all three logics. Moreover, recall that we defined demonic and angelic strategies in a memory-less fashion, i.e., an action of an agent depends on the current pointed model. We would also like to study their perfect recall strategies, where their actions depend on histories consisting of pointed models. We conjecture that the semantics of all logics are equivalent for both types of strategies via an argument similar to the one for OL [29].

While defining SDL, we were inspired by OL. However, having introduced edge-adding strategies, it is very tempting to formalise and study a variant of OL, where instead of the demon we have an angel that adds edges for one turn only. Moreover, related to [31], we find it particularly intriguing to extend SUL to *coalitions of angels and demons*. Finally, we would also like to consider extensions of our logics with greatest and least fixed points in the vein of [70].

REFERENCES

[1] Thomas Ågotnes, Wiebe van der Hoek, and Michael J. Wooldridge. 2010. Robust normative systems and a logic of norm compliance. *Logic Journal of the IGPL* 18, 1 (2010), 4–30. https://doi.org/10.1093/jigpal/jzp070

[2] Thomas Ågotnes and Hans van Ditmarsch. 2008. Coalitions and announcements. In *Proceedings of the 7th AAMAS*, Lin Padgham, David C. Parkes, Jörg P. Müller, and Simon Parsons (Eds.). IFAAMAS, 673–680.

[3] Thomas Ågotnes and Yì N. Wáng. 2017. Resolving distributed knowledge. *Artificial Intelligence* 252 (2017), 1–21. https://doi.org/10.1016/J.ARTINT.2017.07.002

[4] Natasha Alechina, Giuseppe De Giacomo, Brian Logan, and Giuseppe Perelli. 2022. Automatic Synthesis of Dynamic Norms for Multi-Agent Systems. In *Proceedings of the 19th KR*, Gabriele Kern-Isberner, Gerhard Lakemeyer, and Thomas Meyer (Eds.).

[5] Natasha Alechina and Brian Logan. 2002. Ascribing beliefs to resource bounded agents. In *Proceedings of the 1st AAMAS*. ACM, 881–888. https://doi.org/10.1145/544862.544948

[6] Natasha Alechina and Brian Logan. 2009. A Logic of Situated Resource-Bounded Agents. *Journal of Logic, Language and Information* 18, 1 (2009), 79–95. https://doi.org/10.1007/S10849-008-9073-6

[7] Natasha Alechina, Brian Logan, and Mehdi Dastani. 2018. Modeling Norm Specification and Verification in Multiagent Systems. *FLAP* 5, 2 (2018), 457–490.

[8] Natasha Alechina, Brian Logan, Nguyen Hoang Nga, and Abdur Rakib. 2010. Resource-bounded alternating-time temporal logic. In *Proceedings of the 9th AAMAS*, Wiebe van der Hoek, Gal A. Kaminka, Yves Lespérance, Michael Luck, and Sandip Sen (Eds.). IFAAMAS, 481–488.

[9] Natasha Alechina, Brian Logan, Nguyen Hoang Nga, and Abdur Rakib. 2011. Logic for coalitions with bounded resources. *Journal of Logic and Computation* 21, 6 (2011), 907–937. https://doi.org/10.1093/LOGCOM/EXQ032

[10] Natasha Alechina, Brian Logan, and Giuseppe Perelli. 2025. Synthesising Minimum Cost Dynamic Norms. In *Proceedings of the 34th IJCAI 2025*. ijcai.org, 3–11. https://doi.org/10.24963/IJCAI.2025/1

[11] Rajeev Alur, Thomas A. Henzinger, and Orna Kupferman. 2002. Alternating-time temporal logic. *J. ACM* 49 (2002), 672–713. https://doi.org/10.1145/585265.585270

[12] Carlos Areces, Raul Fervari, and Guillaume Hoffmann. 2015. Relation-changing modal operators. *Logic Journal of the IGPL* 23, 4 (2015), 601–627. https://doi.org/10.1093/JIGPAL/JZV020

[13] Carlos Areces, Hans van Ditmarsch, Raul Fervari, Bastien Maubert, and François Schwarzentruber. 2021. Copy and remove as dynamic operators. *Journal of Applied Non-Classical Logics* 31, 3-4 (2021), 181–220. https://doi.org/10.1080/11663081.2021.1964327

[14] Carlos Areces, Hans van Ditmarsch, Raul Fervari, and François Schwarzentruber. 2017. The modal logic of copy and remove. *Information and Computation* 255 (2017), 243–261. https://doi.org/10.1016/J.IC.2017.01.004

[15] Guillaume Aucher, Johan van Benthem, and Davide Grossi. 2018. Modal logics of sabotage revisited. *Journal of Logic and Computation* 28, 2 (2018), 269–303. https://doi.org/10.1093/LOGCOM/EXX034

[16] Edoardo Baccini and Zoé Christoff. 2023. Comparing Social Network Dynamic Operators. In *Proceedings of the 19th TARK (EPTCS, Vol. 379)*, Rineke Verbrugge (Ed.). 66–81. https://doi.org/10.4204/EPTCS.379.8

[17] Alexandru Baltag and Sonja Smets. 2020. Learning What Others Know. In *Proceedings of the 23rd LPAR (EPiC Series in Computing, Vol. 73)*, Elvira Albert and Laura Kovács (Eds.). EasyChair, 90–119. https://doi.org/10.29007/PLM4

[18] Bartosz Bednarczyk, Stéphane Demri, Raul Fervari, and Alessio Mansutti. 2023. On Composing Finite Forests with Modal Logics. *ACM Transactions on Computational Logic* 24, 2 (2023), 12:1–12:46. https://doi.org/10.1145/3569954

[19] Francesco Belardinelli and Stéphane Demri. 2021. Strategic reasoning with a bounded number of resources: The quest for tractability. *Artificial Intelligence* 300 (2021), 103557. https://doi.org/10.1016/J.ARTINT.2021.103557

[20] Gaia Belardinelli and Rasmus K. Rendsvig. 2021. Epistemic Planning with Attention as a Bounded Resource. In *Proceedings of the 8th LORI (LNCS, Vol. 13039)*, Sujata Ghosh and Thomas Icard (Eds.). Springer, 14–30. https://doi.org/10.1007/978-3-030-88708-7_2

[21] Deval Bhamare, Maede Zolanvari, Aiman Erbad, Raj Jain, Khaled Khan, and Nader Meskin. 2020. Cybersecurity for industrial control systems: A survey. *computers & security* 89 (2020), 101677.

[22] Cem Bila, Fikret Sivrikaya, Manzoor A Khan, and Sahin Albayrak. 2016. Vehicles of the future: A survey of research on safety issues. *IEEE Transactions on Intelligent Transportation Systems* 18, 5 (2016), 1046–1065.

[23] Thomas Bolander, Lasse Dissing, and Nicolai Herrmann. 2021. DEL-based Epistemic Planning for Human-Robot Collaboration: Theory and Implementation. In *Proceedings of the 18th KR*, Meghyn Bienvenu, Gerhard Lakemeyer, and Esra Erdem (Eds.). 120–129. https://doi.org/10.24963/KR.2021/12

[24] Nils Bulling and Mehdi Dastani. 2016. Norm-based mechanism design. *Artificial Intelligence* 239 (2016), 97–142.

[25] Rui Cao and Pavel Naumov. 2017. Budget-Constrained Dynamics in Multiagent Systems. In *Proceedings of the 26th IJCAI*, Carles Sierra (Ed.). ijcai.org, 915–921. https://doi.org/10.24963/IJCAI.2017/127

[26] Davide Catta, Angelo Ferrando, and Vadim Malvone. 2024. Resource Action-Based Bounded ATL: A New Logic for MAS to Express a Cost Over the Actions. In *Proceedings of the 25th PRIMA (LNCS, Vol. 15395)*, Ryuta Arisaka, Víctor Sánchez-Anguix, Sebastian Stein, Reyhan Aydogan, Leon van der Torre, and Takayuki Ito (Eds.). Springer, 206–223. https://doi.org/10.1007/978-3-031-77367-9_16

[27] Davide Catta, Rustam Galimullin, and Munyque Mittelmann. 2026. On Angels and Demons: Strategic (De)Construction of Dynamic Models. *CoRR* abs/2601.07690 (2026). https://doi.org/10.48550/arXiv.2601.07690

[28] Davide Catta, Jean Leneutre, and Vadim Malvone. 2023. Attack Graphs & Subset Sabotage Games. *Intelligenza Artificiale* 17, 1 (2023), 77–88. https://doi.org/10.3233/IA-221080

[29] Davide Catta, Jean Leneutre, and Vadim Malvone. 2023. Obstruction Logic: A Strategic Temporal Logic to Reason About Dynamic Game Models. In *Proceedings of the 26th ECAI (Frontiers in Artificial Intelligence and Applications, Vol. 372)*, Kobi Gal, Ann Nowé, Grzegorz J. Nalepa, Roy Fairstein, and Roxana Radulescu (Eds.). IOS Press, 365–372. https://doi.org/10.3233/FAIA230292

[30] Davide Catta, Jean Leneutre, Vadim Malvone, and Aniello Murano. 2024. Obstruction Alternating-time Temporal Logic: A Strategic Logic to Reason about Dynamic Models. In *Proceedings of the 23rd AAMAS*, Mehdi Dastani, Jaime Simão Sichman, Natasha Alechina, and Virginia Dignum (Eds.). IFAAMAS / ACM, 271–280.

[31] Davide Catta, Jean Leneutre, Vadim Malvone, and James Ortiz. 2025. Coalition Obstruction Temporal Logic: A New Obstruction Logic to Reason About Demon Coalitions. In *Proceedings of the 34th IJCAI*. ijcai.org, 21–28. https://doi.org/10.24963/IJCAI.2025/3

[32] Ashok K. Chandra, Dexter Kozen, and Larry J. Stockmeyer. 1981. Alternation. *Journal of the ACM* 28, 1 (1981), 114–133. https://doi.org/10.1145/322234.322243

[33] Zoé Christoff, Jens Ulrik Hansen, and Carlo Proietti. 2016. Reflecting on Social Influence in Networks. *Journal of Logic, Language and Information* 25, 3-4 (2016), 299–333. https://doi.org/10.1007/S10849-016-9242-Y

[34] Edmund M. Clarke and E. Allen Emerson. 1981. Design and Synthesis of Synchronization Skeletons Using Branching-Time Temporal Logic. In *Logics of Programs (LNCS, Vol. 131)*, Dexter Kozen (Ed.). Springer, 52–71. https://doi.org/10.1007/BFB0025774

[35] Edmund M. Clarke, Thomas A. Henzinger, Helmut Veith, and Roderick Bloem (Eds.). 2018. *Handbook of Model Checking*. Springer.

[36] Stefania Costantini, Andrea Formisano, and Valentina Pitoni. 2021. An Epistemic Logic for Multi-agent Systems with Budget and Costs. In *Proceedings of the 17th JELIA (LNCS, Vol. 12678)*, Wolfgang Faber, Gerhard Friedrich, Martin Gebser, and Michael Morak (Eds.). Springer, 101–115. https://doi.org/10.1007/978-3-030-75775-5_8

[37] Tiago de Lima. 2014. Alternating-time temporal dynamic epistemic logic. *Journal of Logic and Computation* 24, 6 (2014), 1145–1178. https://doi.org/10.1093/LOGCOM/EXS061

[38] Stéphane Demri and Morgan Deters. 2015. Separation logics and modalities: a survey. *Journal of Applied Non-Classical Logics* 25, 1 (2015), 50–99. https://doi.org/10.1080/11663081.2015.1018801

[39] Stéphane Demri and Raul Fervari. 2019. The power of modal separation logics. *Journal of Logic and Computation* 29, 8 (2019), 1139–1184. https://doi.org/10.1093/LOGCOM/EXZ019

[40] Stéphane Demri and Raine Rönnholm. 2023. How to Manage a Budget with ATL+. In *Proceedings of the 20th KR*, Pierre Marquis, Tran Cao Son, and Gabriele Kern-Isberner (Eds.). 188–197. https://doi.org/10.24963/KR.2023/19

[41] Vitaliy Dolgorukov, Rustam Galimullin, and Maksim Gladyshev. 2024. Dynamic Epistemic Logic of Resource Bounded Information Mining Agents. In *Proceedings of the 23rd AAMAS*, Mehdi Dastani, Jaime Simão Sichman, Natasha Alechina, and Virginia Dignum (Eds.). IFAAMAS / ACM, 481–489. https://doi.org/10.5555/3635637.3662898

[42] Ho Ngoc Duc. 1997. Reasoning About Rational, But Not Logically Omniscient, Agents. *Journal of Logic and Computation* 7, 5 (1997), 633–648. https://doi.org/10.1093/LOGCOM/7.5.633

[43] Ronald Fagin and Joseph Y. Halpern. 1987. Belief, Awareness, and Limited Reasoning. *Artificial Intelligence* 34, 1 (1987), 39–76. https://doi.org/10.1016/0004-3702(87)90003-8

[44] Raul Fervari and Fernando R. Velázquez-Quesada. 2019. Introspection as an action in relational models. *Journal of Logical and Algebraic Methods in Programming* 108 (2019), 1–23. https://doi.org/10.1016/J.JLAMP.2019.06.005

[45] Rustam Galimullin. 2021. Coalition and Relativised Group Announcement Logic. *Journal of Logic, Language and Information* 30, 3 (2021), 451–489. https://doi.org/10.1007/S10849-020-09327-2

[46] Rustam Galimullin and Thomas Ågotnes. 2021. Dynamic Coalition Logic: Granting and Revoking Dictatorial Powers. In *Proceedings of the 8th LORI (LNCS, Vol. 13039)*, Sujata Ghosh and Thomas Icard (Eds.). Springer, 88–101. https://doi.org/10.1007/978-3-030-88708-7_7

[47] Rustam Galimullin, Maksim Gladyshev, Munyque Mittelmann, and Nima Motamed. 2025. Changing the Rules of the Game: Reasoning About Dynamic Phenomena in Multi-Agent Systems. In *Proceedings of the 24th AAMAS*, Sanmay Das, Ann Nowé, and Yevgeniy Vorobeychik (Eds.). IFAAMAS / ACM, 829–838. https://doi.org/10.5555/3709347.3743601

[48] Rustam Galimullin and Louwe B. Kuijer. 2024. Synthesizing Social Laws with ATL Conditions. In *Proceedings of the 23rd AAMAS*, Mehdi Dastani, Jaime Simão Sichman, Natasha Alechina, and Virginia Dignum (Eds.). IFAAMAS / ACM, 2270–2272. https://doi.org/10.5555/3635637.3663130

[49] Rustam Galimullin, Munyque Mittelmann, and Laurent Perrussel. 2026. Formal Verification of Diffusion Auctions. In *Proceedings of the 40th AAAI*. (to appear).

[50] Rustam Galimullin and Mina Young Pedersen. 2024. Visibility and exploitation in social networks. *Mathematical Structures in Computer Science* 34, 7 (2024), 615–644. https://doi.org/10.1017/S0960129523000397

[51] Rustam Galimullin and Fernando R. Velázquez-Quesada. 2024. Topic-Based Communication Between Agents. *Studia Logica* (2024). https://doi.org/10.1007/s11225-024-10119-z

[52] Saúl Fernández González. 2022. Change in social networks: Some dynamic extensions of Social Epistemic Logic. *Journal of Logic and Computation* 32, 6 (2022), 1212–1233. https://doi.org/10.1093/LOGCOM/EXAC024

[53] Davide Grossi and Simon Rey. 2019. Credulous Acceptability, Poison Games and Modal Logic. In *Proceedings of the 18th AAMAS*, Edith Elkind, Manuela Veloso, Noa Agmon, and Matthew E. Taylor (Eds.). IFAAMASI, 1994–1996.

[54] Wojciech Jamroga and Aniello Murano. 2015. Module Checking of Strategic Ability. In *Proceedings of the 14th AAMAS*, Gerhard Weiss, Pinar Yolum, Rafael H. Bordini, and Edith Elkind (Eds.). ACM, 227–235.

[55] Max Knobbout, Mehdi Dastani, and John-Jules Ch. Meyer. 2016. A Dynamic Logic of Norm Change. In *Proceedings of the 22nd ECAI (Frontiers in Artificial Intelligence and Applications, Vol. 285)*, Gal A. Kaminka, Maria Fox, Paolo Bouquet, Eyke Hüllermeier, Virginia Dignum, Frank Dignum, and Frank van Harmelen (Eds.). IOS Press, 886–894. https://doi.org/10.3233/978-1-61499-672-9-886

[56] Barteld Kooi and Bryan Renne. 2011. Arrow Update Logic. *The Review of Symbolic Logic* 4, 4 (2011), 536–559. https://doi.org/10.1017/S1755020311000189

[57] Orna Kupferman, Moshe Y Vardi, and Pierre Wolper. 2001. Module checking. *Information and Computation* 164, 2 (2001), 322–344.

[58] Jean Leneutre, Vadim Malvone, and James Ortiz. 2025. Timed Obstruction Logic: A Timed Approach to Dynamic Game Reasoning. In *Proceedings of the 24th AAMAS*, Sanmay Das, Ann Nowé, and Yevgeniy Vorobeychik (Eds.). IFAAMAS / ACM, 1272–1281. https://doi.org/10.5555/3709347.3743759

[59] Dazhu Li. 2020. Losing connection: the modal logic of definable link deletion. *Journal of Logic and Computation* 30, 3 (2020), 715–743. https://doi.org/10.1093/LOGCOM/EXZ036

[60] Dazhu Li, Sujata Ghosh, and Fenrong Liu. 2025. Reasoning under uncertainty in the game of Cops and Robbers. *CoRR* abs/2508.00004 (2025). https://doi.org/10.48550/ARXIV.2508.00004

[61] Dazhu Li, Sujata Ghosh, Fenrong Liu, and Yaxin Tu. 2023. A Simple Logic of the Hide and Seek Game. *Studia Logica* 111, 5 (2023), 821–853. https://doi.org/10.1007/S11225-023-10039-4

[62] Christof Löding and Philipp Rohde. 2003. Model checking and satisfiability for sabotage modal logic. In *Proceedings of the 23rd FSTTCS (LNCS, Vol. 2914)*, Paritosh K. Pandya and Jaikumar Radhakrishnan (Eds.). Springer, 302–313. https://doi.org/10.1007/978-3-540-24597-1_26

[63] Bastien Maubert, Sophie Pinchinat, and François Schwarzentruber. 2019. Reachability Games in Dynamic Epistemic Logic. In *Proceedings of the 28th IJCAI*, Sarit Kraus (Ed.). ijcai.org, 499–505. https://doi.org/10.24963/IJCAI.2019/71

[64] Bastien Maubert, Sophie Pinchinat, François Schwarzentruber, and Silvia Stranieri. 2020. Concurrent Games in Dynamic Epistemic Logic. In *Proceedings of the 29th IJCAI*, Christian Bessiere (Ed.). 1877–1883. https://doi.org/10.24963/IJCAI.2020/260

[65] Dario Della Monica, Margherita Napoli, and Mimmo Parente. 2011. On a Logic for Coalitional Games with Priced-Resource Agents. In *Proceedings of the 7th M4M and 4th LAMAS (ENTCS, Vol. 278)*, Hans van Ditmarsch, David Fernández-Duque, Valentin Goranko, Wojciech Jamroga, and Manuel Ojeda-Aciego (Eds.). Elsevier, 215–228. https://doi.org/10.1016/J.ENTCS.2011.10.017

[66] Marc Pauly. 2002. A modal logic for coalitional power in games. *Journal of Logic and Computation* 12, 1 (2002), 149–166. https://doi.org/10.1093/logcom/12.1.149

[67] Mina Young Pedersen. 2024. *Malicious Agents and the Power of the Few : On the Logic of Abnormality in Social Networks.* Ph.D. Dissertation. University of Bergen, Norway. https://bora.uib.no/bora-xmlui/handle/11250/3151733

[68] Amir Pnueli. 1977. The Temporal Logic of Programs. In *Proc. of the Annual Symp. on Foundations of Computer Science.*

[69] John C. Reynolds. 2002. Separation Logic: A Logic for Shared Mutable Data Structures. In *Proceedings of the 17th LICS*. IEEE Computer Society, 55–74. https://doi.org/10.1109/LICS.2002.1029817

[70] Philipp Rohde. 2006. On the μ-Calculus Augmented with Sabotage. In *Proceedings of the 9th FOSSACS (LNCS, Vol. 3921)*, Luca Aceto and Anna Ingólfsdóttir (Eds.). Springer, 142–156. https://doi.org/10.1007/11690634_10

[71] Katsuhiko Sano, Fenrong Liu, and Dazhu Li. 2024. Hybrid logic of the hide and seek game. *Studia Logica* (2024), 1–33.

[72] Anthia Solaki. 2023. Actualizing distributed knowledge in bounded groups. *Journal of Logic and Computation* 33, 6 (2023), 1497–1525. https://doi.org/10.1093/LOGCOM/EXAC007

[73] Johan van Benthem. 2005. An Essay on Sabotage and Obstruction. In *Mechanizing Mathematical Reasoning (LNCS, Vol. 2605)*. Springer, 268–276.

[74] Johan van Benthem and Fenrong Liu. 2020. Graph Games and Logic Design. In *Knowledge, Proof and Dynamics*, Fenrong Liu, Hiroakira Ono, and Junhua Yu (Eds.). Springer, 125–146.

[75] Wiebe van der Hoek, Louwe B. Kuijer, and Yì N. Wáng. 2020. Logics of Allies and Enemies: A Formal Approach to the Dynamics of Social Balance Theory. In *Proceedings of the 29th IJCAI*, Christian Bessiere (Ed.). ijcai.org, 210–216. https://doi.org/10.24963/IJCAI.2020/30

[76] Hans van Ditmarsch, Wiebe van der Hoek, and Barteld Kooi. 2008. *Dynamic Epistemic Logic*. Synthese Library, Vol. 337. Springer.

[77] Hans van Ditmarsch, Wiebe van der Hoek, Barteld Kooi, and Louwe B. Kuijer. 2017. Arbitrary arrow update logic. *Artificial Intelligence* 242 (2017), 80–106. https://doi.org/10.1016/J.ARTINT.2016.10.003

[78] Zuojun Xiong and Meiyun Guo. 2019. A Dynamic Hybrid Logic for Followership. In *Proceedings of the 7th LORI (LNCS, Vol. 11813)*, Patrick Blackburn, Emiliano Lorini, and Meiyun Guo (Eds.). Springer, 425–439. https://doi.org/10.1007/978-3-662-60292-8_31

[79] Francesca Zaffora Blando, Krzysztof Mierzewski, and Carlos Areces. 2020. The Modal Logics of the Poison Game. In *Knowledge, Proof and Dynamics*, Fenrong Liu, Hiroakira Ono, and Junhua Yu (Eds.). Springer, 3–23.