Graph Games and Logic Design

Abstract

Graph games are interactive scenarios with a wide range of applications. This position paper discusses old and new graph games in tandem with matching logics, and identifies general questions behind this match. Throughout, we pursue two strands: logic as a way of analyzing existing games, and logic as an inspiration for designing new games. Our aim is modest: we propose a style of thinking that complements existing game-theoretic and computational ones, we raise questions, make observations, and suggest research directions – technical results are left to future work. But frankly, our main aim with this survey paper is to show that graph games are concrete, fun, easy to grasp, and yet challenging to study.

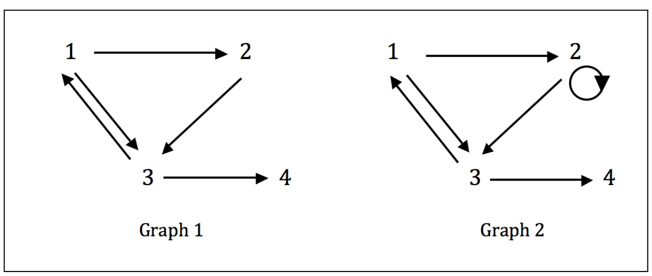
1 Two graph games

Graph games are played on graphs, directed or undirected, with one or many relations, perhaps with annotations at vertices. This setting may seem specialized – but the scope of these games is broad, with applications to computation, argumentation, communication, social networks, warfare, and other scenarios. To introduce our perspective, we start with two concrete examples. For general background on logic and games presupposed in this paper, we refer to [10], [13].

1.1 Traveling along a graph

Travel games Travel games take place on a graph **G** with a starting point s. Two players A, E take turns in going along available edges, thus moving to new points – A starts. A player loses at any point if it is her turn to move, but there is no available edge to travel. Infinite runs are considered a win for player E.

Example A travel game.



First consider Graph 1. If player A starts at point 1, and moves to 3, then E can move to the endpoint 4, and A loses. Note that E also has another strategy in this eventuality, which is to stay in the two-cycle $\{1, 3\}$. However, overall, A has a winning strategy: first go to 2, then player E must go to 3, and A can go to 4. Other positions can be analyzed in a similar manner.

Next, consider Graph 2 with its slight change in accessibility. This time, if A starts at 1, it is E who has a winning strategy. If A moves to 3, E wins as before. And if A moves to 2, E chooses the reflexive loop and stays there. If A wants to prevent E from winning an infinite history staying at 2, then he must move to 3 – but then E wins again by moving to 4.

There are also variations of the travel game, with other winning conventions, or with separate relations for the two players – and we will see others below.

Under suitable choices of vertices and edges, travel games have a broad range of applications: from analyzing complexity of evaluating logical formulas in models, or of finding models, to solving general search problems, [22].

Theoretically, by the Gale-Stewart Theorem, many travel games are 'determined', thanks to the simplicity of their set-up and their winning conditions: in any starting position $(\mathbf{G}, s)$, either player A or E has a winning strategy.

Modal logic There is an illuminating logical side to travel games. Graphs are natural models for a modal logic, and the moves are described by modalities. In particular, a winning response pattern for player E typically involves an iterated universal-existential modality of the form $\Box\Diamond$: E has a responding move to whatever A chooses. In a more elaborate modal formalism, the winning positions for E are defined by the following formula of the modal μ-calculus:

$$\nu p \cdot \Box\Diamond p$$

This fixed-point formula defines a largest subset P of the graph where player E can always keep moving inside P, no matter which move is chosen by A.

Thus, an elementary theory of travel games is contained in well-known systems of modal logic. As a benefit, model checking for modal properties in graphs can be done fast, often in polynomial time. Moreover, the modal language is invariant for bisimulation between graphs on which the game is played, and this leads to standard model-theoretic results on preservation under graph change and other topics. Finally, the valid modal laws of travel games are decidable.

Coda: logic games Interestingly, the travel game itself is not only described by modal formulas. It may also be seen as an evaluation game for a modal language of games, applied to the above special formula $\Box\Diamond p$ describing the existence of winning strategies, suitably iterated to match the size of the graph.[1] Thus, the travel game returns higher up as a 'logic game' for its own modal logic. This intriguing entanglement will return in what follows.

[1] The travel game matches the usual evaluation game for the μ-calculus formula $\nu p \cdot \Box\Diamond p$.

1.2 Travel and Damage

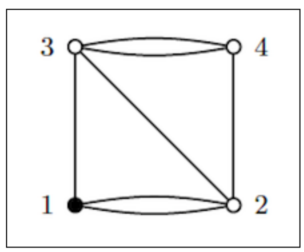
The sabotage game The following graph game has been proposed to model what happens to standard computational tasks, such as path search, in a perturbed environment, [10]. Starting from an initial point s, a 'Traveler' T moves through a graph $\mathbf{G}$ with possibly multiple edges, one link at a time. However, in each round of the game, a 'Demon' D first cuts one link, anywhere in the graph. Traveler wins when she reaches her goal point, or more generally, any point in her perhaps larger goal region in the graph. Also, players lose when they cannot move: this would happen to Demon when there are no more links, or to Traveler if she stands at an isolated point. This time, unlike in the Travel Game, the graph itself changes as the game proceeds.

Many further interpretations exist for sabotage games, including blocking access in information networks, or learning viewed as a process of guiding players into certain subregions of a graph.

Again the game is determined. The conditions of Zermelo's Theorem or the Gale-Stewart Theorem are met, so Traveler or Demon has a winning strategy.

Example A sabotage game.

Let Traveler start in point 1 of the graph depicted, trying to reach point 4.



It is Demon who has the winning strategy, by first cutting one upper link between points 3 and 4, and then responding appropriately to whatever Traveler does. Note, however, that Demon has no winning strategy if he is restricted to 'local sabotage', cutting links at the current position of Traveler. Other positions and other goals can be analyzed in a similar fashion. Also, the example is easily modified to work with directed instead of undirected graphs.

Sabotage modal logic Again there is a logical background here. Just as the travel game, the sabotage game suggests a modal language with standard modalities $\Diamond$ describing moves by Traveler, but it has to add new less standard modalities $\blacksquare$ for Demon's graph change, describing what is true at the current point after one edge has been deleted from the current accessibility relation. Winning strategies for Traveler are then associated with modal patterns

$$\blacksquare \Diamond \blacksquare \Diamond \ldots$$

More general winning positions for players can be defined in the 'sabotage μ-calculus' which adds the $\blacksquare$ modality to the usual syntax of the modal μ-calculus.

An example is the formula for winning positions of Traveler with a goal region defined as all points satisfying the formula γ, [8]:

$$\nu p \cdot (\gamma \vee \Box\bot \vee \blacksquare \Diamond p)$$

Unlike with travel games, even the modal logical theory of sabotage games is not so smooth and simple, and it contains several surprises. Model checking for the basic modal sabotage logic takes Pspace instead of Ptime, so a complexity jump occurs in the logic of gamified simple algorithmic tasks involving perturbation. Moreover, while this modal logic is still effectively translatable into first-order logic, satisfiability is undecidable – and despite its being axiomatizable in principle, for instance using suitably adapted semantic tableaux, no perspicuous axiomatization is known. [2] And while there is a model theory based on a natural notion of sabotage bisimulation, open problems abound.

These issues extend to the sabotage μ-calculus, whose semantics differs from what one might expect prima facie, due to the fact that evaluation of approximations for fixed-points may require jumping across different graphs, [8].

The other side of this additional complexity is, of course, that sabotage games are more interesting to play than travel games.

Digression: concrete cases To explore how this general perspective works in practice, one can consider solvability of the games on special classes of graphs.[3]

Sabotage in trees Consider finite intransitive trees, with arrows pointing from parent to child nodes. In this case, the sabotage game simplifies considerably. First, there is no loss of power for Demon if only local cutting is allowed, that is, just on immediate successors of Traveler's position. For, such moves block the whole subsequent subtree for Traveler – so, if a winning strategy for Demon involved cutting further down at some stage, he could just as well have cut at the top of the subtree where Traveler is standing. Given this, it can be proved that Traveler has a winning strategy standing at a node s iff there is a 'rich subtree' T starting at s with out-degree 2 (there are at least two successors in T for every parent node in T) all of whose end nodes are in the goal region.[4] [5]

[2] [3] has new axiomatization techniques combining hybrid logic and dynamic-epistemic logic.

[3] For a general study of graph change encompassing sabotage, but also other mechanisms, see the systematic approach in [5], which studies more general hybrid logics of graph change.

[4] If there is such a rich subtree, Traveler can obviously stay inside it, no matter where Demon cuts, and eventually, as the game moves down the tree, Traveler ends up in an endpoint in the goal region. If there is no rich subtree, then each point t in the subtree has at most one successor u whose subtree is rich. For, if there were two such successors, then there is a rich subtree at s after all. But then, Demon can cut the connection to this single rich subtree (if one exists at all) and force Traveler to a situation without a rich subtree. In the end, this forces Traveler into an endpoint outside of the goal region. – Note that, if Traveler can move back and forth in the tree, this outcome changes. Suppose s is not a winning node for Traveler, but there are three sibling nodes that are winning. Then, no matter how Demon cuts, Traveler can move back one step, and then still reach a winning node.

[5] The existence of a rich subtree is not definable in basic modal logic, since one can always create rich subtrees modulo bisimulation by duplicating parts of trees. It is not even first-order definable, Valentin Goranko (p.c.). How complex is this condition to check on a given graph?

Sabotage in grids In trees, travel decisions are irrevocable. Choosing a daughter node makes all sibling nodes and their subtrees inaccessible. An almost opposite structure is that of a grid, where any two paths from a current node can still converge to a common point by suitable further moves. For a concrete example, consider a geometric 8-point *Cube*, with a goal region of two points. [6] These points must lie on some face of the cube. Now the game acquires a concrete geometrical meaning. Clearly, Traveler has a winning strategy in some vertices of this face, depending on where the goal points lie, but Demon has a winning strategy when Traveler stands at the opposite four vertices. When the goal region contains three points, Traveler may or may not have a winning strategy everywhere, depending on the distribution of the goal points.

Logic games once more As with the travel game, one can also view the sabotage game as a semantic evaluation game for special formulas of sabotage modal logic with alternating travel and deletion modalities, or in infinite versions, for similar formulas of the sabotage μ-calculus.

Our examples show how design of graph games and design of matching logics go hand in hand, both with fixed and with changing graphs. Of course, one can also study the above and our further graph games on their own, using techniques from computer science or game theory, but the special focus in this paper will be on this game–logic connection: how it works, and what light it sheds. [7]

2 Modified scenarios

Next, we look at a few concrete variations on travel and sabotage scenarios to see how the match with logics persists, but in a fine balance that raises interesting further questions. The examples to follow here will not be discussed in as much detail as the above games, given our more methodological purpose.

2.1 Modified travel games

Two graph travelers This time, both players are localized, so the game starts in a position $(\mathbf{G}, s, t)$ with a starting point s for A and t for E. We can assume that A starts, but we could also let the two players move simultaneously.[8] Now we need to stipulate a winning condition, or rather a goal. Let us say that, for both players, the goal is to keep going forever. This is not a zero-sum game: infinite histories for both agents are a draw – though we might stipulate that, if one agent gets stuck and the other can continue, that first player loses.

[6] This classical geometric setting is also a good antidote to a hidden bias in the literature toward using only simple two-dimensional ‘planar graphs’ in sabotage scenarios.

[7] This is not to say that the connection between graph games and ‘matching’ logics is entirely unproblematic, and we will also raise some conceptual problems in Section 4.

[8] Our examples in this paper are mainly cast as sequential games: but it would make sense to also consider versions that allow simultaneous moves for the players.

This simple modification breaks the game content of the travel game, since players can no longer influence the steps made by other players. Even so, the earlier connection with modal logic remains.

The natural models are now graphs $(\mathbf{G}, s, t)$ with two distinguished points, and the natural modal language is a 'two-dimensional' one, with proposition letters true at single points. Most essential then are two 'travel modalities':

$$(\mathbf{G}, s, t) \models \langle \text{left} \rangle \varphi \quad \text{iff} \quad \text{for some point } s' \text{ with } Rss',\ (\mathbf{G}, s', t) \models \varphi$$
$$(\mathbf{G}, s, t) \models \langle \text{right} \rangle \varphi \quad \text{iff} \quad \text{for some point } t' \text{ with } Rtt',\ (\mathbf{G}, s, t') \models \varphi$$

These two modalities allow us to express the above observations. For instance, in a given finite graph with starting position (s, t), the first player loses if, for some natural number k, the formula $[\text{left}]^k \bot$ holds at s, while $[\text{right}]^k \bot$ does not hold at t. Moreover, the existence of infinite paths for players can be expressed by μ-calculus formulas of the form $\nu p \cdot \Diamond p$.[9]

The basic modal logic for these models is still decidable, and does not differ all that much from a standard minimal modal logic, [39].[10] One way of seeing this is to note that the language we have introduced can be translated into the two-variable fragment of first-order logic, which is known to be decidable.[11]

A natural variation of this scenario would give both players their own goal regions that they are trying to reach. Similar points can be made about the loss of the earlier game character, but the remaining link with modal logic. In our later versions of goal-reaching scenarios, the game character will be reinstated.

Remark on interaction The two-traveler game raises an interesting general issue. *What makes a game scenario on graphs interactive?* Is it a necessary condition that one player's moves modify or restrict those of the other player? This seems overly strong: even a travel game has interaction in the form of players's strategies influencing optimal play for other players. Another dimension of interaction is the pay-offs. When these are completely independent, there is less interaction than when the goals of the scenario have the players entangled.

Meet/avoid game We investigate the second aspect of interaction, entangled goals, with a slight modification of the travel game with players at two different positions. This time, let A win the game when the two players meet at the same point, while E loses then. So, one player wants to avoid the other, like in children's games of 'Catch' or in 'Cops and Robbers', [36].

Now there can be more interesting game scenarios, and it is easy to see that E's winning positions are correlated with the existence of suitable cycles in the graph. It is also easy to see that the game is still determined, since it has an open winning condition for A in the sense of the Gale-Stewart Theorem.

[9]However, there appears to be no single generic formula in the basic modal language, or even in the modal μ-calculus, that expresses the property of reaching the goal first.

[10]There are much more complex modal 'product logics', [17], but these are not needed here.

[11]The crux is that, working with just variables x, y for our pairs, bounded existential quantifiers for travel modalities at a point can reuse the variable marking the other point.

As for a logic counterpart of this modified game, we can have the same two-dimensional models as before, and the same modalities. However, the new goal now calls for one new propositional constant I such that

$$\mathbf{M}, s, t \models I \quad \text{iff} \quad s = t$$

In other words, I denotes the identity relation in a graph model $\mathbf{G}$. Then a typical statement about winning positions for E will be of the form

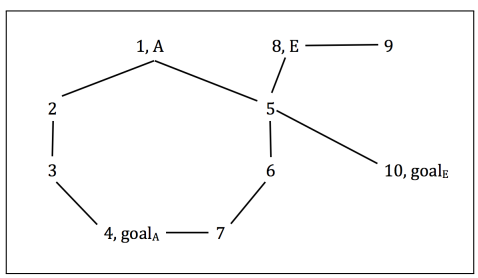
$$[\text{right}]\langle\text{left}\rangle\neg I$$

We believe that this extension of two-dimensional logic is still decidable, even though it gets close to two-dimensional 'compass logics', [32]. But there is a catch. It is not clear how to keep the translation of the above language inside the two-variable fragment of first-order logic, since we need to compare three points: the originally matched s, t, but also the new point s' reached.[12]

Occupation game Our final travel game lets players occupy parts of graphs, which is reminiscent of many popular board games, such as 'Settlers of Catan'.[13] Each player has his own color, and he automatically colors all points that he visits in his own color if no color was assigned yet. When a player reaches a point with the color of the other player, he loses. Again we can set this up with 'keep going' as a winning condition, or with reaching a specified goal region, one for each player. In what follows, we consider the latter format.

Example Outrun or obstruct?

Here is a graph for a scenario like this.



Let the game start with player A moving. Starting in 1, A could reach his goal point 4 in three steps, but that would give player E enough time to reach her goal point 10. So, what A should do is move to 5, thereby blocking E from ever reaching her goal– even though this is not A's shortest path to his goal.

[12]There is an interesting analogy between the meeting game and modal bisimulation. A bisimulation is a relation Z between points s, t in two models $\mathbf{M}$, $\mathbf{N}$ such that each step from s to an accessible point s' in $\mathbf{M}$ can be matched by a step from t to an accessible t' in $\mathbf{N}$. Here 'matched' means that $s'Zt'$ holds. The game version of bisimulation is much like our meeting/avoiding game. Now it is known that the notion of bisimulation cannot be expressed with two first-order variables, [11], whence its meta-theory is somewhat complex.

[13]In other widely played parlor games, such as Halma or Chinese Checkers, barriers of occupied points can be jumped over, but we will not consider such variations here.

It would be of interest to analyze winning strategies in occupation games in graph-theoretic terms, as they seem closely related to fundamental results such as Menger's Theorem on the existence of minimal barriers in networks, [15].

The occupation game also looks like a modified version of the 'poison game' in the graph-theoretic setting of [16] whose corresponding modal logic has been investigated in [33] – be it that both players can poison positions now.

As for a matching logic, to deal with the dynamics of progressive occupation, we need proposition letters for both players whose extension grows as travel steps are made. This requires introducing, on top of the basic modalities that correspond to steps in a graph, at least a further dynamic modality

$$[+p]$$

which records what holds when we make atom p true at the current point:

$$\mathbf{M}, s \models [+p]\varphi \quad \text{iff} \quad \mathbf{M}\text{-with-}p\text{-set-to-true-at } s, s \models \varphi$$

This local valuation change looks like a simple and harmless addition to basic modal logic. However, using an embedding of a known undecidable hybrid logic, [41] shows how basic modal logic extended with $[+p]$ modalities is undecidable.

But the more general, and more explanatory, analogy here is with the 'memory logics' of [6], [4], which allow a growing number of points in a graph to be stored in memory. The relevant known theory of these logics includes both general undecidability proofs and decidability results for special fragments. [14]

Having seen how natural variations on the original travel game sustain a logic connection – though one where slight changes may require big logic adjustments – we now turn to some natural modifications of the earlier sabotage game.

2.2 Modified sabotage games

Sabotage game with localized demon Players in the sabotage game are asymmetric in their powers. Traveler is 'local' in the sense of being restricted to her current position, but Demon is 'global', as his moves can be anywhere in the graph. While this asymmetry of the players is right for modeling some scenarios, such as general disruption of a communication network, many situations in real life have both players localized in the graph. For concreteness, think of a war game, where Traveler is one party, and Demon the enemy. In that case, we can let Demon cut at any point where he stands, but he may also make travel steps to get to other positions and cut there. Note that, unlike in the travel scenario, this localization of both players does not necessarily erase the game structure.

[14] This variety of relevant logics, whose mutual relations are sometimes not so easy to establish, also shows that our matching of games with even just modal logics is not an automatic process, and that there may even be significantly different logics for the same game. For a recent example of such options concerning modal poison logics, cf. [7].

Example Localized travelers and demons.

Recall our sabotage game in Section 1.2. Assume for concreteness that in each round, Demon first cuts and then makes a step along some edge. For an illustration, if Demon stands at the top left, he can still win as follows. First cut one link to the right, then go to the exit point E. Now, whatever Traveler does, Demon can cut a connection at E preventing Traveler from reaching the exit point from where he stands, and he has time to cut the third link if Traveler tries somewhere else. However, if Demon is localized at the bottom right when the game starts, it is Traveler who has the winning strategy.

Clearly, this new game restricts the powers of Demon in the original game. Locality changes things considerably. [42] gives a polynomial time solution algorithm for local sabotage games, in contrast with the Pspace solution complexity for the global sabotage game. Interestingly, on the logical side, locality induces a significant change in the earlier modal fixed-point definition

$$\nu p \cdot (\gamma \vee \Box\bot \vee \blacksquare \Diamond p)$$

for Traveler's winning positions in the sabotage game. Instead of using the dynamic link deletion modality, we can make do with the following formula:

$$\nu p \cdot (\gamma \vee \Box\bot \vee \langle\geqslant 2\rangle\, p)$$

Here $\langle\geqslant 2\rangle\varphi$ is a static graded modality stating the existence of at least two successors of the current point satisfying φ. [15] Proving the equivalence between these two modal fixed-point formulas requires care with comparing winning strategies for Traveler at the same point in different graphs, and the argument typically seems to fail for the global sabotage game, since the solution complexity for the latter is Pspace, [30].

It remains to be seen whether this reduction to static μ–calculi extended with suitable modalities also works for other localized graph games. [16]

Fully symmetric sabotage game An even more realistic version makes players symmetric in their powers: they can now both delete links at their current position and decide to move from there, or vice versa, 'burning bridges'. In this symmetric set up, both players can have goal regions, perhaps disjoint ones in special settings such as war games. [17]

[15] The number 2 here reflects the out-degree 2 of the 'rich subtrees' of Footnote 3, where local sabotage had the same power for Demon as global sabotage.

[16] There are other natural ways of changing the sabotage game. Say, instead of deleting links, we might also let players delete points and all links leading to them. This makes Demon much more powerful than in the original game. Even so, the logical theory of point deletion shows many similarities with that of link deletion, cf. [14], and for a broader study connecting up with hybrid logic and first-order logic, [3]. Another natural variation would allow cutting of all links according to a definable rule in a graph. In a uniform global case, this is like link cutting in dynamic-epistemic logic. But the theory of a more realistic local variant shares many features with other complex logics discussed here, [26].

[17] Such variations need not be a priori, they are just as well suggested by actual board games that are reminiscent of the sabotage game, such as 'Quoridor', [20].

Example Local sabotage with two goals.

Consider the four-point graph for the original sabotage game in Section 1, with players T and D in various initial positions, and the goal point for T bottom–right, and for D top–left. We first need to stipulate moves. For concreteness, take 'go–cut': first move along an edge, then cut some link there, if you want.

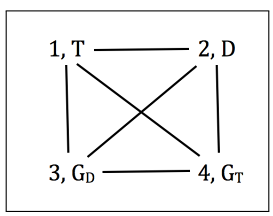
But more interestingly, whatever moves we allow, this new setting obviously invites more refined goals beyond winning and losing. Naturally, both players prefer being in their goal region, while the other is not, slightly worse is both being in their goal region, next down is neither in their goal region, and clearly worst is having the other in his goal region while you are not.

With this refined goal structure, we encounter a significant transition. Despite not just being about winning or losing, the above scenario is still a perfect information game, and it can be solved by the standard game-theoretic method of Backward Induction. Now we get non-trivial Nash equilibria, whose definition involves powers of both players, not just one.

We give a simple concrete illustration, though for convenience, we do not consider a sabotage game, but an occupation game in our earlier sense.

Example Graph games with preferences.

Consider the following graph, with players initial positions and goal points as indicated. The initial positions where players are standing count as occupied. Let us assume this time, for convenience that Traveler starts.



One can easily draw an extensive game tree for the two moves of both players: 'straight down' and 'diagonally down'. It is easy to see that each move by T leaves only one option for D. If T goes straight down, then so does D, and the outcome is that neither reaches his goal. If T goes diagonally down, then so does D, and both players reach their goal. Following the Backward Induction solution procedure then tells us that the latter outcome will be reached.

This example is extremely simple, but it should be easy to see that much more complex graph games with preferences can be solved in the same way.

We are at a significant point here in the matching of graph games and logics. What resources are needed to analyze the Backward Induction style of reasoning? Our modal logics can still formulate goals, and also preferences, at least, when these are defined by simple enumeration of propositional cases like above, cf. the format of priority graphs in [27]. But defining strategy profiles that form Nash equilibria seems to require further resources, coming from richer game logics than mere graph logics. We will return to this issue in Section 4.

3 Parameters for game design

Having seen a budget of graph games, let us summarize some main features. There are two parameters for a classification: general game structure (moves, turns, goals), and graph structure: the board on which the games are played.

Modulating moves In choosing moves for graph games, we found options that impact properties of the games, but also of logics describing these games, [8], [28]. One pervasive distinction is that between *local versus global moves*: whether players are localized in the graph, like Traveler in the sabotage game, or can range at random, like Demon in that game. A second distinction is whether moves are *arbitrary, or definable*: can Demon delete any links, or should he follow some definition? This is related to another distinction: deletion moves can be *stepwise or uniformly defined.* And in making these comparisons, we even crossed another divide: moves can *stay within a graphs, or jump to a changed graph.* These choice points are not totally independent, but they span a space of design options, whose systematic exploration seems well worthwhile. [18]

Ways of winning and losing In simple graph games, goals for players are *independent unary properties* defining points that they must avoid, or regions that they want to be in. However, we can also have *entangled goals*, as in the earlier meeting/avoiding game, itself a variant of so-called 'pursuit-evasion' games on graphs that have been widely used in mathematics and computer science, [38]. Here Traveler loses if she meets with Demon, Demon then wins: this is not a Boolean combination of separate independent unary goals for Traveler and Demon, but a binary relation between their positions. Finally, there may even be still higher-level *procedural goals*, for instance, about how players travel toward the distinguished goal region. For instance, they may not be allowed to visit the same intermediate points twice, or, in occupation games, they may not visit the path visited by the other player. And we also saw how general specification of goals can lead to games that require Backward Induction or other ways of getting at Nash equilibrium outcomes.

Game boards In all this, we may have to loosen our focus on graphs. Our graphs served as game boards, comparable to those of parlor games. But the local states traversed by the above games usually require further structure. At least, we need to indicate the ambient location of players moving through the graph. Moreover, with our more complex occupation scenarios, we need additional propositional information at nodes telling us whether they have been visited already, or even, how often. In general then, the game boards are not just graphs, but 'annotated graphs', where these annotations can be game-induced, but might also be game-external properties of vertices used in defining moves.

Several more general issues are suggested by a classification like this.

[18] There are many further possible moves in graph games than those discussed here. A quick survey of common parlor games reveals of host of further types, having to do with how players can move counters, jump across barriers: in single steps or in iterated sequences, and so on.

Changes and thresholds in behavior Localizing players, or changing their available moves has obvious effects on our simple examples of graph games. However, when are there general effects, and how do we measure these? Say, localizing Demon seems a major shift in favor of Traveler in sabotage games, but how can one make such intuitions precise?

One way of doing this might be to determine independent graph-theoretic properties that correspond to winning positions, such as the association of winning positions for Traveler with 'semi-kernels' in poison games, [16].

Another approach would involve a general analysis of computational complexity: which design changes cross a threshold, making solution complexity a step harder, or simpler, in the complexity hierarchy?

Finally, there can also be other parameters that determine behavior of games. For instance, winning conditions are often 'monotonic': some player only gets better off when we increase the goal region, or the supply of accessible links. Can we also get general insights by restricting to monotonic goals?[19]

When are two graph games equivalent? However, perhaps the most obvious general issue behind our discussion is this. Are there hidden reductions running underneath the playground of designing new games that we have entered? How much variety is there really? *When are two graph games equivalent?*

Many examples in our presentation carry this suggestion. For instance, travel games with occupation make nodes inaccessible to other players, and as such, they are also sabotage games that delete points. Can this Gestalt switch be turned into a precise equivalence? And when all is said and done, even our prima facie clear distinction between travel moves that stay within a given graph and sabotage moves that change the graphs can be questioned. Both moves stay inside a richer 'meta-graph' of different graphs and distinguished points, a perspective that is in fact quite congenial to logics of sabotage games.[20]

4 Logics for graph games

Having seen the design choices for graph games, we now turn to matching logics.

Modal logics We started with modal languages having enough expressive power for describing simple games, and then moved to more expressive formalisms, sometimes rather mild ones such as two-dimensional modal logic. However, not all these logics were standard, and we found interesting open problems. In particular, logics for graph games always contained the basic standard modalities, the difference usually lies in the dynamic modalities. However, there are unresolved issues here, since the expressive surplus of dynamic modalities can often be calibrated in terms of hybrid logics, [5]. And in the limit, first-order logic itself could serve as a language of graph change, [8].

[19] There are precedents for this in other logical literature on game design, [1].

[20] In its turn, this meta-perspective might suggest a generalized 'protocol semantics' where models may constrain available graph changes, [12]. This widening of the standard models has been used to lower complexity of dynamic-epistemic logics, and it might also improve our earlier treatment by lowering the currently high complexity of logics of graph change.

Moreover, these modal languages are not enough to capture game-theoretic notions such as winning positions uniformly. For that, we had to turn to modal fixed-point languages such as the μ-calculus – either in its basic format, or in the more subtle formulation needed when adding modalities for graph change. We have seen how these raise interesting new technical issues, such as when winning positions can be defined alternatively in suitably extended static μ-calculi. [21]

However, toward the end of our discussion we hit a significant border line. When goals or preferences become more sophisticated than winning or losing, real game-theoretic considerations enter concerning various equilibria, and our emphasis on logics for graphs reaches its limits. Richer formalisms are needed.

From graph logics to logics for explicit game structure The other obvious candidate for describing graph games, now in more procedural detail, are standard 'game logics' for analyzing game structure, [10]. These languages are rich, involving moves, preferences, and strategies, and they are usually interpreted on games, either in extensive form or strategic form – or in some cases, on associated models of powers that players have in the game.[22] A typical benchmark for such logics is whether they can describe standard game-theoretic solution procedures such as Backward Induction or Iterated Removal of Strictly Dominated Strategies. And indeed, we encountered the need for the former toward the end of Section 2. And contemporary game logics do not even stop here, they can also deal with more complex scenarios such as players of different abilities playing the same graph game, using type spaces or even richer devices.

In our view, graph games with complex preferences and procedural conventions eventually need to be studied using game logics.[23] However, this still leaves something to be explained. For a surprisingly long time, our discussion got by with austere logics for graphs. What was going on?

Graph invariants for game logics Graphs may be considered a game boards associated with graph games. Now sometimes, strategic properties of games can be expressed purely in terms of the game board. Game theory has many examples of such 'invariants': think of the simple numerical invariant indicating who has the winning strategy in the game of Nim, [37]. Now, our logics for graphs may be viewed as defining invariants that capture the strategic structure of at least simply defined graph games.[24]

It is a very interesting question when such reductions are possible. A deeper study would lead us into results in computational logic on 'memory-free strategies', such as the Positional Determinacy Theorem for parity games, [35].

[21] Even so, this is just one line, and the emphasis on modal logics in this paper may be a bit biased. If our focus had been on infinite graph games, then an alternative attractive formalism would have been that of *temporal logics*. These, too, can be very useful for defining complex behavior and strategies over time.

[22] For an up to date survey, see the Stanford Encyclopedia entry [13].

[23] A simple illustration is that a position in a graph does not tell us how the game will go until we know which player has the turn (a game-internal notion) and needs to move there.

[24] Slightly richer invariants would arise on annotated graphs that encode some procedural game-internal information by means of additional unary properties of nodes.

Here is another well-known example of such a reduction, now inside standard logic. Consider evaluation games for logics, say, Hintikka games for first-order logic, [10]. The typical adequacy result one proves about such games is that the 'truth player' Verifier has a winning strategy for the game concerning a formula φ on a model $\mathbf{M}$ if and only if φ is true at $\mathbf{M}$. But reversing directions, the result tells us that the strategic structure of evaluation games is captured by game-free invariants purely at the level of the model.[25]

It is not our aim to solve all technical problems concerning modal logics of graph games and richer game logics. But hopefully, our discussion did show the interest of taking the joint design stance that we have pursued here. We conclude with a few more general perspectives on this match.

Uses of logic Using logic to describe games has obvious benefits. One can use the logic to express general qualitative reasoning patterns about the interactive scenarios that the games are intended to model. One can use some standard techniques such as model checking to verify that given games have specified properties. But these are generic benefits. In the present setting, we would be most interested in new connections with a special thrust toward games.

One interesting example of this kind concerns the general strategic situation in graph games. It is a significant question concerning many games which player has the systematic advantage, say, the opening player or the responding player. In our present setting, one way of making this issue more precise is as follows.

What happens to the winning positions, say, in the sabotage game, as the size of the finite graphs goes to infinity? Now there is a link with logic here. The *Zero-One Law* for first-order logic says that, for each first-order sentence in the language with a binary relation, its probability of being true goes to either 0 or 1 as graph size goes to infinity. It can even be decided effectively, for any given formula, which of the two options occurs. And this Zero-One Law has been extended to LFP(FO): first-order logic with fixed-point operators, in [25]. Now, we were able to express the winning positions in the sabotage game in the sabotage μ-calculus, which translates into LFP(FO). Therefore, anything we can formulate in that language satisfies the Zero-One Law. Now this does not immediately apply to winning positions, as these refer not just to graphs, but to graphs with distinguished points, for which the Zero-One Law fails. However, here are some assertions that do fall under pure LFP(FO): existence of winning positions for both players, the statement that each winning graph position for Traveler must be connected to at least one other winning position for Traveler, and so on. We believe that this offers a fresh, and to our knowledge, unexplored perspective on the logical analysis of graph games. [26]

[25] To see how remarkable this is, it suffices to make some obvious variations on well-known evaluation games, and observe how an immediate match with standard logics vanishes.

[26] After the first version of this paper started circulating, a solution to the question posed here has been found using the countable random graph, where truth coincides with truth almost everywhere in finite models. [34] shows that, if the goal region has at least two points, all positions in the random graph are winning for Traveler – making the sabotage game massively favorable to Traveler. For special model classes, however, the result remains open.

The match viewed from both sides The connection between logics and games advocated in this paper can be looked at from two sides. So far, we have gone *from games to logics*, presenting a number of ways of achieving this. Here one obvious question is whether this can be made more systematic, say in the form of automatic methods for turning our earlier definitions for graph games into matching logics. There is a system to how one creates these matches, but we have not been able to see the full generality yet.

But there is also an opposite direction going *from logics to games*. In fact, there has long existed a well-established tradition of designing games for various logical purposes that goes back to the 1950s in mathematical logic, and to many centuries earlier in the dialogical debate tradition in various logical cultures. This historical tradition exemplifies the opposite direction to this paper: using logics to design games. [27] Notably, as with the direction from games to logics, there need not be one unique correlation here. For instance, logical languages are associated with evaluation games, but also with proof search games, or with games for comparing models, such as Ehrenfeucht–Fraïssé games. The latter may throw light on graph games, too.

In general, it is a delicate issue how these two perspectives of 'logics for games' and ''logics as games' are related, [10], and many open problems remain. But as we have noted already in Section 1, our graph games have a bit of both, since they suggest logics for their description whose evaluation games, at least for certain well-chosen formulas, are those very graph games. This is one special case of the general question what happens when we pursue cycles of the form

'game, logic of game, game for logic of game', or

'logic, game for logic, logic for game of logic'.

These cycles need not always stabilize, but graph games may be an attractive arena where they do. Getting a bit ahead of the evidence that we presented in this paper, we would like to think of graph games as a natural place where the usual tension between game logics and logic games disappears. One take on this might be that our graph-changing games really are already logical evaluation games, but then games where the model changes in the process of evaluation, as has been suggested for first-order semantics in terms of drawing objects with or without replacement, [3], and as seems even realistic in logical modeling for quantum-mechanical measurement. [28]

A caveat: logical machinery We conclude by pointing out that our matching also has its problems, or limitations. Our logical languages are much richer than the set of assertions one would normally make about graph games, or just the

[27] In a broader perspective, the logic and games connection pursued here would also need a third component, namely, the study of games in computer science. These games go back a long time, cf. [2], [22], and they would form a natural complement to our discussion here.

[28] There is much more to the match between graph games and evaluation games than what we have probed here, especially, when one tries to correlate equivalent formulations of the same graph game with equivalent formulas in matching fixed-point logics, cf. [23].

particular graph game we started with. This surplus may just be ballast. [29] Have we been pursuing a machinery that is much too heavy for graph games?

For a start, it is easy to overstate this objection. To a student of some particular inference patterns, logicians usually offer a complete formal system. This seems largely harmless, like visiting a store where one buys a lot of medical supplies even when there is just this one bandage needed for now. Presumably the logical system can handle lots of games similar to the original one.

But we can go one step further in the present setting. It may be possible to find a game interpretation for the whole logical language that is congenial to the original graph games. For instance, evaluation games can be defined for arbitrary formulas of sabotage modal logic. These games represent an infinity of *related graph games*, with different schedules and winning conditions. Say, the game for $\Box \Diamond \blacksquare \Diamond \gamma$ defines a simple mixed travel-sabotage game where Traveler must first move to some safe point for her against a sabotaging Demon.

Even so, the question remains whether this infinite class of formula-induced games contains interesting ones beyond the original motivating scenarios.

Finally, related to the preceding point, we are not claiming that every property of our logics has immediate import for graph games. For instance, while logicians tend to emphasize computational complexity of logical systems, this merely tells us that the *theory of* certain graph games may be complex – not that solving these games itself is a complex algorithmic task, [9].

5 Digression: Imperfect Information

One further test on the perspective offered here would be how well our analysis stands up to one very realistic feature of many games that we have ignored so far. Many games involve *imperfect information*, not necessarily in the sense of bounded rationality, which is of course unavoidable in practice, but in the sense that even ideal players may still have limited powers of observation in the graph. In contrast, our whole discussion has been about perfect information games.

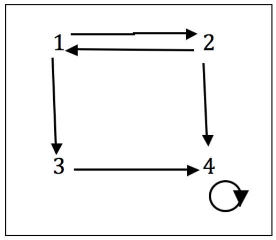
In this section, that can be skipped without loss of continuity, we briefly discuss some graph games or game-like scenarios that involve imperfect information, and the resulting partial information flow during play.

Catch game with questions Suppose that a Traveler moves through a graph in which each point has a successor, one step at the time, and an Observer tries to catch him. In each round, the Observer can ask one question of the form "Are you at point x in the graph?". If Traveler is at that point, the game is over, otherwise, Traveler makes a step again, and so on.

[29] In some cases, the ballast may even be misleading. For instance, it is emphasized in [8] how the valid equivalence $(\Diamond\top \wedge \blacksquare \Diamond p) \leftrightarrow \langle \geqslant 2 \rangle p$ of the modal logic for local sabotage is a fluke, in being not schematically valid. But as we saw in Section 2, for just defining the winning positions of Traveler, this equivalence was the right observation to make.

Depending on the graph, either Traveler or Observer will have a winning strategy. For instance, evidently, in a three-point graph with the universal relation, Observer can never be sure to catch Traveler.

But there can be more interesting cases, as shown in the following figure.



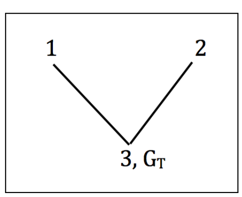
Here Observer has a winning strategy. How many questions does he need? Say, he asks in the following order: 1, 4, 3. The answer to the first question might be negative, so Traveler is in {2, 3, 4} and the next hidden move then takes Traveler into {1, 4}. Suppose the answer to the second question is negative again: then Traveler is at 1, and his next move takes him into {2, 3}. If the answer to the third question is negative once more, Traveler is at 2, so his next move takes him into {1, 4}. Observer still does not know where Traveler is.

However, there is a much better strategy for Observer: just ask *twice* whether Traveler is at 1. As is easy to see, this will pinpoint Traveler's location.

This question scenario looks like a standard information-theoretic one, but the dynamics of moving in graph games add an unexpected flavor. This can be described in a logical framework combining a modal logic for travel steps with an *epistemic logic* for what Observer knows. Worlds in the models for this epistemic logic consist of the above graph with various locations for Traveler. Information flow can then be modeled by dynamic-epistemic techniques, [9].

Games like this resemble 'fugitive problems' from graph theory and computer science: cf. [21] for background and structural characterization results.

Sabotage with imperfect observation The sabotage game, too, suggests natural variations with imperfect information. We can make Demon ignorant of where Traveler is, or vice versa, or we can let both players be ignorant of the precise location of the other. For a very simple example, merely to show what additional things can happen in contrast with the perfect information version, suppose that the graph is as follows, where 3 is the goal point of Traveler.



Demon does not know where Traveler is, but does get some information in each round: Traveler must announce whether he is in his goal point. Suppose that no such announcement is made at the start. Then Traveler must be in point 1 or 2. Demon does not know which of the two – so, whichever link he cuts, there is the possibility that Traveler may reach the goal. Clearly, no player has a winning strategy. Demon might happen to cut the right link, so Traveler is stuck. But Demon has no cutting strategy that guarantees him a win.

All this is typical for games of imperfect information. To be sure, there is still a tight connection with logic here. The information flow in sabotage games with imperfect information of various sorts, generated by many different scenarios, can be described completely in dynamic-epistemic logics, [9], with again, graphs with various positions for Traveler and various cuts made by Demon connected by epistemic uncertainty relations.[30]

But what changes is that equilibria can now be probabilistic, involving mixed strategies. For instance, in the above game, one equilibrium consists of Demon choosing a link to be cut with probability 0.5, while Traveler moves to his goal point if he can. Thus, our graph logics must be supplemented with probabilistic considerations when studying definability and computability, [13].

These are just a few examples of the pervasiveness of imperfect information. Many further natural variations on our graph games deserve investigation [31] Also, as we have noted, the logic connection does not disappear, but appropriate game logics now require more complex languages and models bringing in techniques from dynamic-epistemic logic. The connection with logic games does not go away either, but one would now have to look at logic games with imperfect information, such as those proposed in frameworks like IF logic, [31].

6 Conclusion

This paper has presented a way of thinking about graph games as a natural meeting place with logic. In doing so, we are not claiming any exclusive rights: graph games can equally well be studied in terms of game theory, probability theory, computer science, experimental cognitive science, and so on.[32] But a logical perspective adds a special flavor, as our discussion has tried to show, that leads to many new questions. Even so, we acknowledged that the matching of graph games to logics advocated here is not unique, and more importantly, that a complete logical system may be a heavy machinery to associate with a single concrete game. That machinery should be used with care, and taste.

[30] One concrete way in which players might have limited observational powers in graph games is through 'short sight' in the sense of [24]. We might assume that localized players only see the graph up to a certain reachability depth around them. Again, this dynamics can be described in dynamic-epistemic terms, [29].

[31] Other scenarios with imperfect information that may be connected to graph games are the Boolean Network Games of [40], with its connections with the epistemic friendship logic of [19]. The social network dynamics in this work is not the same as the graph dynamics in the present paper, since agents do not move, but can only change local properties triggered by those of their neighbors – but a serious comparison seems in order.

[32] These approaches can also be combined in felicitous ways, witness the joint methodology of logical analysis of agent types, simulations, and cognitive experiments pursued in [18].

To back up our proposed methodology, we have shown by example how one can analyze and design games at the same time in tandem with ideas from logic. This unity of analysis and design is by no means new. It is a classical theme in computer science, in constructive mathematics, and in Euclid's geometry. We are merely taking a grand tradition to a compact, but hopefully rich arena.

Finally, if we had to point out one omission in our program, it is our a priori stance toward design. As we have noted a number of times, our graph games look like actual games that are played in reality, or like simulation games for practical purposes. In fact, society is one big laboratory where new games are designed all the time. A natural complementary challenge to what we have proposed is linking up with the real world of games and gaming. Obviously, this world is not quite that of this paper, since real games are played on fixed boards, and the variety in their play comes from other factors, such as chance moves, or cognitive bounds on players. But this does not mean that no significant links can be made: it is just that we have nothing to offer at this stage.

References

[1] Ågotnes, T. and H. van Ditmarsch. What Will They Say? Public Announcement Games. *Synthese KRA*, 179(1), 57–85. 2011.

[2] Aigner, M. and M. Fromme. A Game of Cops and Robbers. *Discrete Applied Mathematics*, 8(1), 1–12. 1984.

[3] Areces, C. and J. van Benthem. Stepwise PAL and the Logic of Quantification Without Replacement. Department of Philosophy, Stanford University and Department of Computer Science, University of Cordoba. 2018.

[4] Areces, C., Carreiro, F., Figueira, S,. and S. Mera. Expressive Power and Decidability for Memory Logics. *Proceedings WoLLIC 2008*, Lecture Notes in Computer Science 6642, Springer Verlag, Berlin, 20–34. 2011.

[5] Areces, C., Fervari, R., and G. Hoffmann. Relation-Changing Modal Operators. *Logic Journal of the IGPL*, 23, 601–627. 2015.

[6] Areces, C., Figueira, D., Figueira, S., and S. Mera. Basic Model Theory for Memory Logics. *Proceedings WoLLIC 2011*, Lecture Notes in Computer Science 5110, Springer Verlag, Berlin, 56–68. 2008.

[7] Areces, C., Mierzewski, K., and F. Zaffora Blando. Poison Logics and Memory Logics. Manuscript, Department of Philosophy, Stanford University. 2018.

[8] Aucher, G., van Benthem, J., and D. Grossi. Modal Logics of Sabotage Revisited. *Logic and Computation*, 28(2), 269–303. 2017.

[9] van Benthem, J. *Logical Dynamics of Information and Interaction*. Cambridge University Press, Cambridge UK. 2011.

[10] van Benthem, J. *Logic in Games*. The MIT Press, Cambridge MA. 2014.

[11] van Benthem, J., B. ten Cate and J. Väänänen. Lindström Theorems for Fragments of First-Order Logic. *Logical Methods in Computer Science*, 5(3), 1–27. 2009.

[12] van Benthem, J., J. Gerbrandy, T. Hoshi and E. Pacuit. Merging Frameworks for Interaction. *Journal of Philosophical Logic*, 38(5), 491–526. 2009.

[13] van Benthem, J. and D. Klein. Interfaces of Logic and Games. *Stanford On-Line Encyclopedia of Philosophy.* 2018.

[14] Chen, Y. Model Theory for Modal Logics of Definable Point Deletion. Department of Philosophy, Tsinghua University. 2018.

[15] Diestel, R. *Graph Theory.* Springer Verlag, Heidelberg. 1997.

[16] Duchet, P. and H. Meyniel. Kernels in Directed Graphs: A Poison Game. *Discrete Mathematics*, 115, 273–276. 1993.

[17] Gabbay, D. and V. Shethman. Products of Modal Logics. *Logic Journal of the IGPL*, 6, 73–146. 1998.

[18] Ghosh, S. 2018. Strategizing: A Meeting of Methods. Indian Institute of Statistics, Chennai. Under submission.

[19] Girard, P., F. Liu, and J. Seligman. Logical Dynamics of Belief Change in the Community. *Synthese* 191, 2403–2431. 2014.

[20] Glendenning, L. *Mastering Quoridor*. D.Sc. Thesis, University of New Mexico, Albuquerque. 2005.

[21] Goranko, V. A Characterization of Observer-Fugitive Games. Philosophical Institute, University of Stockholm. 2018

[22] Grädel, E., W. Thomas, and T. Wilke, eds. *Automata, Logics, and Infinite Games: A Guide to Current Research*. Lecture Notes in Computer Science 2500, Springer Verlag, Berlin. 2002.

[23] Grossi, D. Abstract Argument Games via Modal Logic. *Synthese* 190:5, 5–29. 2013.

[24] Grossi, D. and P. Turrini. Short Sight in Extensive Games. *Proceedings AAMAS 12*, Valencia, 805–812. 2012.

[25] Gurevich, Y. and S. Shelah. Fixed-Point Extensions of First-Order Logic. *Annals of Pure and Applied Logic*, 32. 1986.

[26] Li, D. Losing Connections: Modal Logics of Definable Link Deletion. Department of Philosophy, Tsinghua University. Presented at LOFT 13, Milano. 2018.

[27] Liu, F. *Reasoning about Preference Dynamics*. Springer Science Publishers, Dordrecht. 2011.

[28] Liu, F. Choice Points for Graph Games. Seminar Note. Department of Philosophy, Tsinghua University. 2017.

[29] Liu, C., F. Liu and K. Su. A Dynamic-Logical Characterization of Solutions in Sight-Limited Extensive Games. *Proceedings PRIMA 2015*, 467–480. 2015.

[30] Löding, C. and Ph. Rohde. Model Checking and Satisfiability for Sabotage Modal Logic. *Proceedings FSTTCS 2003*, Springer Lecture Notes in Computer Science, 302–313. 2003.

[31] Mann, A., G. Sandu and M. Sevenster. *Independence-Friendly Logic: A Game- Theoretic Approach.* Cambridge University Press, Cambridge UK. 2011

[32] Marx, M. and Y. Venema. *Multi-Dimensional Modal Logic.* Springer Science Publishers, Dordrecht. 1998.

[33] Mierzewski, K. and F. Zaffora Blando. The Modal Logic(s) of Poison Games. Department of Philosophy, Stanford University. 2016. New version with co-author Carlos Areces to appear in *Proceedings Fourth Asian Workshop on Philosophical Logic*, Beijing. 2018.

[34] Mierzewski, K. Sabotage in the Random Graph. Department of Philosophy, Stanford University. 2018.

[35] Mostowski, A. Games with Forbidden Positions. Technical Report 78, University of Danzig, Institute of Mathematics and Informatics. 1991.

[36] Nowakowski, R. and P. Winkler. Vertex-to-Vertex Pursuit in a Graph. *Discrete Mathematics* 43, 235–239. 1983.

[37] Osborne, M. and A. Rubinstein. *A Course in Game Theory.* The MIT Press, Cambridge MA. 1994.

[38] Parsons, T. Pursuit-Evasion in a Graph. *Theory and Applications of Graphs*, Springer Lecture Notes in Mathematics, 642, 426–441. 1976.

[39] Segerberg, K. Two-Dimensional Modal Logic. *Journal of Philosophical Logic*, 2(1), 77–96. 1973.

[40] Seligman, J. and D. Thompson. Boolean Network Games and Iterated Boolean Games. *Proceedings LORI Taiwan*, Springer Lecture Notes in Computer Science, 9394, 353–365. 2015.

[41] Thompson, D. A Modal Logic of Local Graph Change. Department of Philosophy, Stanford University. 2018. To appear in *Proceedings Fourth Asian Workshop on Philosophical Logic*, Beijing. 2018.

[42] Zhang, T. Localized Sabotage Algorithm. Manuscript, Tsinghua University. 2018.