Decisions, actions, and games: A logical perspective

van Benthem, J.

DOI
10.1007/978-3-540-92701-3_1

Publication date
2009

Document Version
Final published version

Published in
Logic and Its Applications

DECISIONS, ACTIONS, AND GAMES, A LOGICAL PERSPECTIVE

Johan van Benthem

Amsterdam & Stanford, http://staff.science.uva.nl/~johan

October 2008

1 Introduction: logic and games

Over the past decades, logicians interested in rational agency and intelligent interaction studied major components of these phenomena, such as knowledge, belief, and preference. In recent years, standard 'static' logics describing information states of agents have been generalized to *dynamic* logics describing actions and events that produce information, revise beliefs, or change preferences, as explicit parts of the logical system. Van Ditmarsch, van der Hoek & Kooi 2007, Baltag, van Ditmarsch & Moss 2008, van Benthem, to appear A, are up-to-date accounts of this dynamic trend (the present paper follows Chapter 9 of the latter book). But in reality, concrete rational agency contains all these dynamic processes entangled. A concrete setting for this entanglement are *games* – and this paper is a survey of their interfaces with logic, both static and dynamic. Games are intriguing also since their analysis brings together two major streams, or tribal communities: 'hard' mathematical logics of computation, and 'soft' philosophical logics of propositional attitudes. Of course, this hard/soft distinction is spurious, and there is no natural border line between the two sources: it is their congenial mixture that makes current theories of agency so lively.

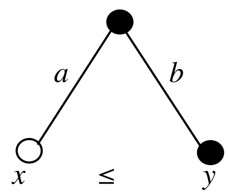
We will discuss both *statics*, viewing games as fixed structures representing all possible runs of some process, and the *dynamics* that arises when we make things happen on such a 'stage'. We start with a few examples showing what we are interested in. Then we move to a series of standard logics describing static game structure, from moves to preferences and epistemic uncertainty. Next, we introduce dynamic logics, and see what they add in scenarios with information update and belief revision where given games can change as new information arrives. This paper is meant to make a connection. It is not a full treatment of logical perspectives on games, for which we refer to van Benthem, to appear B.

2 Decisions, practical reasoning, and 'solving' games

Action and preference Even the simplest scenarios of practical reasoning about agents involve a number of notions at the same time:

Example One single decision.

An agent has two alternative courses of action, but prefers one outcome to the other:



A proto-typical form of reasoning here would be the 'Practical Syllogism':

(i) the agent can do both *a* and *b*,

(ii) the agent prefers the result of *a* over the result of *b*,

and therefore,

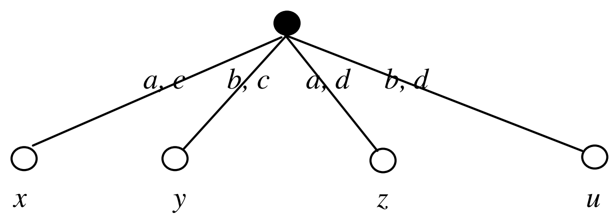
(iii) the agent will do [or maybe: should do?] *b*. ■

This predictive inference, or maybe requirement, is in fact the basic notion of *rationality* for agents throughout a vast literature in philosophy, economics, and many other fields. It can be used to predict behaviour beforehand, or rationalize observed behaviour afterwards.

Adding beliefs In decision scenarios, *preference* crucially occurs intertwined with *action*, and a reasonable way of taking the conclusion is, not as knowledge ruling out courses of action, but as supporting a *belief* that the agent will take action *b*: the latter event is now more plausible than the world where she takes action *a*. Thus, modeling even very simple decision scenarios involves logics of different kinds. Beliefs come in even more strongly when one models uncertainty about possible states of nature, and one is told to choose the action with the highest *expected value*, a probabilistically weighted sum of utility values for the various outcomes. The probability distribution over states of nature represents *beliefs* we have about the world, or the behaviour of an opponent. Here is a yet simpler scenario:

Example Deciding with an external influence.

Nature has two moves *c, d*, and the agent must now consider combined moves:

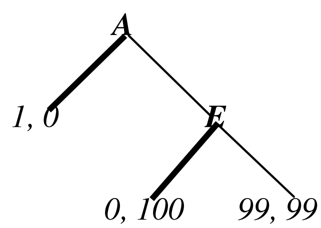


Now, the agent might already have good reasons to think that Nature's move *c* is more plausible than move *d*. This turns the outcomes into a natural 'epistemic-doxastic model' (van Benthem 2007A): the epistemic range has *4* worlds, but the most plausible ones are just *2: x, y*, while an agent's preference might now just refer to the latter area. ■

Multi-agent decision: 'solving' games by Backward Induction In a multi-agent setting, behaviour is locked in place by *mutual expectations*. This requires an interactive decision dynamics, and standard game solution procedures like *Backward Induction* do exactly that:

Example Reasoning about interaction.

In the following game tree, players' preferences are encoded in the utility values, as pairs '(value of ***A***, value for ***E***)'. Backward Induction tells player ***E*** to turn left when she can, just like in our single decision case, which gives ***A*** the belief that this would happen, and therefore, based on this belief about his counter-player, ***A*** should turn left at the start:



Why should players act this way? The reasoning is again a mixture of all notions so far. ***A*** turns left since she believes that ***E*** will turn left, and then her preference is for grabbing the value *1*. Thus, practical reasoning intertwines action, preference, and belief. ■

Here is the rule which drives all this, at least when preferences are encoded numerically:

Definition Backward Induction algorithm.

Starting from the leaves, one assigns values for each player to each node, using the rule

> Suppose ***E*** is to move at a node, and all values for daughters are known. The ***E***-value is the maximum of all the ***E***-values on the daughters, the ***A***-value is the minimum of the ***A***-values at all ***E***-best daughters. The dual calculation for ***A***'s turns is completely analogous. ■

This rule is so obvious that it never raises objections when taught, and it is easy to apply, telling us what players' best course of action would be (Osborne & Rubinstein 1994). And yet, it is packed with various assumptions. We will perform a 'logical deconstruction' of the underlying reasoning later on, but for now, just note the following features:

(a) the rule assumes that the situation is viewed in the same way by both players: since the calculations are totally similar,

(b) the rule assumes worst-case behaviour on the part of one's opponents, since we take a minimum of values in case it is not our turn,

(c) the rule changes its interpretation of the values: at leaves they encode plain utilities, while higher up in the game tree, they represent *expected utilities*.

Thus, despite its numerical trappings, Backward Induction is an inductive mechanism for generating a *plausibility order* among histories, and hence, it relates all notions that we are interested in. There has been a lot of work on 'justifying' this solution method. Personally, I am not committed to this particular style of solving games, but understanding what Backward Induction does is a logically rich subject, which can be pursued in many ways.

But for now, we step back, and look at what 'logic of games' would involve *ab initio*, even without considering any preferences at all. So, let us first consider pure action structure, because even that has a good deal of logic to it, which can be brought out as such. We will add further preferential and epistemic structure toward more realistic games in due course.

3 Games and process equivalence

One can view extensive games as multi-agent processes that can be studied just like any process in logic and computer science, given the right logical language. Technically, such structures are models for a poly-modal logic in a straightforward sense:

Definition Extensive games.

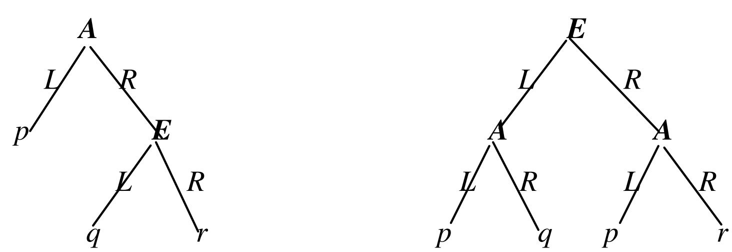
An *extensive game form* is a tree $\boldsymbol{M}$ = *(NODES, MOVES,* ***turn****,* ***end****, V)* which is a modal model with binary transition relations taken from the set *MOVES* pointing from parent to daughter nodes. Also, intermediate nodes have unary proposition letters $\boldsymbol{turn}_i$ indicating the unique player whose turn it is, while ***end*** marks end nodes without further moves. The valuation *V* for proposition letters may also interpret other relevant predicates at nodes, such as utility values for players or more external properties of game states. ■

But do we really just want to jump on board of this analogy, comfortable as it is to a modal logician? Consider the following fundamental issue of invariance in process theories. At which level do we want to operate in the logical study of games, or in Clintonesque terms:

When are two games are the same?

Example The same game, or not?

As a simple example that is easy to remember, consider the following two games:



Are these the same? As with general processes in computer science, the answer crucially depends on our level of interest in the details of what is going on:

(a) *If we focus on turns and moves, then the two games are not equivalent.*

For they differ in 'protocol' (who gets to play first) and in choice structure. For instance, the first game, but not the second has a stage where it is up to $\boldsymbol{E}$ to determine whether the outcome is *q* or *r*. This is indeed a natural level for looking at game, involving local *actions* and choices, as encoded in modal *bisimulations* – and the appropriate language will be a standard modal one. But one might also want to call these games equivalent in another sense: looking at achievable outcomes only, and players powers for controlling these:

(b) *If we focus on outcome powers only, then the two games are equivalent.*

The reason is that, regardless of protocol and local choices, players can force the same sets of eventual outcomes across these games, using strategies that are available to them:

A can force the outcome to fall in the sets *{p}, {q, r},*
E can force the outcome to fall in the sets *{p, q}, {p, r}*.

In the left-hand tree, ***A*** has 2 strategies, and so does ***E***, yielding the listed sets. In the right-hand tree, ***E*** has *2* strategies, while **A** has *4: LL, LR, RL* and *RR*. Of these, *LL* yields the outcome set *{p},* and *RR* yields *{q, r}*. But *LR, RL* guarantee only supersets *{p, r}, {q, p}* of *{p}:* i.e., weaker powers. Thus the same 'control' results in both games. ■

We will continue on extensive games, but the coarser power level is natural, too. It is like ‘strategic forms’ in game theory, and it fits well with ‘logic games’ (van Benthem 2007B):

Remark: game equivalence as logical equivalence In an obvious sense, the two games in the preceding example represent the two sides of the following valid logical law

$$p \wedge (q \vee r) \leftrightarrow (p \wedge q) \vee (p \wedge r) \qquad \textit{Distribution}$$

Just read conjunction and disjunction as choices for different players. In a global input–output view, Distribution switches scheduling order without affecting players’ powers. ■

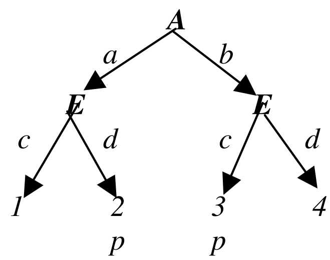
4 Basic modal action logic of extensive games

Basic modal logic On extensive game trees, a standard modal language works as follows:

Definition Modal game language and semantics.
Modal formulas are interpreted at nodes *s* in game trees ***M***. Labeled modalities $\langle a \rangle \phi$ express that some move *a* is available leading to a next node in the game tree satisfying ϕ. Proposition letters true at nodes may include special-purpose constants for typical game structure, such as markings for turns and end-points, but also arbitrary local properties. ■

In particular, modal operator combinations now describe potential interaction:

Example Modal operators and strategic powers.
Consider a simple 2-step game like the following, between two players ***A***, ***E****:*



Player ***E*** clearly has a strategy making sure that a state is reached where p holds. And this feature of the game is directly expressed by the modal formula $[a]\langle d\rangle p \wedge [b]\langle c\rangle p$. ■

Letting *move* be the union of all moves available to players, a modal operator combination $[move\text{-}\boldsymbol{A}]\langle move\text{-}\boldsymbol{E}\rangle\phi$ says that, at the current node, player ***E*** has a strategy for responding to ***A***'s initial move which ensures that the property expressed by ϕ results after two steps. [1]

Excluded middle and determinacy Extending this observation to extensive games up to some finite depth k, and using alternations $\Box\Diamond\Box\Diamond\ldots$ of modal operators up to length k, we can express the existence of winning strategies in fixed finite games. Indeed, given this connection, with finite depth, standard logical laws have immediate game-theoretic import. In particular, consider the valid *law of excluded middle* in the following modal form

$$\Box\Diamond\Box\Diamond\ldots\phi \;\vee\; \neg\Box\Diamond\Box\Diamond\ldots\phi$$

or after some logical equivalences, pushing the negation inside:

$$\Box\Diamond\Box\Diamond\ldots\phi \;\vee\; \Diamond\Box\Diamond\Box\ldots\neg\phi,$$

where the dots indicate the depth of the tree. Here is its game-theoretic import:

Fact Modal excluded middle expresses the determinacy of finite games.

Determinacy is the key property that *one of the two players has a winning strategy*. This need not be true in infinite games (players cannot both have one, but maybe neither has).

Zermelo's theorem This brings us to perhaps the oldest game-theoretic result proved in mathematics, even predating Backward Induction, proved by Ernst Zermelo in 1913:

---

[1] One can also express the existence of 'winning strategies', 'losing strategies', and so forth.

Theorem Every finite zero-sum 2-player game is determined.

Proof Here is a simple algorithm determining the player having the winning strategy at any given node of a game tree of this finite sort. It works bottom-up through the game tree. First, colour those end nodes *black* that are wins for player ***A***, and colour the other end nodes *white*, being the wins for ***E***. Then extend this colouring stepwise as follows:

If all children of node *s* have been coloured already, do one of the following:

(a) if player ***A*** is to move, and at least one child is black: colour *s black;* if all children are white, colour *s white*

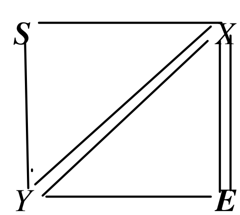
(b) if player ***E*** is to move, and at least one child is white: colour *s white;* if all children are black, colour *s black*

This procedure eventually colours all nodes black where player ***A*** has a winning strategy, making those where ***E*** can win white. The reason for its correctness is easy to see. ■

Zermelo's Theorem is widely applicable. Consider the following Teaching Game:

Example Teaching, the grim realities.

A Student located at position *S* in the next diagram wants to reach the *escape E* below, while the Teacher wants to prevent him from getting there. Each line segment is a path that can be traveled. In each round of the game, the Teacher cuts one connection, anywhere, while the Student can, and must travel one link still open to him at his current position:



General education games like this arise on any graph with single or multiple lines. ■

We now have an explanation why Student or Teacher has a winning strategy: the game is two-player zero sum and of finite depth – though it need not have an effective solution. Zermelo's Theorem implies that in Chess, one player has a winning strategy, or the other a non-losing one, but a century later, we do not know which: the game tree is too large.

5 Fixed-point languages for equilibrium concepts

A good test for logics is their expressive power in representing proofs of significant results. Now our modal language cannot express the *generic character* of the Zermelo solution. Here is what the colouring algorithm really says. Starting from atomic predicates $\boldsymbol{win_i}$ at end nodes indicating which player has won, we inductively defined predicates WIN_i ('player $\boldsymbol{i}$ has a winning strategy at the current node') through the following recursion:

$$WIN_i \leftrightarrow (\boldsymbol{end} \,\&\, \boldsymbol{win_i}) \vee (\boldsymbol{turn_i} \,\&\, \langle E \rangle\, WIN_i) \vee (\boldsymbol{turn_j} \,\&\, [A]\, WIN_i)$$

Here E is the union of all available moves for player $\boldsymbol{i}$, and A that of all moves for the counter-player $\boldsymbol{j}$. This schema is an inductive definition for the predicate WIN_i , which we can also write as a *smallest fixed-point* expression in an extended modal language:

Fact The Zermelo solution is definable as follows in the modal *μ–calculus*:

$$WIN_i \;=\; \mu p \bullet\; (\boldsymbol{end} \,\&\, \boldsymbol{win_i}) \vee (\boldsymbol{turn_i} \,\&\, \langle E \rangle\, p) \vee (\boldsymbol{turn_j} \,\&\, [A] p)$$ [2]

Here the formula on the right-hand side belongs to the *modal μ–calculus*, an extension of the basic modal language with operators for *smallest* (and greatest) *fixed-points* defining inductive notions. This system was originally invented to increase the power of modal logic as a process theory. We refer to the literature for details, cf. Bradfield & Stirling 2006. Fixed-points fit well with strategic equilibria, and the μ–calculus has further uses in games.

Definition Forcing modalities.
Forcing modalities are interpreted as follows in extensive game models as defined earlier: $\boldsymbol{M}, s \models \{i\}\phi$ iff player i has a strategy for the sub-game starting at s which guarantees that only nodes will be visited where ϕ holds, whatever the other player does. ■

Forcing talk is widespread in games, and it is an obvious target for logical formalization: [3]

Fact The modal μ–calculus can define forcing modalities.

---

[2] Note that the defining schema only has *syntactically positive occurrences* of the predicate p.

[3] Note that $\{i\}\phi$ talks about intermediate nodes, not just the end nodes of a game. The existence of a winning strategy for player i can then be formulated by restricting to endpoints: $\{i\}\, (\boldsymbol{end} \rightarrow win_i)$.

Proof The formula $\{i\}\phi = \mu p\bullet\ (\mathbf{end}\ \&\ \phi) \vee (\mathbf{turn}_i\ \&\ \langle E\rangle p) \vee (\mathbf{turn}_j\ \&\ [A]p)$ defines the existence of a strategy for ***i*** ensuring that proposition ϕ holds, whatever the other plays. ■

But many other notions are definable. For instance, the recursion

$$COOP\ \phi \leftrightarrow \mu p\bullet (\mathbf{end}\ \&\ \phi) \vee (\mathbf{turn}_i\ \&\ \langle E\rangle p) \vee (\mathbf{turn}_j\ \&\ \langle A\rangle p)$$

defines the existence of a *cooperative outcome* ϕ, just by shifting some modalities. [4]

Digression: from smallest to greatest fixed-points The above modal fixed-point definitions reflect the equilibrium character of basic game-theoretic notions (Osborne & Rubinstein 1994), reached through some process of iteration. In this general setting, which includes infinite games, we would switch from smallest to *greatest fixed-points*, as in the formula

$$\{i\}\phi = \nu q\bullet\ (\phi\ \&\ (turn_i\ \&\ \langle move\text{-}i\rangle q) \vee (turn_j\ \&\ [move\text{-}j]q)).$$

This is also more in line with our intuitive view of strategies. The point is not that they are built up from below, but that they can be used as needed, and then remain at our service as pristine as ever the next time – the way we think of doctors. This is the modern perspective of *co-algebra* (Venema 2006). More generally, greatest fixed-points seem the best logical analogue to the standard equilibrium theorems from analysis that are used in game theory.

But why logic? This may be a good place to ask *what is the point* of logical definitions of game-theoretic notions? I feel that logic has the same virtues for games as elsewhere. Formalization of a practice reveals what makes its key notions tick, and we also get a feel for new notions, as the logical language has myriads of possible definitions. Also, the theory of expressive power, completeness, and complexity of our logics can be used for model checking, proof search, and other activities not normally found in game theory.

But there is also another link. Basic notions of logic *themselves* have a game character, such as argumentation, model checking, or model comparison. Thus, logic does not just *describe games*, it also *embodies games*. Pursuing the interface in this dual manner, the true grip of the logic & games connection becomes clear: cf. van Benthem, to appear B.

---

[4] This fixed point can still be defined in propositional dynamic logic, using the formula $\langle ((turn_i)\mathbf{?}\ \mathbf{;}\ E) \cup (turn_j)\mathbf{?}\ \mathbf{;}\ A))^*]\ (\mathbf{end}\ \&\ \phi)$, – but we will only use the latter system later in the game setting.

6 Dynamic logics of strategies

Strategies, rather than single moves, are protagonists in games, Moving them in focus requires an extension of modal logic to *propositional dynamic logic* (*PDL*) which describes structure and effects of imperative programs with operations of (a) sequential composition *;*, (b) guarded choice *IF ... THEN... ELSE...*, and (c) guarded iterations *WHILE... DO...*:

Definition Propositional dynamic logic.

The *language of PDL* defines formulas and programs in a mutual recursion, with formulas denoting sets of worlds ('local conditions' on 'states' of the process), while programs denote binary transition relations between worlds, recording pairs of input and output states for their successful terminating computations. Programs are created from

> atomic actions ('moves') $a, b, \ldots$ and tests $?\varphi$ for arbitrary formulas φ,
> using the three operations of $;$ (interpreted as sequential composition),
> $\cup$ (non-deterministic choice) and * (non-deterministic finite iteration).

Formulas are as in our basic modal language, but with modalities $[\pi]\varphi$ saying that φ is true after every successful execution of the program π starting at the current world. ■

The logic *PDL* is decidable, and it has a transparent complete set of axioms for validity. This formalism can say a lot more about our preceding games. For instance, the *move* relation in our discussion of our first extensive game was really a union of atomic transition relations, and the pattern that we discussed for the winning strategy was as follows:

$$[a \cup b] \langle c \cup d \rangle p.$$

Strategies as transition relations Game-theoretic strategies are *partial transition functions* defined on players' turns, given via a bunch of conditional instructions of the form "if she plays this, then I play that". More generally, strategies may be viewed as binary transition relations, allowing for non-determinism, i.e., more than one 'best move', like *plans* that agents have in interactive settings. A plan can be useful, even when it merely constrains my future moves. Thus, on top of the 'hard-wired' moves in a game, we get defined relations for players' strategies, and these definitions can often be given explicitly in a *PDL*-format.

In particular, in finite games, we can define an explicit version of the earlier forcing modality, indicating the strategy involved – without recourse to the modal μ–calculus:

Fact For any game program expression σ, *PDL* can define an explicit forcing modality $\{\sigma, i\}\phi$ stating that *σ is a strategy for player i forcing the game, against any play of the others, to pass only through states satisfying ϕ.*

The precise definition is an easy exercise (cf. van Benthem 2002). Also, given strategies for both players, we should get to a unique history of a game, and here is how:

Fact Outcomes of running joint strategies σ, τ can be defined in *PDL.*

Proof The formula $[((?turn_E ; \sigma) \cup (?turn_A ; \tau))^*] (\mathbf{end} \rightarrow p)$ does the job. [5] ■

Also 'locally', *PDL* can define specific strategies. Take any finite game $\boldsymbol{M}$ with strategy σ for player i. As a relation, σ is a finite set of ordered pairs (s, t). Thus, it can be defined by a program union, if we first define these ordered pairs. To do so, assume we have an 'expressive' model $\boldsymbol{M}$, where states s are definable in our modal language by formulas def_s. [6] Then we define transitions (s, t) by formulas $def_s; a; def_t$, with a being the relevant move:

Fact In expressive finite extensive games, all strategies are *PDL*-definable.

Dynamic logic can also define strategies running over only part of a game, and their *combination*. The following modal operator describes the effect of such a partial strategy σ for player $\boldsymbol{E}$ running until the first game states where it is no longer defined:

$$\{\sigma, \boldsymbol{E}\}\phi \qquad [(?turn_E ; \sigma) \cup (?turn_A ; \textbf{move-}A)^*]\ \phi$$ [7]

[5] Dropping the antecedent '***end*** $\rightarrow$' here will describe effects of strategies at intermediate nodes.

[6] This expressive power can be achieved in several ways: e.g., using temporal *past modalities* involving converse moves which can describe the total history leading up to s.

[7] *Stronger modal logics of strategies?* The modal μ-calculus is a natural extension of *PDL*, but it lacks explicit programs or strategies, as its formulas merely define properties of states. Is there a version of the μ–calculus that extends *PDL* in defining more transition relations? Say, a simple strategy 'keep playing a' guarantees infinite a-branches for greatest fixed-point formulas like $\nu p\bullet \langle a\rangle p$. Van Benthem & Ikegami 2008 look at richer fragments than *PDL* with explicit programs as solutions to fixed-point equations of special forms, guaranteeing uniform convergence by stage ω.

7 Preference logic and defining backward induction

Real games go beyond game forms by adding preferences for players over outcome states, or numerical utilities beyond ‘win’ and ‘lose’. In this area, defining the Backward Induction procedure for solving extensive games, rather than computing binary Zermelo winning positions, has become a benchmark for game logics – and many solutions exist:

Fact The Backward Induction path is definable in modal preference logic.

Solutions have been published by many logicians and game-theorists in recent years, cf. de Bruin 2004, van der Hoek & Pauly 2006. We do not state an explicit *PDL*-style solution here, but we give one version involving a *modal preference language* with this operator:

$<pref_i>\phi$: player i prefers some node where ϕ holds to the current one.

The following result from van Benthem, van Otterloo & Roy 2006 defines the backward induction path as a unique relation σ: not by means of any specific modal formula in game models $\boldsymbol{M}$, but rather via the following *frame correspondence* on finite structures:

Fact The *BI* strategy is definable as the unique relation σ satisfying the following axiom for all propositions P – viewed as sets of nodes –, for all players i:
$(turn_i \,\&\, <\sigma^*>(\boldsymbol{end} \,\&\, P)) \rightarrow [move\text{-}i]<\sigma^*>(\boldsymbol{end} \,\&\, <pref_i>P)$.

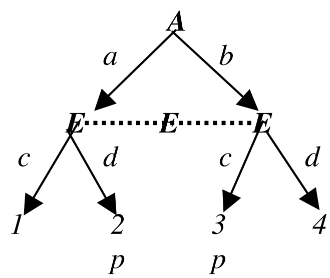
Proof The axiom expresses a form of rationality: at the current node, no alternative move for a player guarantees outcomes that are all strictly better than those ensuing from playing the current backward induction move. The proof is by induction on the game tree. ■

8 Epistemic logic of games with imperfect information

The next level of static game structure gives up the presupposition of perfect information. Consider extensive games with *imperfect information*, whose players need not know where they are in a tree. This happens in card games, electronic communication, through bounds on memory or observation. Such games have ‘information sets’: equivalence classes of relations $\sim_i$ between nodes which players i cannot distinguish. Van Benthem 2001 shows how these games model an epistemic modal language including knowledge operators $K_i\phi$ interpreted in the usual manner as "ϕ is true at all nodes $\sim_i$–related to the current one".

Example Partial observation in games.

In this imperfect information game, the dotted line indicates player ***E***'s uncertainty about her position when her turn comes. Thus, she does not know the move played by player ***A***: [8]



■

Structures like this are game models of the earlier kind with added epistemic *uncertainty relations* $\sim_I$ for each player. Thus, they interpret a combined *dynamic-epistemic* language. For instance, after ***A*** plays move *c* in the root, in both middle states, ***E*** knows that playing *a* or *b* will give her *p* – as the disjunction $<a>p \lor <b>p$ is true at both middle states:

$$K_E(<a>p \lor <b>p)$$

On the other hand, there is no *specific* move of which ***E*** knows at this stage that it will guarantee a *p*–outcome – and this shows in the truth of the formula

$$\neg K_E<a>p \ \ \& \ \ \neg K_E<b>p$$

Thus, ***E*** knows *de dicto* that she has a strategy which guarantees *p*, but she does not know, *de re*, of any specific strategy that it guarantees *p*. Such finer distinctions are typical for a language with both actions and knowledge for agents. [9]

We can analyze imperfect information games studying properties of players by modal frame correspondences. An example is the analysis of Perfect Recall for a player *i*:

Fact The axiom $K_i[a]\phi \to [a]K_i\phi$ holds for player *i* w.r.t. any proposition ϕ iff ***M*** satisfies *Confluence*: $\forall xyz: ((\ xR_ay \ \&\ y\sim_i z) \to \exists u: ((\ x\sim_i u \ \ \&\ uR_az).$

[8] Maybe ***A*** put his move in an envelope, or ***E*** was otherwise prevented from observing.

[9] You may know that the ideal partner for you is around on the streets, but tragically, you might never convert this $K\exists$ combination into $\exists K$ knowledge that some particular person is right for you.

Similar frame analyses work for memory bounds, and observational powers. For instance, agents satisfy 'No Miracles' when epistemic uncertainty relations can only disappear by observing subsequent events they can distinguish. The preceding game has Perfect Recall, but it violates No Miracles: ***E*** suddenly knows where she is after she played her move.

Uniform strategies Another striking aspect of our game is *non-determinacy*. ***E***'s playing 'the opposite direction from that of player ***A***' was a strategy guaranteeing outcome p in the matching game with perfect information – but it is unusable now. For, ***E*** cannot tell if the condition holds! Game theorists only accept *uniform strategies* here, prescribing the same move at indistinguishable nodes. But then no player has a winning strategy, with p as '***E*** wins' (and $\neg p$ as a win for player ***A***). ***A*** did not have one to begin with, ***E*** loses hers. [10]

As for explicit strategies, we can again use *PDL*-style programs, but with a twist. We need the '*knowledge programs*' of Fagin et al. 1995, whose only test conditions are knowledge statements. In such programs, actions can only be guarded by conditions that the agent knows to be true or false. It is easy to see that knowledge programs can only define uniform strategies. A converse also holds, modulo some assumptions on expressiveness of the game language defining nodes in the game tree (van Benthem 2001):

Fact On expressive finite games of imperfect information, the uniform strategies are precisely those definable by knowledge programs in epistemic *PDL*.

9 From statics to dynamics: *DEL*-representable games

Now we make a switch. Our approach so far was *static*, using modal-preferential-epistemic logics to describe properties of fixed games. But it also makes sense to look at *dynamic* scenarios, where games can change. As an intermediate step, we analyze how a static game model might have come about by some dynamic process – the way we see a dormant volcano but can also imagine the tectonic forces that shaped it originally. We provide two illustrations, linking games of imperfect information first to dynamic-epistemic logic *DEL*, and then to epistemic-temporal logics (*ETL*, Parikh & Ramanujam 2003; cf. van Benthem,

[10] The game does have probabilistic solutions in random strategies: it is like Matching Pennies.

Gerbrandy, Hoshi & Pacuit 2007 on connections). Our sketch will make most sense to readers already familiar with these logics of short-term and long-term epistemic dynamics.

Imperfect information games and dynamic-epistemic logic Dynamic-epistemic logic describes how uncertainty is created systematically as initial uncertainty in an agent model ***M*** combines with effects of partially observed events ***E*** to create *product models* ***M x E***. Which imperfect information games 'make sense' with concrete sequences of update steps – as opposed to being just arbitrary placements of uncertainty links over game forms?

Theorem An extensive game is isomorphic to a repeated product update model *Tree(**M**, **E**)* for some sequence of epistemic event models ***E*** iff it satisfies, for all players:

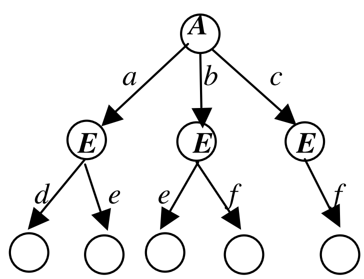
(a) Perfect Recall, (b) No Miracles, and (c) Bisimulation Invariance

for the domains of all the move relations. [11]

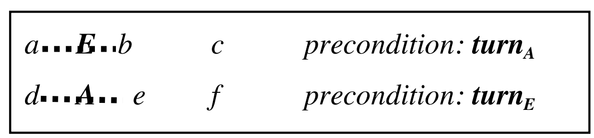
Here Perfect Recall is essentially the earlier commutation between moves and uncertainty. We do not prove the Theorem here: cf. van Benthem & Liu 2004. Here is an illustration:

Example Updates during play: propagating ignorance along a game tree.

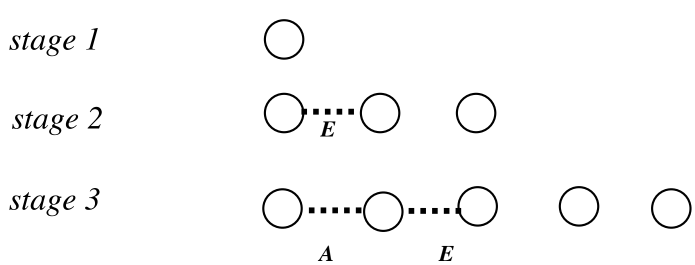
Game tree



Event model



Here are the successive updates that create the right uncertainty links:



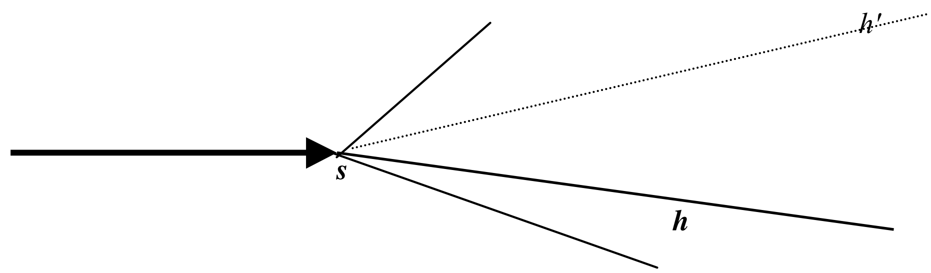
■

[11] I.e., two epistemically bisimilar nodes in the game tree make the same moves executable.

10 Future uncertainty, procedural information, and branching temporal logic

A second logical perspective on games notes that 'imperfect information' has two senses. One is *observation uncertainty*: players may not have seen all events so far, and so they do not know where they are in the game. This is the 'past-oriented' view of *DEL*. But there is also 'future-oriented' *expectation uncertainty*: even in perfect information games players who know where they are may not know what others, or they themselves, are going to do. The positive side is this. In general, players have some *procedural information* about what is going to happen. Whether viewed negatively or positively, the latter future-oriented kind of knowledge and ignorance need not be reducible to the earlier uncertainty between local nodes. Instead, it naturally suggests current uncertainty between whole future histories, or between players' strategies (i.e., whole ways in which the game might evolve).

Branching epistemic temporal models The following structure is common to many fields. In tree models for branching time, 'legal histories' h represent possible evolutions of some process. At each stage of the game, players are in a node s on some actual history whose past they know, either completely or partially, but whose future is yet to be fully revealed:



This can be described in an action language with knowledge, belief, and added temporal operators. We first describe games of perfect information (about the past, that is):

(a) $\boldsymbol{M}, h, s \models F_a\phi$ iff $s^\frown\langle a\rangle$ lies on h and $\boldsymbol{M}, h, s^\frown\langle a\rangle \models \phi$

(b) $\boldsymbol{M}, h, s \models P_a\phi$ iff $s = s'^\frown\langle a\rangle$, and $\boldsymbol{M}, h, s' \models \phi$

(c) $\boldsymbol{M}, h, s \models \Diamond_i\, \phi$ iff $\boldsymbol{M}, h', s \models \phi$ for some h' equal for i to h up to stage s.

Now, as moves are played publicly, players make public observations of them:

Fact The following valid principle is the *ETL* equivalent of the key *DEL* recursion axiom for public announcement: $F_a\Diamond\phi \leftrightarrow (F_aT \,\&\, \Diamond F_a\phi)$.

Trading future for current uncertainty Again, there is a 'dynamic reconstruction' closer to local *DEL* dynamics. Intuitively, each move by a player is a public announcement that changes the current game model. Here is a folklore observation (van Benthem 2004, 2008) converting 'global' future uncertainty into 'local' present uncertainty:

Fact Trees with future uncertainty are isomorphic to trees with current uncertainties.

11 Intermezzo: three levels of logical game analysis

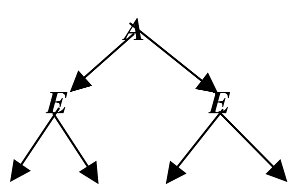
At this point, it may be useful to distinguish three natural levels at which games have given rise to models for logics. All three come with their own intuitions, both static and dynamic.

Level One takes *extensive game trees* themselves as models for modal logics, with nodes as worlds, and accessibility relations over these for actions, preferences, and uncertainty. *Level Two* looks at extensive games as *branching tree models*, with nodes and complete histories, supporting richer epistemic-temporal (-preferential) languages. The difference with Level One seems slight in finite games, where histories may be marked by end-points. But the intuitive step seems clear, and also, Level Two does not reduce in this manner when game trees are infinite. But even this is not enough for some purposes!

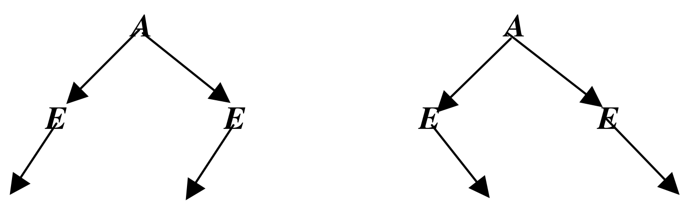
Consider 'higher' hypotheses about the future, involving procedural information about other players' strategies. I may know that I am playing against either a 'simple automaton', or a 'sophisticated learner'. Modeling this may go beyond epistemic-temporal models:

Example Strategic uncertainty.

In the following simple game, let ***A*** know that ***E*** will play the same move throughout:



Then all four histories are still possible. But ***A*** only considers two *future trees* possible, viz.



In longer games, this difference in modeling can be highly important, because observing only one move by ***E*** will tell ***A*** exactly what ***E***'s strategy will be in the whole game. ■

To model these richer settings, one needs full-fledged *Level Three* epistemic *game models*.

Definition Epistemic game models.

Epistemic game models for an extensive game ***G*** are epistemic models $\boldsymbol{M} = (W, \sim_i, V)$ whose worlds are abstract indices including local (factual) information about all nodes in ***G***, plus whole strategy profiles for players, i.e., total specifications of everyone's behaviour throughout the game. Players' global information about game structure and procedure is encoded by uncertainty relations $\sim_i$ between the worlds of the game model. ■

The above uncertainty between two strategies of my opponent would be naturally encoded in constraints on the set of strategy profiles represented in such a model. And observing some moves of yours in the game telling me which strategy you are actually following then corresponds to dynamic update of the initial model, in the sense of our earlier chapters.

Level-Three models are a natural limit for games and other scenarios of interactive agency. Our policy is always to discuss issues at the simplest model level where they make sense.

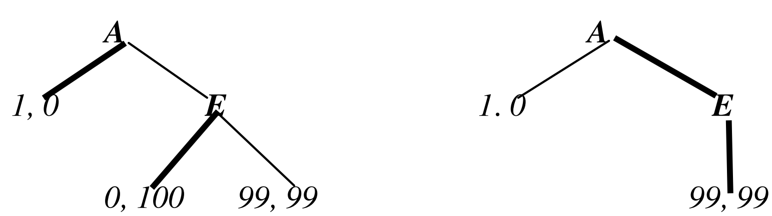
12 Game change: public announcements, promises and solving games

Now let us look at actual transformations that *change games*, and triggers for them.

Promises and intentions Following van Benthem 2007D, one can break the impasse of a bad Backward Induction solution by changing the game through making *promises*.

Example Promises and game change.

In this earlier game, the 'bad Nash equilibrium' *(1, 0)* can be avoided by ***E***'s *promise* that she will not go left, by public announcement that some histories will not occur (we may make this binding, e.g., by attaching a huge fine to infractions) – and the new equilibrium *(99, 99)* results, making both players better off by restricting the freedom of one of them!



■

But one can also add moves to a game, [12] or give information about players' preferences.

Theorem The modal logic of games plus public announcement is completely axiomatized by the modal game logic chosen, the recursion axioms of *PAL* for atoms and Booleans, plus the following law for the move modality: $\langle !P\rangle\langle a\rangle\varphi \leftrightarrow (P \wedge \langle a\rangle(P \wedge \langle !P\rangle\varphi))$.

Using *PDL* again for strategies, this leads to a logic *PDL+PAL* with public announcements $[!P]$. It is easy to show that *PDL* is closed under relativization to definable sub-models, both in its propositional and its program parts, and this underlies the following result:

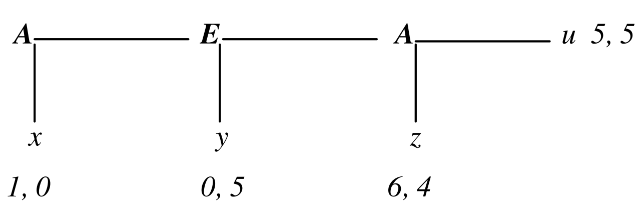
Theorem *PDL+ PAL* is axiomatized by merging their separate laws, while adding the following reduction axiom: $[!P]\{\sigma\}\,\phi \leftrightarrow (P \rightarrow \{\sigma|P\}[!P]\phi)$.

But of course, we also want to know about versions with epistemic preference languages – and hence there are many further questions following up on these initial observations.

Solving games by announcements of rationality Another type of public announcement in games iterates various assertions expressing that players are rational, as a sort of 'public reminders'. Van Benthem 2007C has the following result for extensive games:

Theorem The Backward Induction solution for extensive games is obtained through repeated announcement of the assertion "no player chooses a move all of whose further histories end worse than all histories after some other available move".

Proof This can be proved by a simple induction on finite game trees. The principle will be clear by seeing how the announcement procedure works for a 'Centipede game', with three turns as indicated, branches indicated by name, and pay-offs given for ***A***, ***E*** in that order:



[12] Yes, in this way, one could code up all such game changes beforehand in one grand initial 'Super Game' – but that would lose all the flavour of understanding what happens in a stepwise manner.

Stage 0 of the announcement procedure rules out branch u,

Stage 1 then rules out z, while Stage 2 finally rules out y. ■

This iterated announcement procedure for extensive games ends in largest sub-models in which players have common belief of rationality, or other doxastic-epistemic properties.

Alternatives Of course, a logical language provides many other assertions to be announced, such as history-oriented alternatives, where players steer future actions by reminding themselves of *legitimate rights of other players*, because of 'past favours received'.

The same ideas work in strategic games, using assertions of *Weak Rationality* ("no player chooses a move which she knows to be worse than some other available one") and *Strong Rationality* ("each player chooses a move she thinks may be the best possible one"):

Theorem The result of iterated announcement of *WR* is the usual solution concept of Iterated Removal of Strictly Dominated Strategies; and it is definable inside $\boldsymbol{M}$ by means of a formula of a modal μ–calculus with inflationary fixed-points. The same for iterated announcement of *SR* and game-theoretic Rationalizability. [13]

13 Belief, update and revision in extensive games

So far, we studied players' knowledge. We merely indicate how one can also study their equally important beliefs. For a start, one can use Level-One game models with relations of relative plausibility between nodes inside epistemic equivalence classes. Players' beliefs then hold in the most plausible epistemically accessible worlds, and conditional beliefs can be defined as an obvious generalization. But perhaps more vivid is a Level-Two view of branching trees with belief structure. Recall the earlier *ETL* models, and add binary relations $\leq_{I,\, s}$ of state-dependent *relative plausibility* between histories:

Definition Absolute and conditional belief.

We set $\boldsymbol{M}, h, s \models \langle B, i\rangle\phi$ iff $\boldsymbol{M}, h', s \models \phi$ for some history h' coinciding with h up to stage s and most plausible for i according to the given relation $\leq_{I,\, s}$. As an extension, $\boldsymbol{M}, h, s$

[13] If the iterated assertion A has 'existential-positive' syntactic form (for instance, *SR* does), the relevant definition can be formulated in a standard epistemic μ–calculus.

$\models \langle B, i\rangle^{\psi}\phi$ iff $\boldsymbol{M}, h', s \models \phi$ for some history h' most plausible for i according to the given $\leq_{I, s}$ among all histories coinciding with h up to stage s and satisfying $\boldsymbol{M}, h', s \models \psi$. ■

Now, belief revision happens as follows. Suppose we are at node s in the game, and move a is played which is publicly observed. At the earlier-mentioned purely epistemic level, this event just eliminates some histories from the current set. But there is now also belief revision, as we move to a new plausibility relation $\leq_{I, s^\wedge a}$ describing the updated beliefs.

Hard belief update First, assume that plausibility relations are not node-dependent, making them global. In that case, we have belief revision under *hard information*, eliminating histories. The new plausibility relation is the old one, restricted to a smaller set of histories. Here is the characteristic recursion law that governs this process. A temporal operator $F_a\varphi$ says a is the next event on the current branch, and that φ is true immediately after:

Fact The following temporal principles are valid for hard revision along a tree:

$$F_a\langle B, i\rangle\phi \leftrightarrow (F_aT \;\&\; \langle B, i\rangle(F_aT, F_a\phi))$$
$$F_a\langle B, i\rangle^{\psi}\phi \leftrightarrow (F_aT \;\&\; \langle B, i\rangle(F_a\psi, F_a\phi))$$ [14]

Soft update But belief dynamics is often driven by events of *soft information*, which do not eliminate worlds, but merely rearrange their plausibility ordering (van Benthem 2007A), as happens in the familiar model-theoretic 'Grove sphere semantics' of belief revision theory. In the above, we already cast Backward Induction in this manner, as a way of creating plausibility relations in a game tree – but beyond such an 'off-line' preprocessing phase of a given game, there can also be dynamic 'on-line' events that might change players' beliefs and expectations in the course of an actual play of the game. With doxastic-temporal models adapted to this setting, we get representation theorems (van Benthem & Dégrémont 2008) that say which doxastic-temporal models are produced by plausibility update in the style of Baltag & Smets 2006. Also, Baltag, Smets & Zvesper 2008 provide a striking new dynamic alternative to Aumann-style characterization theorems for Backward Induction.

Further entanglements: dynamics of rationalization In all our scenarios and logics, knowledge and belief have been entangled notions – and this entanglement even extends to

---

[14] Similar 'coherence' laws occur in Bonanno 2007, which formalizes games using *AGM* theory.

players' preferences (Girard 2008, Liu 2008). But there are many other dynamic scenarios. For instance, van Benthem 2007D discusses *rationalization* of observed behaviour in games, adapting preferences, beliefs, or both, to make observed behaviour rational.

14 Conclusion

We have shown how games naturally involve static and dynamic logics of action, knowledge, belief, and preference. We gave pilot studies rather than grand theory, and we found more open problems than final results. It would be easy to pile up further topics (cf. van Benthem, to appear A), pursuing issues of procedural knowledge, soft update and genuine belief revision in games, agent diversity and bounded rationality, infinite games, or connections to explicit automata-theoretic models of agents (as urged by Ram Ramanujam in his 2008 invited lecture at the Workshop on 'Logics of Intelligent Interaction', ESSLLI Hamburg). True. But even at the current level of detail, we hope to have shown that logic and games is an exciting area for research with both formal structure and intuitive appeal.

15 References

A. Baltag, H. van Ditmarsch & L. Moss, 2008, 'Epistemic Logic and Information Update', to appear in P. Adriaans & J. van Benthem, eds., *Handbook of the Philosophy of Information*, Elsevier Science Publishers, Amsterdam.

A. Baltag & S. Smets, 2006, 'Dynamic Belief Revision over Multi-Agent Plausibility Models', in G. Bonanno, W. van der Hoek, M. Wooldridge , eds., *Proceedings of the 7th Conference on Logic and the Foundations of Game and Decision (LOFT 2006),* p. 11-24, University of Liverpool,

A. Baltag, S. Smets & J. Zvesper, 2008, 'When All is Said and (Not Yet) Done; Dynamic Rationality in Games', *Proceedings LOFT 2008*, University of Amsterdam.

J. van Benthem, 2001, 'Games in Dynamic Epistemic Logic', *Bulletin of Economic Research* 53:4, 219 – 248.

J. van Benthem, 2002, 'Extensive Games as Process Models', *Journal of Logic, Language and Information* 11, 289–313.

J. van Benthem, 2004, 'Update and Revision in Games', lecture notes, ILLC Amsterdam & Philosophy Stanford.

J. van Benthem, 2007A, ‘Dynamic Logic of Belief Revision’, *Journal of Applied Non-Classical Logics* 17:2, 129 – 155.

J. van Benthem, 2007B, ‘Logic Games, From Tools to Models of Interaction’, in A. Gupta, R. Parikh & J. van Benthem, eds., *Logic at the Crossroads*, Allied Publishers, Mumbai, 283 – 317.

J. van Benthem, 2007C, ‘Rational Dynamics’, *International Game Theory Review* 9:1, 2007, 13 – 45. Erratum reprint, Volume 9:2, 377 – 409.

J. van Benthem 2007D, ‘Rationalizations and Promises in Games’, *Philosophical Trends*, ‘Supplement 2006’ on logic, Chinese Academy of Social Sciences, Beijing, 1–6.

J. van Benthem, 2008, ‘The Information in Intuitionistic Logic’, to appear in *Synthese*.

J. van Benthem, to appear A, *Logical Dynamics of Information and Interaction*, Book manuscript, Institute for Logic, Language and Computation, University of Amsterdam.

J. van Benthem, to appear B, *Logic in Games*, Book manuscript, Institute for Logic, Language and Computation, University of Amsterdam.

J. van Benthem & C. Dégrémont, 2008, ‘Multi-Agent Belief Dynamics: building bridges between dynamic doxastic and doxastic temporal logics’, in G. Bonanno, W. van der Hoek & B. Loewe, eds., *Proceedings LOFT VIII,* University of Amsterdam.

J. van Benthem, J. Gerbrandy, T. Hoshi & E. Pacuit, 2007, ‘Merging Frameworks for Interaction’, *Proceedings TARK 2007*, University of Namur. To appear in the *Journal of Philosophical Logic*.

J. van Benthem & D. Ikegami, 2008, ‘Modal Fixed-Point Logic and Changing Models’, in A. Avron, N. Dershowitz & A. Rabinovich, eds., *Pillars of Computer Science: Essays Dedicated to Boris (Boaz) Trakhtenbrot on the Occasion of his 85th Birthday*, Springer, Berlin.

J. van Benthem & F. Liu, 2004, ‘Diversity of Logical Agents in Games’, *Philosophia Scientiae 8:2,* 163 – 178.

J. van Benthem, van Otterloo & Roy, 2006, ‘Preference Logic, Conditionals, and Solution Concepts in Games’, in H. Lagerlund, S. Lindström & R. Sliwinski, eds., *Modality Matters*, University of Uppsala, 61 – 76.

G. Bonanno, 2007, 'Temporal Belief Revision Rationalized by Plausibility Orderings', Department of Economics, UC Davis. To appear in K. Apt & R. van Rooij, eds., *Games and Interaction*, Texts in Logic and Games, Amsterdam University Press.

J. Bradfield & C. Stirling, 2006, 'Modal μ–Calculi', in P. Blackburn, J. van Benthem & F. Wolter, eds., *Handbook of Modal Logic*, Elsevier, Amsterdam,721–756.

B. de Bruin, 2004, *Explaining Games: on the Logic of Game Theoretic Explanations*, Dissertation, ILLC, University of Amsterdam.

H. van Ditmarsch, W. van der Hoek & B. Kooi, 2007, *Dynamic-Epistemic Logic,* Cambridge University Press, Cambridge.

R. Fagin, J. Halpern, Y. Moses & M. Vardi, 1995, *Reasoning about Knowledge*, The MIT Press, Cambridge (Mass.).

P. Girard, 2008, *Modal Logic for Belief and Preference Change*, Dissertation, Department of Philosophy, Stanford University & ILLC Amsterdam.

W. van der Hoek & M. Pauly, 2006, 'Modal Logic for Games and Information', in P. Blackburn, J. van Benthem & F. Wolter, eds., *Handbook of Modal Logic*, Elsevier, Amsterdam, 1077–1148.

F. Liu, 2008, *Changing for the Better: Preference Dynamics and Preference Diversity*, Dissertation DS-2008-02, ILLC, University of Amsterdam.

M. Osborne & A. Rubinstein, 1994, *A Course in Game Theory*, The MIT Press, Cambridge (Mass.).

R. Parikh & R. Ramanujam, 2003, 'A Knowledge-Based Semantics of Messages', *Journal of Logic, Language and Information* 12, 453 – 467.

Y. Venema, 2006, 'Algebras and Co-Algebras', in P. Blackburn, J. van Benthem & F. Wolter, eds., *Handbook of Modal Logic*, Elsevier, Amsterdam, 331–426.