OPEN PROBLEMS IN LOGIC AND GAMES

Johan van Benthem, Amsterdam & Stanford, February 2005

FIRST ROUGH DRAFT ***formal definitions to be added!***

This short paper is a description of some research lines on logic and games, which occur in logic itself, computer science, and game theory. What I personally find interesting about these recent interfaces is the importance of interaction between several agents as a fundamental theme in logic itself, and the new ways in which mathematical logics of computation and philosophical logics of epistemic attitudes come together. The story is just my attempt at systematizing issues and problems. There is no pretense at completeness. Comments – and solutions – are welcome!

A warning beforehand. This paper is more like a tourist guide for Places To Visit, than for specific Things To Do. This reflects the tentative state of the area, which is less centered around one family of formal systems than e.g., dynamic epistemic logic.

1 Logic and Games

Logic and games meet in several different ways.

Logic games First, argumentation itself is a sort of game where opposing players can win or lose. And thus, in addition to the more dominant semantic or deductive intuitions, logical validity also has a game-like aspect of *winning strategies* for players defending valid conclusions from given data. In addition to argumentation or dialogue games, modern logicians also use a host of other scenarios, usually two-player games of perfect information, for tasks of semantic evaluation in given models, model construction, comparison of two models, proof search, or even general interaction. Some well-known names in these developments are Lorenzen, Ehrenfeucht-Fraïssé, Hintikka – but one can also mention more recent authors like Hodges, Abramsky, Girard, or Hirsch & Hodkinson. For references to this literature, cf. my lecture notes *Logic in Games* (van Benthem 1999–..., still under construction), whose main line of exposition for the general 'Logic & Games' interface has been followed here.

Game logics But general games of any sort have an obvious logical structure as process graphs that can be described in some logical language. This invites the use of logical machinery, in addition to the standard mathematics of game theory. One stream here consist of process languages like modal or dynamic logic, fixed-point languages, temporal logic, or linear logic. This links up with research in logics of computation, and in principle, it provides all the benefits achieved there for games as well: such as better understanding of algorithms, and perhaps even better design.

But there is also another stream. From the outset, the predictions of game theory about equilibria that 'rational' players will or must choose have been a matter of intensive debate. Here, logic has entered as an analysis of the knowledge and beliefs of players underpinning their choices, and the deliberations that go into them. Thus, epistemic logic, conditional logic, and other high-lights of philosophical logic have entered the scene (partially discovered independently by game theorists), promising conceptual clarification of the issues involved, as well as a more systematic view of options for 'rational' agents and rational procedures.

'Game logics' are logical systems designed for the purpose of analyzing games. Modern game logics often combine the preceding two aspects, so that one could – and does – have 'epistemic dynamic logics' for analyzing the strategies that a player might consider or choose in a given game. Other current topics in this area concern more generic structure of games in general, such as the analysis of general game-forming operations. Such issues often cross over into the special area of logic games – making the above distinction between logic games and game logics one of convenience, rather than of principle. E.g., van Benthem 200**x** shows that predicate-logical evaluation games are complete for the algebra of sequential operations on general games. Finally, as in other newly developing areas, there is a tendency to design *new logics* and coin new terminology, rather than going through the more boring expedient of using existing ones, such as standard first-order and modal logic. Who wants to use old tools when the World looks fresh and new? We will also be guilty of this, though we also mention some more conservative approaches.

General activities and information Material on the Logic/Games interface may be found at several places in the literature, though there is no standard source, let alone a textbook. But cf. van Benthem 1999–..., Hodges 200**x**, van der Hoek & Pauly 2004 for some broader perspectives. The web page http://www.illc.uva.nl/~lgc is a public resource under construction, pointing at some relevant papers and journals. Also, there are some conferences and workshops which serve as a forum for work in this area, such as *TARK* (www.tark.org), *LOFT* (url), *LAMACS* (url),and *GLC* (url).

Here and elsewhere, for precise definitions of basic notions concerning games, we refer to the literature. A good compact reference is Osborne & Rubinstein 1994, while ***XYZ*** is an up-to-date one on modern evolutionary game theory.

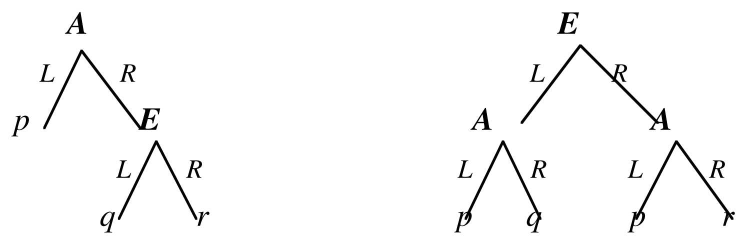
2 Extensive Games as Processes

Extensive games of perfect information are trees whose nodes represent stages of the game, while leaves represent possible final outcomes, which players can evaluate, and compare via their individual preferences. Players turns are indicated at all non-final nodes, and arrows pointing to daughter nodes represent their possible moves there. Game trees can be finite or infinite. In the latter case, infinite branches may sometimes be a nuisance, such as a computer getting stuck forever in a loop, or a person's eternal inability to come to the point. But infinite branches can also be viewed as unbounded histories of ongoing interaction, as with unlimited computational facilities like the internet, or indeed, the functioning of social life. Whether finite or infinite, game structures are much like those used in computer science, or standard mathematical logic, for representing processes via graphs, trees, or other mathematical notions. Hence, immediate analogies spring to mind with well-known logics for describing computational processes, as well as general action. We list a few topics here.

Caveat Note that my questions at this 'conservative' level are not spectacular new mathematical ones, but rather issues of comparing different approaches, and unification across different traditions. In my personal opinion, the most innovative 'game logic' to-day in a standard logical setting is the extensive body of work on games in *temporal logic*, where issues arise by mixing questions about games with those about computational processes (cf. van der Meyden 2005, Ramanujam 2005).

2.1 Game equivalence and bisimulation

Computational logics of processes and actions do not have one fixed level of detail for studying these phenomena. Just as with the many different mathematical theories of Space (affine or metric geometry, topology, linear algebra), there are legitimate choices of structural similarity relations, reflecting what structure of a process one finds of interest (van Benthem 1996 *ELD*). The spectrum runs from output-oriented identifications like *finite trace equivalence*, through finer ones like *modal bisimulation*, which also record internal choice points for agents involved in the process, to the more demanding notion of first-order *isomorphism*. The same spectrum makes sense for games (van Benthem *JoLLI*), running from *equivalence of players' powers* for determining final outcomes, through modal bisimulation, to again stronger notions of isomorphism preserving more game structure. The same 'equivalence levels approach' also works when extra structure is present in games, such as players preferences. Consider the following two games:



Are these the same? The answer depends on our level of interest:

(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. This natural level for looking at games with local moves and choices is that of modal bisimulations. But one might also want to call our games equivalent in another sense – if only, because they are evaluation games for the two sides of the valid logical law

$p \wedge (q \vee r) \leftrightarrow (p \wedge q) \vee (p \wedge r)$ *Distribution*

The sense of equivalence involved then looks at achievable outcomes only:

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

The reason is that players can force the same sets of outcomes across games:

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

Here 'forcing' refers to sets of outcomes guaranteed by strategies for players, their 'powers'. A strategy forces a set *X* if all outcomes of the game, under arbitrary play by the others fall inside *X*. With this understanding, in the left-hand tree, ***A*** has 2 strategies, and so does ***E***, yielding the listed sets. In the right-hand tree, ***E*** has again 2 strategies, while ***A*** has 4: *LL, LR, RL, 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. More generally, at an input–output level, propositional distribution switches the scheduling of a game without affecting players' powers. An appropriate bisimulation for this coarser level of game equivalence has been proposed in many areas independently:

A *power bisimulation* between game models ***M, N*** is a relation
$\boldsymbol{Z}$ between game states in ***M, N*** satisfying the two conditions:

(1) if $x \boldsymbol{Z} y$, then *x, y* satisfy the same proposition letters.

(2a) for each ***i***, if $x \boldsymbol{Z} y$ and ***i*** can force *U* starting from *x*,
then there is a set *V* which ***i*** can force starting from *y*,
such that $\forall v \in V\ \exists u \in U: u \boldsymbol{Z} v$

(2b) vice versa from *y* to *x*.

Thus, game equivalences come in varieties depending on one's level of interest: coarser or finer. But there has been no systematic theory so far of all natural levels.

Open Problem 1 What are natural structural equivalences for games?

In logic, structural similarities usually come with a *language*, describing just those properties that are relevant at the given level. A typical connection result of this sort says that finite models are isomorphic iff they satisfy the same first-order sentences. In process logics, a similar connection result is that there is a bisimulation between two finite rooted models *(**M**, s)* and *(**N**, t)* iff the roots *s, t* satisfy the same formulas in the modal propositional logic describing available moves and atomic properties of nodes. Similarly for games, choosing a description level is correlated with using a particular logical language, modal, first-order, or yet other, to describe properties of nodes in game trees. We start with some obviously available candidates.

2.2 Modal and dynamic logics for moves and strategies

Modal logic Propositional modal logic describes process models

$$\mathbf{M} = (S, (R_a\}_a, V)$$

with modal formulas ϕ stating properties of states $s \in S$, such as

$[a]<b>p$: after every R_a-step from s to any t, there is an R_a–step from t to some state u where p holds.

This $\forall\exists$ pattern of successive modalities is typical for interaction between players: any *a*-move can be countered by some suitably chosen *b*-move leading to an outcome *p*. Thus, modal logic describes possible moves and choices for players in a game tree. The modal similarity type for the latter looks roughly like this:

M = *(NODES, MOVES, PLAYERS,* ***turn****,* ***end****, VAL)*

Dynamic logic A more explicit account of players' plans and strategies requires a richer propositional dynamic logic *PDL* which also has programs π describing binary relations between states, representing the transitions corresponding to successful executions of π. One can think here of computation steps – but by now, *PDL* is used as a very general logic for describing any complex action. These are constructed

from atomic moves *a* and tests $?\phi$ by the three sequential operations of composition **;**, choice $\cup$, and finite iteration *.

This language can describe game trees in more detail than basic modal logic, using e.g., iterations of single moves to describe arbitrary finite paths. But even more importantly, *PDL* can describe the fundamental game-theoretic notion of a *strategy*. For, a player's strategy is nothing but a binary relation giving her a move at each of her turns – where non-deterministic strategies may even allow more than one option.

And natural descriptions of interactive strategies have precisely the sequential and conditional format of *PDL*:

> "*IF* your opponent plays *a*, *THEN* play *b* *ELSE* play *c*",
> "*WHILE* your have not reached some goal, *DO* move *a*".

It is also easy to see that, at least in finite games, *PDL* can easily describe the unique outcome states of games when players *i* play a profile of functional strategies σ_i.

μ*–*Calculus Beyond *PDL*, there are richer fixed-point languages such as the modal *μ*–calculus which can define arbitrary smallest and greatest fixed-point predicates in the modal language by means of recursive definitions. This genuine extension of *PDL* is needed for a faithful rendering of basic game-theoretic algorithms such as Zermelo Colouring when showing that finite two-player zero-sum games are determined. E.g, winning nodes for player *i* in such a game tree are defined by the following recursion:

$$WIN_i \;\leftrightarrow\; (\mathbf{end}\ \&\ \mathbf{win}_i) \vee (\mathbf{turn}_i\ \&\ \langle E\rangle\, WIN_i) \vee (\mathbf{turn}_j\ \&\ [A]\, WIN_i)$$

Thus we can view the predicate WIN_i as the smallest fixed-point defined by

$$\mu p\bullet\ (\mathbf{end}\ \&\ \mathbf{win}_i) \vee (\mathbf{turn}_i\ \&\ \langle E\rangle\, p) \vee (\mathbf{turn}_j\ \&\ [A]p)$$

Van Benthem 2004 claims that game-theoretic equilibrium has to do with fixed-points in a general mathematical sense. But which ones? The *μ*–calculus can also define behaviour of infinite branches, by means of ν–operators for *greatest* fixed-points. This reflects another strong intuition about games, viz. the infinite-stream-like behaviour of strategies. If I am ill, my strategy is to consult my doctor, and extract an advice. After that, my strategy returns - intuitively - to exactly the same state as before. This suggests that game logics will either involve both types of fixed-point, and may even suggest a co-algebraic treatment – as has been proposed in Baltag 200**x**.

I would conclude that existing modal fixed-point languages, or their first-order extensions such as *LFP(FO,* provide excellent means for describing interaction games, as long as we talk about the structure of moves and abstract outcomes – i.e., about what game theorists would call 'game forms'.

Open Problem 2 Do a standard formalization program for key theorems, proofs, and algorithms in game theory, and see which existing logics are necessary.

E.g., De Bruin 200**x** analyzes Backward Induction in a *μ*–calculus setting, with some atomic propositions added for utility values. Van Benthem 200**x** analyzes the proof of the Gale-Stewart Theorem (extending Zermelo's colouring argument to infinite games), identifying its Key Lemma as a law in a temporal logic of players' powers. So, why don't we just use standard logical systems as game logics – and import what we already know about their deductive apparatus and computational complexity for task like model checking or satisfiability? Part of this may just be the New World philosophy that I mentioned before: 'never keep old clothes when you can buy new ones'. Part is also the more general 'modal philosophy' in process logic: try to see what simple special-purpose languages do the job of analyzing classes of games, striking a good balance between expressive power and computational complexity.

2.3 Adding preferences

Now a game form only becomes a genuine game with some real drama when we look at pay-offs and preferences. For instance, consider the earlier two games for Distribution, but now with the following preferences for players:

E	*p: 0*	*q: 2*	*r: 1*
A	*p: 1*	*q: 0*	*r: 2*

Here are the pairs (***A***–value, ***E***–value) computed by the usual *BI* algorithm:

	1, 0		
1, 0		*0, 2*	
	0, 2		*2, 1*

	2, 1		
1, 0		*2, 1*	
1, 0	*0, 2*	*1, 0*	*2, 1*

These trees correspond to unique outcomes for the joint behaviour of the players. Note that these predictions are different!

Open Problem 3 Define good game equivalence when preferences are present.

Further analysis of games with preferences can be done in many ways (van Benthem 1999, JoLLI), but it does need a merge modal dynamic logics with *preference logics*. I do not seee a best formalism yet, but cf. Bonanno 199**x**, van der Hoek, Harrenstein & Meijer 200**x**, van der Hoek, van Otterloo & Wooldridge 200**x**, Pauly 200**x**, and Van Otterloo & Roy 2005 for ongoing attempts. The latter paper has a perspicuous analysis of Backward Induction arguments with minimal logical means, viz. simple reduction axioms relating backward induction subgames to available future moves.

One limitation to all these analyses, however, is the compositional simplicity of Backward Induction, with current best actions built up in terms of those in subtrees lower down. Logical analyses of more complex game-theoretic solution concepts in a direct modal-dynamic-preference setting are scarce. But cf. again De Bruin 2004 on extensive games in a 'proof-theoretic' format which achieves greater generality.

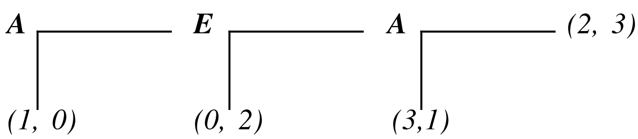
Open Problem 4 Integrate game logics with the older preference logics.

Another take on preferences involves *deontic logics* of obligations and permissions. A deontic statement $O\phi$ says that ϕ is true in all 'best' worlds accessible to the present one, with 'best' as seen from the viewpoint of some moral authority. More generally, conditional obligations $O\phi\psi$ say that ψ is true in all the best worlds satisfying the antecedent condition ϕ. One can let the 'authority' vary here, including a players in games themselves, who try to achieve best outcomes from their own private perspectives. Still deontic logic does not use binary comparison between situations – as in "anything you can do I can do better". But it does suggest other interesting variations. Van der Meyden 199**x** has an account of deontic preferences as located, not between worlds, but between available actions in a process – or we might also say: moves in a game. This suggests that players' preferences might be located elsewhere than their locus at end states in the standard definition of a game.

Open Problem 5 Integrate deontic logic and dynamic-preference logic.

2.4 Rationality assumptions

The main mathematics of standard game theory consists of a definition of games and Nash equilibrium between strategies, some basic existence theorems for strategic equilibrium due initially to Von Neuman and Nash – and a host of refined notions of equilibrium in the decades after the 1950s which try to zero in more closely on natural or useful equilibria. Moreover, there has always been controversy surrounding game-theoretic predictions, or recommendations, as the mathematical model seems too poor to make all relevant considerations explicit. In particular, whether players will play a Backward Induction solution depends on assumptions that underlie their deliberation. This is well-illustrated in the *Centipede Game*.



Backward Induction predicts that ***A*** plays *down*, blocking the better right region!

So, which additional assumptions underwrite the *BI* prediction? A famous example is *Rationality*: the statement that every player will always opt for those available moves that make her off best in future play. Much work on game logics is about formalisms defining Rationality and the reasoning based upon it, locking players into the *BI* solution, or whatever other notion of equilibrium may be bolstered by additional assumptions like this. In particular, Aumann has shown that, if players have common knowledge of rationality in an extensive game, then backward induction must result.

To a first approximation, these arguments require just dynamic-preference logic, as above. And their structure is remarkably similar to that in the philosophy of action, where e.g., the famous Practical Syllogism runs as follows from 'Is's to 'Ought's:

> You *can do a* and *b*.
> You *prefer* the outcome of *a* over that of *b*.
> Therefore, you *will do a*.

Open Problem 6 Connect the game logic of rationality with the philosophy of action and practical reasoning.

2.5 Epistemic and doxastic logic

There is more to rationality analysis for deriving equilibria. One crucial aspect to the whole scenario of deliberation is that players do not know yet what others will do, and what they are going to do themselves. Thus, *knowledge* becomes important, and here is where *epistemic logic* has first entered game theory. This requires an expansion of the notion of a game to that of a *model for a game* (Stalnaker 199**x**), with a space of possible worlds containing different strategy profiles, and players knowledge encoded in the usual manner by relations of epistemic indistinguishability. Since this topic takes us into philosophical logic, we postpone further details until Section 4.

2.6 Game theory and logic: what good does it do?

Game logics look at fine-structure of games below the usual strategic forms, and they describe step-by-step behaviour over time which remains hidden in the usual strategic equilibria. But what good does this do? There have been deep existence results for equilibria in games, and on the logic side, there are deep results on meta-properties of first-order, modal, or yet other calculi. Still, does the combination of good things automatically make sense ? Here are a few things that might be expected:

(a) Systematizing the theory of possible equilibria via their properties as programs
(b) Cleaning up overly complicated notions of game model, such as 'type spaces'.
(c) Finding new mathematical results with a certain depth which integrate Nash-style equilibrium existence theorems with logical meta-theorems.

Very little has happened so far justifying these expectations.

Open Problem 7 Do any of the preceding!

2.7 Infinite games and temporal logic

For infinite games, tree models still suffice – but they suggest other languages in addition to the preceding modal and dynamic ones. In particular, the computational tradition of linear and branching temporal logics seems relevant, as these describe universes of branching time where infinite games can unfold. This is the perspective suggested by Halpern et al. 1995, Parikh & Ramanujam 200**x** (both with epistemic structure added), Alur & Henzinger 199**x**, Goranko 200**x**, and many others. In particular, Halpern 200**x** uses this sort of modeling to take a look at open issues in game theory. Current expressive game logics of this sort include alternating-time temporal logic (*ATL*) and its epistemic version *ATEL* (van der Hoek & Wooldridge 200**x**). Also relevant is the temporal *STIT* formalism of Belnap et al. 200**x**.

Open problem 8 Compare and relate existing modal and temporal game logics.

2.8 Games in set theory

Infinite games have also played a role in descriptive set theory, where the Gale-Stewart Theorem culminated in 'Martin's Theorem' saying that each infinite two-player game of perfect information is determined, i.e., one of the players has a winning strategy., provided the set of winning histories for one of the players lies in the Borel Hierarchy. More foundationally, the Axiom of Determinacy even says that all such games are determined, contradicting the Axiom of Choice and thereby providing an alternative set theory. Loewe 2004 is a nice example of a paper linking up between the notion of game solution in the set-theoretic tradition, and extended notions of Backward Induction on infinite games, provided that players have suitably simple preference ranking over the possible outcomes.

Open Problem 9 Merge between set theory and game theory of infinite games.

2.9 Operations on games

Another tradition in process logics looks at processes as models identified under some equivalence relation, such as bisimulation, and then studies operations defined on equivalence classes that form new processes out of old. Examples are the operations of *process algebra* (Handbook of Process Algebra), which include Choice, Sequential Composition, as well as Parallel Merges of various kinds. There is also a logical system in this tradition, viz. *linear logic*, whose game semantics (Abramsky 199**x**, Japaridze 200**x**) studies operations on infinite game trees such as

choices $G_1 + G_2$, role switch G^d, and parallel compositions $G_1 \bullet G_2$.

Linear logic provides a sort of algebra for dealing with equivalences and implications between complex games – with 'validity' meaning that some designated player P has a winning strategy in all games of the given form.

Open Problem 10 What is a natural repertoire of operations on games?

In particular, there are several parallel compositions for playing a number of games 'together'. One would like to have an expressive completeness result for a natural set of game operations, on the analogy of those for process algebra in Hollenberg 2000.

3 Strategic Forms and Powers

3.1 Powers and game representation

At a coarser level of identification, one looks merely at player's powers. Let's write

$\rho_G^{\boldsymbol{i}}\ s,\ X$ player $\boldsymbol{i}$ has a strategy for playing game G from *state s* onward whose resulting states are always in the set X

It is easy to see that all games satisfy the following two conditions:

C1 if $\rho_{\boldsymbol{G}}{}^{\boldsymbol{i}}\ s,\ Y$ and $Y \subseteq Z$, then $\rho_{\boldsymbol{G}}{}^{\boldsymbol{i}}\ s,\ Z$ *Monotonicity*

C2 if $\rho_{\boldsymbol{G}}{}^{\boldsymbol{A}}\ s,\ Y$ and $\rho_{\boldsymbol{G}}{}^{\boldsymbol{E}}\ s,\ Z,$ then Y, Z overlap *Consistency*

Determined two-player games, such as those of Zermelo's Theorem, also satisfy

C3 if not $\rho_{\boldsymbol{G}}{}^{\boldsymbol{A}}\ s,\ Y$, then $\rho_{\boldsymbol{G}}{}^{\boldsymbol{E}}\ s,\ S-Y;$
and the same holds for $\boldsymbol{E}$ vis-á-vis $\boldsymbol{A}$ *Completeness*

Here is a converse representation theorem:

Fact Any two families F_1, F_2 of subsets of some set S satisfying *C1, C2, C3* are the root powers for the two players in some two-step game.

This result concerns games of perfect information: an analogue for games with imperfect information requires just *C1, C2* (van Benthem 2001). The proofs of these representation results show one peculiarity though: it is crucial to be able to have the same outcome at different leaves of the game tree. If we require that each leave in the game tree is a unique outcome, then further principles beyond *C1, C2, C3* will be valid on finite games, and the restriction to two-step games is no longer appropriate. Here is a more technical question for puzzle solvers:

Open Problem 11 Find a representation theorem like the preceding for players' powers in determined games with unique outcomes.

3.2 Dynamic game logic

One can now also introduce more global modal languages that deal with players' powers directly, in the following format:

$\boldsymbol{M},\ s \models \{G,\ \boldsymbol{i}\}\phi$ iff there is a set X with $\rho_G{}^{\boldsymbol{i}}\ s,\ X$ and $\forall s \in X\ \boldsymbol{M},\ s \models \phi$

These are related to neighbourhood versions of modal logic (cf. Pauly 200**x**). This differ from the earlier cases mainly in that the new modality *{}* with its $\exists\forall$ character does not distribute over either $\wedge$ or $\vee$.

This modal language becomes more expressive when we also add *operations* on games to obtain 'dynamic game logic' *DGL* (Parikh 1985). Models $\boldsymbol{M} = (S, \{R_g\}_g, V)$ then stand for 'game boards' with a universe *S* of states associated with (though not necessarily identical to) game states, forcing relations R_g for given atomic games, and appropriate semantic clauses defining forcing relations for compound games constructed using sequential composition *G;H*, union (choice) $G \cup H$, game dual G^d (role switch), and finite game iteration G^*. Modalities now refer to a game expression:

$\{G, \boldsymbol{i}\}\phi$ $\boldsymbol{i}$ has a strategy making sure game *G* ends only in runs satisfying ϕ

Typical axioms validated by this semantics resemble those of *PDL*, such as

$$\{G \vee H, \boldsymbol{i}\}\phi \leftrightarrow \{G, \boldsymbol{i}\}\phi \vee \{H, \boldsymbol{i}\}\phi$$
$$\{G ; H, \boldsymbol{i}\}\phi \leftrightarrow \{G, \boldsymbol{i}\}\{H, \boldsymbol{i}\}\phi$$

The characteristic axiom for role switch runs as follows in determined games:

$$\{G^d, \boldsymbol{i}\}\phi \leftrightarrow \neg\{G, \boldsymbol{i}\}\neg\phi$$

In non-determined games, these principles need to refer to different players explicitly. Cf. Parikh & Pauly 200**x** for the state of the art with *DGL*. In particular, the system is completely axiomatizable and decidable without the dual operation. Here is one issue, concerning the latter operation, which has no counterpart in *PDL*:

Open Problem 12   Axiomatize *DGL* completely with game dual G^d added.

The modality *{}* is an existential quantifier over strategies, which are not mentioned explicitly. One obvious question is how *DGL* relates to explicit *PDL*-style analysis of games. Van Benthem 1999 suggests a two-pronged approach of game boards in tandem with families of games over them, relating the *DPL*-language over those games with a *DGL* language of the board.

Open Problem 13   Merge *DGL* and *DPL* in some natural way.

DGL's main novelty is in terms of *game algebra*. Dynamic game logic encodes the notion $G \approx H$: players have same powers in every concrete interpretation of *G, H* on a game board. Valid principles of game algebra include:

(a)   *De Morgan Algebra* for choice and dual:
Boolean Algebra minus special laws for ***0, 1***

(b) *Relation Algebra* for composition, choice and dual:
; is *associative, left-distributive, right-monotone*,
and also we have $(G;H)^d \approx G^d;H^d$

Typically invalid for games is *right*-distributivity: $G;(H \cup K) \approx (G;H) \cup (G;K)$. Complete axiomatizations have been given in Goranko 199**x**, Venema 200**x**. Here is a connection with logic games after all (van Benthem *SL*):

Theorem First-order evaluation games are complete for basic game algebra.

Lacking here are non-sequential *parallel* game operations. A natural stipulation for games played concurrently is the following product *G* ***x*** *H*, which involves ordered pairs of states in both component games:

$$\rho^i_{GxH}(s, t), X \text{ iff } \exists U: \rho^i_G\, s, U, \exists V: \rho^i_H\, t, V : UxV \subseteq X$$

Open Problem 14 Find the complete game algebra of *DGL* plus product.

3.3 Temporal game logics

Games with players' powers can also be described in a temporal language. E.g., van Benthem 1999 identifies this key lemma in the proof of the Gale-Stewart Theorem:

$\{G, \mathbf{E}\}\phi \vee \{G, \mathbf{A}\}\, A\, \neg\{G, \mathbf{E}\}\phi$
with A the temporal operator "always on the current branch"

This generalized determinacy holds for all games, and it says that either one player has a winning strategy, or the other has a strategy preventing the first from ever reaching a position where she has a winning strategy.

Open Problem 15
Axiomatize the temporal logic of players' powers over arbitrary games.

The temporal logic *ATL* used in computer science does part of this job, but it does not seem to have sufficient expressive power.

On the analogy of *DGL*, one would also want to add game operations here. The best known system for that is linear logic, as mentioned above.

Open Problem 16 Design and axiomatize a temporal version of linear logic which can also speak of truth on branches of infinite games.

There are many epistemic versions of temporal logics, such as the system of Fagin et al. 1995, or *ATEL*(van der Hoek & Wooldridge 200**x**). These also make sense for the preceding systems. E.g., non-determined games are of the essence in linear logic, and they naturally involve imperfect information (see our next Section 4) and lack of knowledge by players, even in the finite case.

Open Problem 17 Find good epistemic versions of *DGL* and Linear Logic.

In particular, standard game semantics for linear logics has to use infinite games of perfect information as non-determined counter-examples to Excluded Middle. With game s of imperfect information, finite models might suffice for completeness.

4 Knowledge, Belief and Update

As stated before, logics of knowledge, belief, and other paraphernalia of philosophical logic have entered game theory through the analysis of 'rationality' underpinning the choice of specific strategic equilibria. In particular, epistemic logic plays several roles in this connection, and it raises some new questions.

3.1 Epistemic characterizations

First, there is an extensive literature on epistemic characterizations of various game-theoretic notions of equilibrium, including Iterated Removal of Strictly Dominated Strategies, or Perfect Rationalizability. The format is as follows:

> Model *M(G)* of game *G* satisfies epistemic condition *E* iff the only strategy profiles occurring in the model are those satisfying game solution concept *C*.

Often, the condition *E* is some form of iterated or common knowledge of rationality among players. De Bruin 2004 is a survey, as well as a proposal for a uniform logical format of analysis for these results. Still, the current literature consists mainly of a small bunch of such characterization results, without any obvious system.

Open Problem 18 Find a logical analysis of epistemic characterization results which establish a systematic equilibrium theory.

The full form of such results in game theory usually also involves notions of *belief* and *probability* – where the latter serves both to define equilibria requiring mixed strategies, and also beliefs of players in the sense of subjective probability. This makes the logical analysis more complex, and it points to further issues below.

3.2 Update about the future: knowledge and belief

The main epistemic characterization results have been about games in strategic form. But knowledge and belief also come up naturally in thinking about player's moves in an extensive game, and their deliberations about the remainder to be played. In this area, one would want to move away from the excessive emphasis on Backward Induction, and consider other scenarios – cf. van Benthem 2005 for some alternatives, such as 'repaying past favours'. Presumably, existing logic of update and revision can handle the game situation, but the issue is what good they can do.

Open Problem 19 Use epistemic update and belief revision to obtain a richer set of solution methods for extensive games.

3.3 Imperfect information and dynamic-epistemic logic

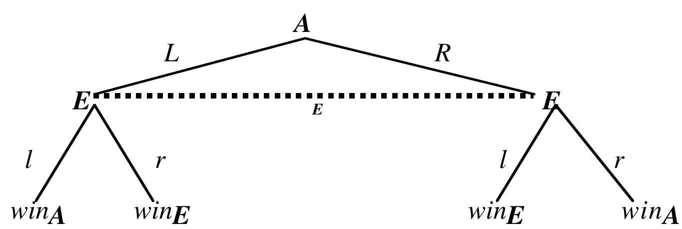
Knowledge also prepares the way for games of imperfect information, where players do not know their current situation exactly. Card games are a good example, and so is warfare. It is easy to add epistemic structure to games of imperfect information. The usual game-theoretic trees with information sets, or dotted lines, are already models for a combined dynamic-epistemic language combining action modalities $[\pi]\phi$ with epistemic operators. Here, knowledge assertions occur for individuals or groups:

$\boldsymbol{M}, s \models K_i\, \phi$ player i knows that ϕ is the case if ϕ is true in all states i~indistinguishable from s

$\boldsymbol{M}, s \models C_G\phi$ ϕ is common knowledge in the group G if ϕ is true in all states reachable from s by a finite sequence of accessibility steps for any player.

This is a modal language again, but now over models consisting of worlds with equivalence relations for players' indistinguishabilities, and typical interactions.

A combined dynamic-epistemic language *DEL* can describe many situations of interest. The following is a brief survey of van Benthem 2001, which explains the *DEL* view of imperfect information games for both extensive and strategic forms. For the sake of concrete illustration, consider the following two-step game:



Allowing only strategies that can be played despite her uncertainty, ***E*** has only two: 'left', 'right'. Note that determinacy is lost here: neither player has a winning strategy. The following formulas describe player ***E***'s plight at her right-most turn:

(a) $K_E(\langle l\rangle win_E \vee \langle r\rangle\ win_E)$

E knows that some move will make her win

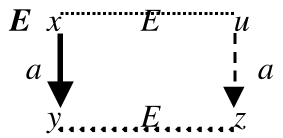
(b) $\neg K_E \langle l\rangle win_E) \wedge \neg K_E \langle r\rangle win_E)$

there is no particular move of which *E* knows that it will make her win.

The complete logic of this system is a well-known fusion of *PDL* plus epistemic *S5*. It has been proposed by Moore 1985 as a logic of *planning* in situations where agents do not know all the relevant information. In particular, no interaction axioms occur between knowledge and action modalities. When these occur, they express special features of agents. This is brought out by standard modal *correspondence* analysis.

Theorem The *DEL* formula $K_E[a]\phi \rightarrow [a]K_E\phi$ is true in a frame iff *E* has Perfect Recall in the sense that, if $R_a\, xy$ and $y \sim_E z$, then there exists a u with $x \sim_E u$ and $R_a\, uz$:

The latter condition requires this commutative diagram in the game trees:



More complex versions of Perfect Recall yield to exactly the same analysis (Bonanno 200**x**). And one can also analyze other types of player, such as those having finite-state memories. One virtue of logic analysis here is the discovery that complexity of valid reasoning may differ widely for these different types of agent (Halpern & Vardi 199**x**). Here are two questions about further possible connections with game theory:

Open Problem 20 Use *DEL* to analyse the various notions of game-theoretic equilibrium that have been proposed for imperfect information games.

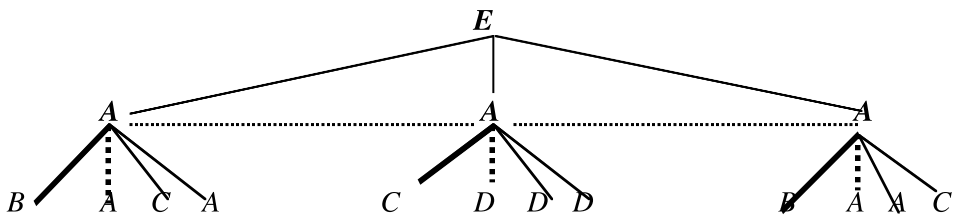
Open Problem 21 Analyze the 'Harsanyi trade-off' between incomplete information about the future in perfect information games and imperfect information games in logical *DEL* terms.

One can also add a calculus of strategies again, in a *PDL*-style extension. In game theory, imperfect information games involve *uniform strategies*, which prescribe the same move for a player across her uncertainty link. These can be correlated with programs whose only test conditions are formulas which the relevant players know. (cf. Fagin et al. 1995 on 'knowledge programs', van Benthem 2001).

Incidentally, *DEL* also has a counterpart at the *power level*. As we noted above, any two families of sets satisfying only the monotonicity and consistency conditions on powers can be realized by means of a two-step *imperfect* information game. Here is an example displaying the necessary tricks. Suppose we are given:

powers for ***E*** *{A, B, C}, {C, D}* powers for ***A*** *{B, C}, {A, D}*

An appropriate game with just these powers is



But the language of powers, or even the full *DGL* of Section 3 can be easily combined with epistemic logic. In a sense (van Benthem 200x), Hintikka's '*IF* logic' of Section 5 below is already a calculus of game operations plus implicit knowledge operators.

3.4 Information update in games

'Dynamic epistemic logic' is also used in another sense these days, as a name for concrete systems that update information. Van Benthem 2005 is an extensive survey of open problems in this area. Here, we just mention a few issues related to games. Let's use the letter combination *EDL* for this second sense of dynamics., which may be viewed as an instantiation of *DEL* in the above more abstract sense. One now adds *PDL* style operators referring to informational actions. The basic case is *public announcement !A* of a proposition *A*, removing all worlds from the current model ***M*** where *A* does not hold to obtain the relativized submodel ***M***|*A*:

$$\boldsymbol{M}, s \models [!A]\, \phi \qquad \text{iff} \qquad \boldsymbol{M}|A, s \models \phi$$

This says that, after truthful announcement of *A*, ϕ holds. Public announcement has a complete decidable logic (cf. van Benthem, van Eijck & Kooi 2005 for an up-to-date version), whose axioms essentially compute $[!A]\phi$ by *relativizing* ϕ to *A*.

More complex informative events over an epistemic model ***M*** involve hiding of one's own actions, or partial observation by others. This happens frequently in ordinary communication, where we whisper in lecture theatres, use *bcc* in emails to make information flow in complex manners, or just cheat. A sharp focus for all this are parlour games where not all players can see all details of some current move.

To deal with such more sophisticated informative events, the right update mechanism is not just world elimination. One has to first form an *event model* ***A*** of all relevant actions with their preconditions, where we encode which agent can distinguish which events. E.g., if I read my card in your view, you know that I am performing one of a number of possible reading actions, but you do not know which one precisely. And If I read my card secretly, you even think my action is the same as doing nothing.

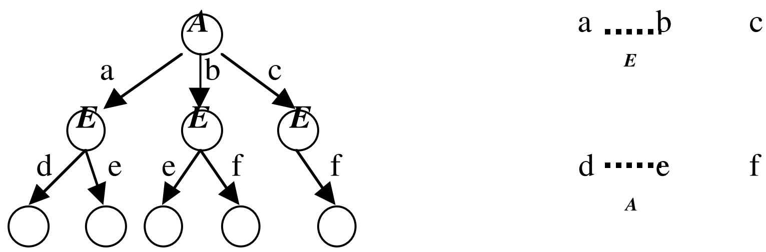
The next stage then computes a *product model* ***MxA*** whose worlds are pairs *(s, a)* of current states *s* in ***M*** and those events *a* from ***A*** which are possible at *s* (Baltag, Moss & Solecki 199**x**). For game trees, this specializes as follows:

> Take a game with current state *x*, and uncertainty relations $\sim_i$ among the nodes at *x*'s tree level computed so far. Let a new move be made. The new states are the nodes at the next level of the game tree, which can be identified with ordered pairs (previous state, action last made). Now we define the uncertainties at the next tree level as follows:
>
> $(y, b) \sim_i (z, c)$ iff $y \sim_i z$ *and* $b \sim_i c$

Thus, new uncertainty equals 'old uncertainty + indistinguishable actions'.

Example Updates during play: propagating ignorance along a game tree

Game Tree Action Diagram

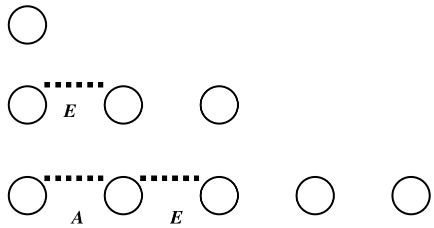


Here are the successive updates:

stage 1

stage 2

stage 3



One can analyze this update mechanism by its dynamic-epistemic properties. For product update, we find two major ingredients, the earlier *Perfect Recall*, and

> *Uniform No Learning*: if two actions are ever indistinguishable , then they will never make indistinguishable states distinguishable.

Uniform No Learning typically validates a converse of the earlier knowledge-action interchange law for players who obey it. Note that this means that product update is not a neutral stipulation: it only works for idealized agents.

The impact of such assumptions may be studied in the following mathematical setting. One forms infinite trees whose nodes are finite sequences of events (cf. Fagin et al. 1995, Parikh & Ramanujam 2002), and allows arbitrary uncertainty relations for agents between nodes. Such uncertainty patterns will encode what agents know, and more generically, what types of agent are present. One important special kind of such trees arises from product update. Define the infinite tree-like structure

*Tree(**M, A**)* to consist of all successive product levels
M**, **MxA**, **(MxA)xA, ...

The above game tree was an example. Now we can ask which patterns of epistemic uncertainty relations are characteristic for the latter setting. Van Benthem & Liu 2004 give the following representation result

Theorem An arbitrary epistemic event tree is isomorphic to some model *Tree(**M, A**)* iff its uncertainty relations for all agents satisfy both Perfect Recall and Uniform No Learning.

Van Benthem & Liu 2004 also discuss the possibility of a classification of agents' *strategies* in terms of restrictions on their format of definability. General epistemic tree structures support epistemic temporal languages, just as much as dynamic epistemic logics of information update (cf. van Benthem 2005 for a first attempt):

Open Problem 22 Can the theory of *EDL* be generalized to an epistemic temporal logic allowing more general temporal pre- and postconditions for events?

3.5 Diversity of agents

The preceding characterization result can be extended to other types of agent, corresponding to other structural constraints. For instance, it is easy to characterizes those corresponding to *memory-free agents*.

Open Problem 23 Find a general *EDL*-style classification of types of agent based on logically expressible dynamic-epistemic constraints.

Perhaps the more interesting question, however, is about groups of agents with different logical behaviour. How do we detect the 'memory-type' of an opponent, and can we take advantage of it, once we know? The movie "Memento" provides nice examples of this scenario, and so do many other more *SF*-like current films.

3.6 Other uses of EDL: solution algorithms for strategic games

Van Benthem 2002 uses scenarios with repeated announcement of 'Rationality' of players to analyze game solutions, not as sets of strategy profiles, but rather directly on the algorithmic procedures themselves which solve games. This works over a simple account of *game models* as sets of strategy profiles where players know their own action, but not that of the others. Standard algorithms iteratively eliminate profiles from the initial model, in a process whose steps can be viewed as announcements of some suitable epistemic statement. In any model ***M***, any statement ϕ, repeated sufficiently often, gets to a fixed-point #(**M**, ϕ). Either the latter submodel is empty (ϕ is *self-defeating*), or it is non-empty, while announcing it has no further effect, so that the statement has become common knowledge in #(**M**, ϕ) (ϕ is *self-fulfilling*).

Theorem The algorithm SD^{ω} of Iterated Removal of Strictly Dominated Strategies produces just the models #(M, WR) where WR is the statement that no one plays an action for which there is one that she knows to be better.

This type of result establishes a correspondence between (a) epistemic assertions whose announcement can be iterated, and (b) game solution algorithms. Van Benthem 2002 gives an illustration, defining a new solution algorithm stronger than SD^{ω} using the statement that "no one thinks that her current action is ever a worst response".

He also shows that for existential epistemic assertions ϕ, the solution models #(**M**, ϕ) can be defined inside the initial model by greatest fixed-point formulas in an *epistemic μ–calculus*. But the general case requires an extension with *inflationary fixed-points*.

Open Problem 24 Develop the dynamic epistemic analysis of game solution procedures more systematically.

General epistemic inflationary fixed-point logic is ill-behaved. E.g., it is undecidable (Dawar, Grädel & Kreutzer 2004) and other quirks occur, too.

Open Problem 25 Find well-behaved fragments of epistemic inflationary fixed-point logic which suffice for game analysis.

3.7 Belief revision

EDL may be viewed also as a logic for update of *belief*, rather than knowledge. But, in the context of games, we also seem to need *belief revision*, as we encounter unexpected behaviour of our opponents which contradicts our expectations so far. Now, as explained in the companion survey van Benthem 2005, product update does not achieve belief revision. But its methodology can be extended to a system which does, by enriching models with *plausibility values* for worlds according to agents (Spohn 1988). Then, product update can be extended to plausibility update in a natural manner (Aucher 2003), supporting the same sorts of perspicuous logics which are the trademark of *EDL*. But there is a difference in spirit. As pointed out in Liu 2004, *diversity of revision policies* (from more radical to more conservative) is a desideratum in belief revision theory, and hence no unique update rule can work for everyone. Her proposal is to parametrize the plausibility assignment function to pairs *(s, a)* with weight for the past world *s* and the last-observed event *a*. This is like parametrized update rules in inductive logic or Bayesian probability.

Open Problem 26 Apply the Spohn-Aucher-Liu analysis to extensive games.

Open Problem 27 Extend the analysis of game solution with iterated epistemic statements to a scenario where players' beliefs are taken into account.

3.8 Preference Dynamics

While we are dynamifying various aspects of games, we might also consider doing the same with preferences. One might imagine that there are actions which change preferences. Zarnic 199**x** analysis goals and planning in this way, with commands of the form *FIAT* ϕ: "make ϕ become the case". Yamada 2004 takes the same thinking to *deontic logic*. A command like "you ought to make sure that ϕ"is some authority's instruction to give possible future outcomes satisfying ϕ high(er) priority. (Related ideas are in Tan 199**x**.) Thus, one can have an account of dynamic commands and their effects in the same style as *EDL*. But there are non-trivial issues here, having to do with the fact that *EDL* events are 'precondition oriented'. They convey information about the situation when the event took place, while we just have to see or deduce what will hold afterwards. By contrast, commands are *postcondition-oriented*. They tell us to do something that will make sure a certain condition will hold afterwards.

Open Problem 28 Develop a complete expressive dynamic logic of commands.

5 Logic Games

As mentioned above, many logical tasks themselves have a game character. Most of these involve two-person zero-sum games of perfect information, some with finite extensive game trees, some allowing infinite runs. Van Benthem 1999 has chapters on most of the basic varieties. We run through a few basic cases.

Evaluation games Semantic evaluation games exist for most logical languages. The original case are Hintikka games for first-order logic, whose basic feature is an equivalence between

(a) truth of ϕ in some model $\boldsymbol{M}$ with assignment s,
(b) the existence of a *winning strategy* for Verifier
in the associated evaluation game *game*($\boldsymbol{M}$, s, ϕ).

These games have a finite length measured by the quantifier depth of the formula. With this bridge, many logical and game-theoretic notions can be correlated. E.g., the game-theoretic import of Excluded Middle is that evaluation games are determined by Zermelo's Theorem. More complex games are needed for more expressive languages, such as *LFP(FO)* with monotone fixed-point operators. These involve infinite runs, and counting of parity of infinitely recurring μ– or ν–subformulas.

Open Problem 29 Find a systematic logical use for the surplus information in the game account, viz. the different specific winning strategies for players.

Open Problem 30 Develop an abstract model theory of logical languages based on a natural classification of game types..

Model construction games Games for constructing models exist in great variety (cf. Hodges 1985). One simple option is to turn semantic tableaus into games. Now there can be infinite branches, corresponding to the construction of an infinite model. The tableau rules and processing procedure can be manipulated in many different ways for this. Here a model exists for a given set of statements iff Builder has a winning strategy, and different such strategies even encode different possible models directly. This setting also suggests introducing new vocabulary, such as an explicit instruction for dealing with a true universal quantifier more than once.

Open Problem 31 Use variations on tableau games to model various substructural (categorial, linear) versions of first-order logic.

Proof games and argumentation The oldest logic games are Lorenzen's for dialogue, where the main result is that sequents are intuitionistically provable iff the Proponent of the conclusion has a winning strategy against an Opponent granting the premises. Even more constructively, there is an effective correspondence between

(a) *proofs*,
(b) *winning strategies* in the dialogue game.

These systems are driven by a mixture of 'logical rules' for dealing with complex statements, and 'procedural rules' setting the schedule and other points of order. More informal versions of Lorenzen games, with both features, are used in argumentation theory, as a model for rational debate.

Open Problem 32 Analyse the procedural component of dialogue games more systematically, and find a systematic prediction what logic comes out of what package of logical and procedural rules.

Open Problem 33 Compare dialogue games with model construction games, and also the co-existence of strategies-as-proofs and strategies-as-models.

By the way, van Benthem 2004 has a more gloomy game analysis of argumentation, as involving little logic, but rather a calculus of values of arguments decreasing with familiarity, and making one's statements, whether weak or strong, with the right timing.

Model comparison games The most widely used logic games are Ehrenfeucht-Fraïssé games of model comparison, where two models ***M***, ***N*** satisfy the same first-order sentences up to quantifier depth *k* iff Duplicator has a winning strategy in the game *comp(**M**, **N**, k)* over *k* rounds. Here, the concrete correspondence is between

(a) first-order formulas distinguishing the models, and
(b) winning strategies for the 'difference player' Spoiler.

Comparison games give fine-structure to more global structural equivalences such as isomorphism, potential isomorphism, or bisimulation. Thus, we could also use them to compare extensive games! Here we just mention a technical desideratum.

Open Problem 34 Find a useful comparison game matching first-order logic with fixed-points *LFP(FO)*.

Operations on logic games Other logic games include more complex constructions, such as model extension (Hirsch and Hodkinson 200**x**) or recursion-theoretic priority arguments (Moschovakis 198x). Many of these, intuitively, involve combination of subgames. Some of these are the earlier-mentioned choices, compositions, and role switches. others involve more complex parallel product formation. Here is an illustration from van Benthem 1999.

Theorem Ehrenfeucht-Fraïssé games are isomorphic to 'interleaved products' of Hintikka evaluation games.

Open Problem 35 Find a good typology of game operations used for logic games.

Even though logic games have largely developed in isolation, there have been some cultural influences from game theory coming this way.

Imperfect information Logic games with imperfect information have been proposed by Hintikka and the '*IF* school' (Hintikka & Sandu 1997) for the purpose of analysing more freely scoped quantifier languages, with motivations from linguistics, philosophy, mathematics, and these days even quantum mechanics. Here is the well-known 'slash notation' for such a new kind of logical evaluation game:

$\forall x\ \exists y/\boldsymbol{x}\ x \neq y$.

This expresses, when played in any model ***M***, that Verifier has a uniform strategy for winning this semantic game, even when she does not know the value of *x* chosen earlier by her opponent Falsifier. At the level of powers, this game is equivalent to the reverse scope order $\exists y\ \forall x/\boldsymbol{y}\ \ Rxy$. This observation is just one instance of general *IF* logic, for which we refer to Sandu & Tulenheimo 2005. In particular, the full power of the system is equivalent to that of existential second-order logic. But Tulenheimo 2004 shows that some *IF* variants of modal logic remain decidable. In addition to the host of more specialized questions in this technical area, we mention

Open Problem 36 What is the complete game algebra of *IF* logic?

Open Problem 37 What are natural decidable fragments of the *IF* language?

The move from sequential to uniform strategies makes sense for any logic game.

Open Problem 38 Develop *IF* versions of games of proof or model comparison.

Preferences Finally, one can also add more finer preferences to logic games, and introduce more sophisticated equilibria than just those far, being 'winning strategy versus any counter-play '. Harrenstein 2004 develops some quite original extensions of logical notions of consequence along these lines. Other possibilities abound.:

(a) Measure *effort*, in making branch length to end node
an additional factor in the pay-offs for players.

(b) Form new games out of existing logical tasks.

A good source for the latter is computing updates in Section 4. Parts of the update universe can be turned into *conversation games* with a restricted set of available assertions, where the point is for one player to be the first to find out the actual world. Nash values for these games can be hard to compute, as complexity increases quickly. Winning strategies clearly have something to do with solving the *Reachability Problem* from one epistemic model to another by a sequence of admissible actions satisfying certain epistemic assertions.

Open Problem 39 Develop a general theory of conversation games.

6 Further Topics

This survey of issues and questions, broad as it is, has still left out a number of issues that are crucial to games. We conclude by merely listing a few directions:

Probability Integrate all of the above with probability theory, Bayesian update (cf. van Benthem 2002 for general product update in this setting), and mixed strategies. This trend is emerging, interestingly, in *IF* logic with Hintikka's use of 'generalized Skolem functions'. to deal with quantum-mechanical particles whose positions and momenta satisfy the Heisenberg Uncertainty Principle.

Infinite and evolutionary games Infinitely repeated games are essential in modern game theory, starting from Axelrod 198**x** on the emergence of cooperation in Prisoner's Dilemma encounters, and continuing into modern evolutionary game theory (*XYZ*). A systematic connection between such games and those found in logic seems rewarding. E.g., how does the strategy calculus of infinite linear logic games, with its emphasis on memory-free strategies like Copy-Cat (i.e., Tit-for-Tat) or finite-automaton computable strategies relate to similar topics in game theory? More generally, evolutionary game theory turns around the mathematics of dynamical systems – and its relationship with epistemic temporal and update logics seems a potentially important short-term/long-term interface.

Topics to be added

Parallel/simultaneous action