Logics of strategies and preferences

Sujata Ghosh[1] and R. Ramanujam[2]

[1] Indian Statistical Institute, Chennai, India
sujata@isichennai.res.in
[2] Instititute of Mathematical Sciences, Chennai, India
jam@imsc.res.in

Abstract. Reasoning about games involves elucidating the rational structure of preference that players have over outcomes, as well as the strategies they employ to achieve their preferred outcomes. This involves mutual intersubjectivity of epistemic attitudes. We discuss some propositional modal logics of strategic interaction and point to interesting questions for further research.

1 Introduction

Games are models of interaction where individual players (or agents) make choices, and obtain outcomes based on what everyone chooses. They have preferences over the outcomes, so each player would make a choice anticipating what others might do, so that they can yet obtain the best possible outcome.

Such a simple description already leads to interesting notions. A player might want to get the best that she can, *no matter* what others do, simply choose the best outcome possible for herself. She might actually do better if others might choose appropriately, but she might not want to make assumptions about others' behaviour. If everyone behaves like this, we have a situation where they all choose conservatively, and we would have a *dominant strategy equilibrium*. This is when everyone acts unilaterally, but still the outcome is stable: none of the players would choose differently, even after being informed of the others' choices.

Alternatively, an agent may not settle for this; he can reason that everyone will want the best, and hence will assume others to do the same. If it does turn out that everyone does this, we have a *Nash equilibrium*. Again, such an outcome would be stable, now in a slightly different sense: none of the players would have any incentive to unilaterally deviate from this choice.

Note that the latter behaviour involves mutual intersubjectivity: a player's decision depends on what she believes that others would do, but in turn what they do depends on what they believe she would do, and so on. This iteration of epistemic attitudes constitutes the foundational justification of the equilibrium notion. See [Osborne and Rubinstein, 1994, Perea, 2012] for a detailed discussion of the basic notions of game theory and their epistemic foundations.

An underlying assumption in such reasoning is that players are **rational**: their awareness of their own preferences, ability to introspect on their actions, beliefs about others' preferences and capacity to act, are all based on logically consistent reasoning. Indeed, not only are players rational, but believe that everyone is rational, that everyone believes this, and so on.

This brief discussion suggests that the logical foundations of game theory involve indexical epistemic attitudes, preferences, strategization towards achievement of outcomes, and ability of players. Logicians would like to pin down the logical resources necessary for the description of such interaction and for the reasoning involved. For instance, are strategies first-class objects of a logical language, or are they composite entities built from other first-class objects? If the former, what are the characteristic properties of quantification over strategies? Fixed-point operators are natural in the description of equilibria; what is their expressiveness in the presence of epistemic modal operators? And so on.

This chapter is situated in such a discourse. We take up a very small fragment of the logical study of strategic interaction, highlighting some of our own work in this arena. This involves an exploration of the rational structure of preferences, algebraic compositional structure in games and a similar structure in strategies. Indeed we can see a pleasing duality between studying strategization in composite games, and studying the composition of strategies in the course of large games. These are elements of theories of play, and the notion of player types in such theories raises interesting questions for study. We also refer to how similar theories may be built for games with a large number of players.

1.1 On preferences

While strategic reasoning focusses on the relationship between individual choices and social outcomes, reasoning about preferences focusses on the interaction between individual preferences and group preferences. This study puts the notions of aggregation and deliberation in the spotlight, and these are the two main approaches in collective decision making.

Aggregation is mostly achieved by voting where the origins of the individual preferences do not come under consideration, only the actual preferences do.There is a plethora of work on aggregation of preferences together with critical formal studies on the advantages and disadvantages of different aggregation processes (e.g., see [Arrow et al., 2002, Grüne-Yanoff and Hansson, 2009, Endriss, 2011]). However, the effectivity of the aggregation process has been questioned by philosophers like Elster [1986], Habermas [1996] and others, who point to the merits of the deliberative process which make people reflect on their preferences thus influencing possible changes. Thus, the process of deliberation is also an important aspect of group decision making - in case unanimity is reached via deliberation, there is no need to consider some (possibly artificial) aggregation process.

Sometimes, even though deliberation may not lead to unanimity in preferences, the ensuing discussion may lead to certain form of 'preference uniformity' (see how deliberation can help in bypassing social choice theory's impossibility results in [Dryzek and List, 2003]), which might facilitate their eventual aggregation. In addition, a combination of both aggregation and deliberation processes may provide a more realistic model for decision making scenarios [Perote-

Pena and Piggins, 2015]. In light of this status quo, more focus can be given on the formal study of achieving such preference uniformities, e.g., single-peaked, single-caved, single-crossing, value-restricted, best-restricted, worst-restricted, medium-restricted, or group-separable profiles (see [Bredereck et al., 2013]) and references there in for relevant results and discussion).

What we are essentially analyzing are the compositions of individual structured preferences which bring about the group preferences, and as mentioned earlier, logic can play an important role in such composition analyses. In this part of the chapter, we provide a comparative analysis of the processes of aggregation and deliberation from the logic perspective. The main objective here is to motivate a combined formal analysis of these aspects towards determining the subtle commonalities as well differences.

1.2 Related Work

A substantial part of the work discussed here originates from the seminal work of Parikh [1985] on Propositional Game Logic, which suggested algebraic game composition as a tool for logical study. Goranko [2003] looks at an algebraic characterisation of games and presents a complete axiomatization of identities of the basic game algebra. Pauly [2001] has built on this to provide interesting relationships between programs and games, and to describe coalitions to achieve desired goals. Goranko [2001] relates Pauly's coalition logics with alternating temporal logic [Alur et al., 1998], which combines temporal logic with a form of quantification over strategies. The logics presented here can be seen as a process logic extension within a branching time framework [Ramanujam and Simon, 2009].

Another major influence on the work presented here is that of Johan van Benthem (and co-authors). van Benthem [2001] considered both perfect and imperfect information games and analysed them at the local action level as well as the global outcome level. From a purely logic perspective, van Benthem [2003b] established a connection between the algebra of game operations considered in the game logics [Parikh, 1985, Parikh and Pauly, 2003] and that of logical evaluation games, showing the latter to be quite general. van Benthem [2012] highlights the importance of having logics expressing strategies explicitly which could in turn give more pragmatic models for interaction. van Benthem et al. [2011] provided a distinct dynamic perspective into the continuing studies on logic and games. Different notions of information and interaction related to agency were brought in the foray to propose a 'theory of play' interlacing the notions of logic and games. Finally, van Benthem [2014], gave a detailed description of his research agenda at the interface of logic and games. He provided a host of perspectives into the ever-continuing studies on logic and games, bringing out a number of open research problems. In his words, 'this book is meant to open up an area, not to close it'.

We discuss other related work in context, in the course of developing the notions. Parts of the work presented here on games arose from joint work with Soumya Paul and Sunil Simon.

2 Games as models of interaction

Consider the classic situation of two children wanting to divide a piece of cake among themselves. The solution is to let one child divide the cake and let the other choose which piece she wants. Each child wants to maximise the size of his piece, and therefore this process ensures fair division. The first child cannot complain that the cake was divided unevenly and the second child cannot object since she has the piece of her choice. This is a very simple example of a game where two players have conflicting interests and each player is trying to maximize his payoff. The final outcome of the division depends on how well each child can anticipate the reaction of the other and this makes the situation game-theoretic.

A game can be presented by specifying the players, the strategies available to each player and the payoff for each player. In the case of a two person game, this can be presented efficiently in a matrix form. For instance the cake cutting game can be represented using the matrix shown in Figure 1. We will refer to the players as *cutter* and *chooser*. Here both players have two strategies, each can choose to cut the cake evenly or to make one piece bigger than the other, which corresponds to picking one of the rows of the matrix. Chooser can choose the bigger piece or the smaller piece, which corresponds to picking one of the columns of the matrix. The outcome for the cutter, after both the players choose their strategies, is the corresponding entry in the matrix. For instance, if the cutter chooses to make one piece bigger and the chooser picks the bigger piece, then the outcome will be that the smaller piece goes to the cutter (bottom left cell). The chooser's outcome is the complement of the cutter's. An equivalent representation of the game can be obtained by replacing the outcomes with numbers representing *payoffs* as shown in Figure 2.

	Choose bigger piece	Choose smaller piece
Cut the cake evenly	Half of the cake	Half of the cake
Make one piece bigger	Small piece	Big piece

Fig. 1. Cutting a cake: Instinctive payoffs

The cake cutting game captures the situation of pure conflict, where cutter's gain is chooser's loss and vice-versa. Such games are called *zero-sum* or *win-loss* games. If the cutter had the option of choosing any of the four available outcomes, he would prefer to have the big piece. However, he realizes that expecting this outcome is highly unrealistic. He knows that if he were to make one piece bigger, then the chooser will pick the bigger piece leaving him with the remaining smaller one. If he divides evenly, then he will end up with half of the cake. The cutter's

	Choose bigger piece	Choose smaller piece
Cut the cake evenly	0	0
Make one piece bigger	-1	1

Fig. 2. Cutting a cake: Cardinal payoffs

choice is really between the smaller piece and half of the cake. Therefore he will choose to take half of the cake (top left cell) by making an even split of the cake. This amount is the maximum row minimum and is referred to as the *maximin*.

Now consider a variation of this game where the chooser is required to announce her choice (big or small piece) before the cake is cut. This does not change the situation: The chooser would still choose a bigger piece irrespective of how the cutter divides the cake. That is, the chooser looks for the minimum column maximum (*minimax*) value, which happens to be the top left cell. In this example, the maximin value and the minimax value both happen to be the top left cell. For a game when the maximin value and the minimax value is identical, the outcome is called the *saddle point*. When a game has a saddle point it is the expected rational play since either player cannot unilaterally improve his payoff. A win-loss game is said to be *determined* if a saddle point exists.

Formally a two-player zero-sum game where player 1 has m strategies and player 2 has n can be represented by an $m \times n$ array A, where the (i, j)th entry $a_{i,j}$ represents the payoff of player 1 when he chooses the strategy i and player 2 picks strategy j. The payoff for player 2 for the corresponding entry is $-1 \times a_{i,j}$. Note that non-zero-sum payoffs can easily be represented by replacing each matrix entry by a tuple of payoffs for each player.

	Heads	Tails
Heads	1	-1
Tails	-1	1

Fig. 3. Matching pennies

Unfortunately not all games have saddle points. One of the simplest examples is the game of "Matching pennies" depicted in Figure 3. In this game two players simultaneously place a penny (heads or tails up). When both the pennies match, player 1 gets to keep both. If the pennies do not match, then player 2 gets to keep both. Its easy to see from the payoff matrix that maximin is -1 whereas minimax is 1. It is well known that the best way of playing matching pennies is to play heads with probability half and tails with probability half. This amounts to a *mixed strategy* rather than the *pure strategy* of picking an action with absolute certainty. The minimax theorem [von Neumann and Morgenstern, 1947] asserts that for all two player zero-sum games, there is a rational outcome in mixed strategies.

The theory can be extended from zero-sum objectives to non-zero-sum objectives with more than just two players. In this case the outcome of the game will specify a payoff for each of the players. The commonly used solution concept in this context is that of Nash equilibrium which corresponds to a profile of strategies, one for each player which satisfies the property that no player gains by unilaterally deviating from his equilibrium strategy. Nash [1950] formulated this notion of equilibrium for multiplayer non zero-sum games and proved the analogue of the min-max theorem for such games. The result states that for all finite multiplayer games, there exists a mixed strategy (Nash) equilibrium profile.

Much of the mathematical theory developed for games talks about existence of equilibrium and does not shed light on how the players should go about playing the game. For two-person zero-sum games, one can show that the maximin theorem is equivalent to the LP (linear programming) duality problem. Therefore construction of optimal strategies is possible using linear programming techniques [von Stengel, 2002]. For two person non zero-sum games, optimal strategies can be constructed using techniques for solving the linear complementarity problem as shown in [Lemke and Howson, 1964]. For a multi-player game, Nash's theorem talks of existence of equilibrium but it is not known how to actually construct the equilibrium strategy.

Strategic form games give a highly abstracted presentation of a game. The representation typically assumes "small" games where the structure of the strategy (individual moves which build up to form the strategy) is absent (or abstracted away). The existence theorems suggest which strategy a player would employ in the game. However, we also need to analyse larger games where the players' actions are part of the representation. We now address reasoning in such a context.

2.1 Game logics

One natural way is to consider a large game as being built up structurally from small atomic games by means of composition. This suggests an algebraic structure in games, and one line of work in game logics proceeds by imposing a program-like compositional structure on games.

Program logics like the propositional dynamic logic (PDL) [Harel et al., 2000] have been developed to reason about programs. The idea here is to model programs as being constructed using operations like sequential composition, iteration, etc. on simple atomic programs. This compositional approach in program reasoning has been successful in the analysis and verification of programs, especially in giving us insights into the expressive power of various programming constructs. The natural extension to this methodology is to come up with a dynamic logic to reason about multi-agent programs and protocols. Game logic [Parikh, 1985] that we briefly introduced in Section 1.2 addresses this issue. Game logic (GL) is a generalisation of PDL for reasoning about determined two person games.

Let the two players be denoted as player 1 and player 2. Like PDL, the language of GL consists of two sorts, games and propositions. Let Γ_0 be a set of atomic games and P a set of atomic propositions. The set of GL-games Γ and the set of GL-formulas Φ is built from the following syntax:

$$\Gamma := g \mid \gamma_1; \gamma_2 \mid \gamma_1 \cup \gamma_2 \mid \gamma^* \mid \gamma^d$$
$$\Phi := p \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid \langle\gamma\rangle\varphi$$

where $p \in P$ and $g \in \Gamma_0$. Let $[\gamma]\varphi := \neg\langle\gamma\rangle\neg\varphi$ and $\gamma_1 \cap \gamma_2 := (\gamma_1^d \cup \gamma_2^d)^d$.

The formula $\langle\gamma\rangle\varphi$ asserts that player 1 has a strategy in game γ to ensure φ and $[\gamma]\varphi$ expresses that player 1 does not have a strategy to ensure $\neg\varphi$, which by determinacy is equivalent to the fact that player 2 has a strategy to ensure φ. The intuitive definitions of the games are as follows: $\gamma_1; \gamma_2$ is the game where γ_1 is played first followed by γ_2, $\gamma_1 \cup \gamma_2$ is the game where player 1 moves first and decides whether to play γ_1 or γ_2 and then the chosen game is played. In the iterated game γ^*, player 1 can choose how often to play γ (possibly zero times). He need not declare in advance how many times γ needs to be played, but is required to eventually stop. The dual game γ^d is the same as playing the game γ with the roles interchanged. The formal semantics is given below.

A game model $M = ((S, \{E_g \mid g \in \Gamma_0\}), V)$ where S is a set of states, $V : P \to 2^S$ is the valuation function and $E_g : S \to 2^{2^S}$ is a collection of *effectivity functions* which are monotonic, i.e. $X \in E_g(s)$ and $X \subseteq X'$ imply $X' \in E_g(s)$. The idea is that $X \in E_g(s)$ holds whenever player 1 has a strategy in game g to achieve X.

The truth of a formula φ in a model M at a state s (denoted $M, s \models \varphi$) is defined as follows:

$$\begin{array}{ll} M, s \models p & \text{iff } s \in V(p) \\ M, s \models \neg\varphi & \text{iff } M, s \not\models \varphi \\ M, s \models \varphi_1 \wedge \varphi_2 & \text{iff } M, s \models \varphi_1 \text{ or } M, s \models \varphi_2 \\ M, s \models \langle\gamma\rangle\varphi & \text{iff } \varphi^M \in E_\gamma(s) \end{array}$$

where $\varphi^M = \{s \in S \mid M, s \models \varphi\}$. The effectivity function E_γ is defined inductively for non-atomic games as follows. Let $E_\gamma(Y) = \{s \in S \mid Y \in E_\gamma(s)\}$. Then

$$\begin{aligned} E_{\gamma_1;\gamma_2}(Y) &= E_{\gamma_1}(E_{\gamma_2}(Y)) \\ E_{\gamma_1 \cup \gamma_2}(Y) &= E_{\gamma_1}(Y) \cup E_{\gamma_2}(Y) \\ E_{\gamma^d}(Y) &= \overline{E_\gamma(\overline{Y})} \\ E_{\gamma^*}(Y) &= \mu X.Y \cup E_\gamma(X) \end{aligned}$$

where μ denotes the least fixpoint operator. It can be shown that the monotonicity of E_g is preserved under the game operations and therefore the least fixpoint $\mu X.Y \cup E_\gamma(X)$ always exists.

Since game logic was designed to reason about multi-agent programs, the modelling approach is quite different from traditional game theoretic notions. Pauly [2000] presents a semantics for Game logic which is closer to the standard game-theoretic approach.

Theorem 1. *(Parikh [1985]) The satisfiability problem for Game Logic is in EXPTIME.*

Theorem 2. *Given a Game Logic formula φ and a finite game model M, model checking can be done in time $\mathcal{O}(|M|^{ad(\varphi)+1} \cdot |\varphi|)$ where $ad(\varphi)$ is the alternation depth of φ.*

A proof of this theorem can be found in Pauly [2001], Theorem 6.21 (page 122).

As shown in [Parikh, 1985], it is possible to interpret Game logic over Kripke structures. Over Kripke structures, Game logic can be embedded into μ-calculus [Kozen, 1983]. Whether Game logic is a proper fragment of the μ-calculus is not known. It is quite conceivable that model checking for Game logic is easier than model checking for the full μ-calculus. However, Berwanger [2003] shows that this is not the case.

A long standing open problem in Game logic, to give a complete axiomatization of valid formulas of the logic, was settled recently (Enqvist et al. [2019]). Parikh [1985] proposed an axiom system and conjectured that it is complete, which has now been proven. For the dual free fragment of Game logic, a complete axiomatization is presented in [Parikh, 1985].

In Game logic, starting with simple atomic games, one can construct large complex games using operators like composition and union. Due to the presence of the Box-Diamond duality $\langle\gamma\rangle\varphi \equiv \neg[\gamma]\neg\varphi$, it is easy to see that the games constructed remain determined. The compositional syntax of Game logic presents an algebra for game construction. Rather than look at arbitrarily large games, this approach gives us a way of systematically studying complex games in a structured manner and to also look at their algebraic properties. One should however note that the emphasis in this approach is to reason about games, to study the structure of games with interesting properties and definability conditions.

2.2 Extensive-form games

When games are finite, strategies are complete plans and each player has only finitely many strategies to choose from, normal form games abstract strategies and sets of choices and studies the effect of each player making a choice simultaneously. Extensive-form games retain the structure of games and we study a game as a tree of possible sequences of player moves. Then a backward induction procedure (BI procedure) can be employed to effectively compute optimal strategies for players, leading to predictions of stable play by rational players. Questions of how players may arrive at selecting such strategies and playing them, and their expectations of other players symmetrically choosing such strategies are (rightly) glossed over.

However, when we consider players as being decisive and active agents but who are limited in their computational and reasoning ability, the situation changes entirely[3]. To see this, note that the BI-procedure works bottom up on the game tree, whereas strategizing follows the flow of time and hence works top down. Hence, unless a player has access to the entire subtree issuing at a node, she cannot compute optimal strategies, however well she is assured of their existence. It is in fact for this reason that though the determinacy of chess was established by Zermelo [1913] the game remains fascinating to play as well as study even today.

Indeed resource-limited players working top down are forced to strategize *locally*, by selecting what part of the past history they choose to carry in their memory, and how much they can look ahead in their analysis. In combinatorial games, complexity considerations dictate such economizing in strategy selection. Predicting rational play by resource-limited players is then quite interesting.

Knowing that other players are also similarly limited does not trivialise the problem in any way, it only leads to more interesting *epistemic* situations. Since each player is (symmetrically) strategizing top down, not only is the strategy of each player partial, so is her expectation of the strategies being followed by other players. Such a dependence is recursive and leads to considerable complexity in epistemic attitudes.

When game situations involve *uncertainty*, as inevitably happens in the case of games with large structure or large number of players, such top-down strategizing is further necessitated by players having only a partial view of not only the past and the future but also the present as well. Once again, we are led to the notion of a strategy as something different from a complete plan, something analogous to a heuristic, whose applicability is dictated by local observations of game situations, for achieving local outcomes, based on expectations of other players' locally observed behaviour. The notion of locality in this description is imprecise, and pinning it down becomes an interesting challenge for a formal theory.

As an example, consider a heuristic in chess such as *pawn promotion*. This is generic advice to any player in any chess game, but it is local in the sense that it fulfils only a short-term goal, it is not an advice for winning the game. A more interesting example is the heuristic employed by the computer Deep Blue against Gary Kasparov (on February 10, 1996) threatening Kasparov's queen with a knight (in response to Kasparov's 11th move). The move famously slowed down Kasparov for 27 minutes, and was later hailed as an important strategy.[4] The point is that such strategizing involves more than "look-ahead".

The foregoing discussion motivates a formal study of strategies in extensive-form games, where we go beyond looking for existence of strategies for players to ensure desired outcomes, but take into account strategy structure as well.

[3] The notion of players whose rationality is also limited in some way is interesting but more complex to formalize; for our considerations perfectly rational but resource bounded players suffice.

[4] http://www.research.ibm.com/deepblue/meet/html/d.2.html

This can be carried out in two ways: one way is to consider strategies to be local partial plans on a tree. A dual approach is to keep the notion of strategy simple, but consider the game tree to be structured, and composed of many simple subgames. In this view, a strategy would be seen as a complete plan to ensure a local outcome (which may be the initiation of a desired subgame).

Thus we are led to the notion of strategy composition and game composition. We present two contrasting propositional modal logics embodying the two approaches and present complete axiom systems for the logics. The former can be seen to be in the spirit of a process logic [Harel et al., 1982] and the latter in that of dynamic logic [Harel et al., 2000]. Indeed, the latter logic is based on Parikh's game logic discussed earlier. This work is placed in the context of logical studies on games (cf. see [van Benthem, 2003a, 2012, Harrenstein et al., 2003, Bonanno, 2001]).

Before we proceed with the formal development, we observe that the study of strategy structure (rather than only the existence of strategies) may be relevant not only in games that "people play" but also in abstract game situations. For instance, consider the class of *evaluation games* studied by logicians [Ebbinghaus et al., 1996]. Indeed, such games have inspired the idea that semantics of logics may be provided using games, thus offering an "operational" or "constructive" view of logical truth [Hintikka, 1968]. Here again, the truth of a formula in a structure is equated with the existence of a strategy for an associated player. However, a computational or process interpretation of such a player (representing perhaps the evaluation or model checking algorithm) would be able to *determine* the truth or falsity of the formula only by considering the entire evaluation tree.[5] When the tree is finite and large or infinite (as in the case of fixpoint logics), bottom up constructions may not be easy or even available. Here again strategizing can be viewed top down or by bottom up composition of subgames. The work presented here is based on earlier work in games and strategies dealt with in [Ghosh, 2008] and [Ramanujam and Simon, 2008b,a]. Ghosh and Ramanujam [2011] and Paul et al. [2015] present automata theoretic accounts of the logics discussed here.

2.3 Strategy specifications

Game logic asserts the existence of strategies that achieve specified outcomes. The question remains how a player selects a strategy, or indeed, constructs a strategy. Reasoning about strategies can be carried out in a compositional manner, much as we spoke of composing games.

We conceive of strategy specifications as being built up from atomic ones using some grammar. The atomic case specifies, for a player, what conditions she tests for before making a move. These constitute positional strategies and the pre-condition for the move depends on observables that hold at the current game position and some finite look-ahead that each player can perform in terms of the

[5] unless the semantics of the logic somehow can cause truth in the structure to be determined by truth in substructures.

structure of the game tree. One elegant method is to state these preconditions as future time formulas of a simple action indexed tense logic over the observables. The structured strategy specifications are then built from atomic ones using connectives.

Let N denote a finite non-empty set of players. Let $P^i = \{p_0^i, p_1^i, \ldots\}$ be a countable set of observables for $i \in N$ and $P = \cup_{i \in N} P^i$. The syntax of strategy specifications is given by:

$$Strat^i(P^i) := [\psi \mapsto a]^i \mid \sigma_1 + \sigma_2 \mid \sigma_1 \cdot \sigma_2$$

where $\psi \in BF(P^i)$, the boolean closure of P.

The idea is to use the above constructs to specify properties of strategies. For instance the interpretation of a player i specification $[p \mapsto a]^i$ where $p \in P^i$ is to choose move "a" at every player i game position where p holds. At positions where p does not hold, the strategy is allowed to choose any move that is possible at that node. $\sigma_1 + \sigma_2$ says that the strategy of player i conforms to the specification σ_1 or σ_2. The construct $\sigma_1 \cdot \sigma_2$ says that the strategy conforms to specifications σ_1 and σ_2.

Let $\Sigma = \{a_1, \ldots, a_m\}$, we also make use of the following abbreviation.

- $null^i = [\top \mapsto a_1] + \cdots + [\top \mapsto a_m]$.

It will be clear from the semantics (which is defined shortly) that any strategy of player i conforms to $null^i$, or in other words this is an empty specification. The empty specification is particularly useful for assertions of the form "there exists a strategy" where the property of the strategy is not of any relevance.

2.4 Semantics

An extensive form game tree is a tuple $T = (S, \Rightarrow, s_0, \widehat{\lambda})$ where $\mathbb{T} = (S, \Rightarrow, s_0)$ is a tree. The set S denotes the set of game positions with s_0 being the initial game position. The edge function $\Rightarrow$ specifies the moves enabled at a game position and the turn function $\widehat{\lambda} : S \to N$ associates each game position with a player. [Technically, we need player labelling only at the non-leaf nodes. However, for the sake of uniform presentation, we do not distinguish between leaf nodes and non-leaf nodes as far as player labelling is concerned.]

Let $M = (T, V)$ where $T = (S, \Rightarrow, s_0, \widehat{\lambda})$ is an extensive form game tree and $V : S \to 2^P$ a valuation function. The truth of a formula $\psi \in BF(P)$ at the state s, denoted $M, s \models \psi$ is defined as follows:

- $M, s \models p$ iff $p \in V(s)$.
- $M, s \models \neg\psi$ iff $M, s \not\models \psi$.
- $M, s \models \psi_1 \vee \psi_2$ iff $M, s \models \psi_1$ or $M, s \models \psi_2$.

Strategy specifications are interpreted on strategy trees of T. We assume the presence of two special propositions $\mathbf{turn}_1$ and $\mathbf{turn}_2$ that specifies which player's turn it is to move, i.e. the valuation function satisfies the property

– for all $i \in N$, $\mathbf{turn}_i \in V(w)$ iff $\lambda(w) = i$.

Recall that a strategy μ of player i is a subtree of T. For a strategy specification $\sigma \in \mathit{Strat}^i(P^i)$, we define the notion of μ conforming to σ (denoted $\mu \models_i \sigma$) as follows:

– $\mu \models_i \sigma$ iff for all player i nodes $s \in \mu$, we have $\mu, s \models_i \sigma$.

where we define $\mu, s \models_i \sigma$ as,

– $\mu, s \models_i [\psi \mapsto a]^i$ iff $M, s \models \psi$ implies $out_\mu(s) = a$.
– $\mu, s \models_i \sigma_1 + \sigma_2$ iff $\mu, s \models_i \sigma_1$ or $\mu, s \models_i \sigma_2$.
– $\mu, s \models_i \sigma_1 \cdot \sigma_2$ iff $\mu, s \models_i \sigma_1$ and $\mu, s \models_i \sigma_2$.

Above, $\psi \in \mathit{BF}(P^i)$ and $out_\mu(s)$ is the unique outgoing edge in μ at s. Recall that s is a player i node and therefore by definition there is a unique outgoing edge at s.

A strategy logic: We now discuss how we may embed structured strategies in a formal logic. Formulas of the logic are built up using structured strategy specifications. The formulas describe the game arena in a standard modal logic, and in addition specify the result of a player following a particular strategy at a game position, to choose a specific move a. Using these formulas one can specify how a strategy helps to eventually *win* (ensure) an outcome β.

The syntax of the logic is given by:

$$\Pi := p \in P \mid (\sigma)_i : c \mid \neg\alpha \mid \alpha_1 \vee \alpha_2 \mid \langle a \rangle \alpha \mid \sigma \leadsto_i \beta.$$

where $a, c \in \Sigma$, $\sigma \in \mathit{Strat}^i(P^i)$, $\beta \in \mathit{Bool}(P^i)$. The derived connectives $\wedge$, $\supset$ and $[a]\alpha$ are defined as usual. Let $\bigcirc\alpha = \bigvee_{a \in \Sigma} \langle a \rangle \alpha$ and $\odot\, \alpha = \neg\bigcirc\neg\alpha$.

The formula $(\sigma)_i : c$ asserts, at any game position, that the strategy specification σ for player i suggests that the move c can be played at that position. The formula $\sigma \leadsto_i \beta$ says that from this position, following the strategy σ for player i ensures the outcome β. These two modalities constitute the main constructs of our logic.

Model: As mentioned earlier, models of the logic are of the form $M = (T, V)$ where $T = (S, \Rightarrow, s_0, \widehat{\lambda})$ is an extensive-form game tree and $V : S \to 2^P$ is a valuation function that satisfies the condition:

– For all $s \in S$ and $i \in N$, $\mathbf{turn}_i \in V(s)$ iff $\widehat{\lambda}(s) = i$.

For the purpose of defining the logic it is convenient to define the notion of the set of moves *enabled* by a strategy specification σ at a game position s (denoted $\sigma(s)$). These moves can also be thought of as those which conform to σ at s.

For a tree $T = (S, \Rightarrow, s_0, \widehat{\lambda})$, a node $s \in S$ and a strategy specification $\sigma \in \mathit{Strat}^i(P^i)$ we define $\sigma(s)$ as follows:

– $[\psi \mapsto a]^i(s) = \begin{cases} \{a\} \text{ if } \widehat{\lambda}(s) = i, T, s \models \psi \text{ and } a \in moves(s). \\ \emptyset \quad \text{if } \widehat{\lambda}(s) = i, T, s \models \psi \text{ and } a \notin moves(s). \\ \Sigma \quad \text{otherwise.} \end{cases}$
– $(\sigma_1 + \sigma_2)(s) = \sigma_1(s) \cup \sigma_2(s)$.
– $(\sigma_1 \cdot \sigma_2)(s) = \sigma_1(s) \cap \sigma_2(s)$.

We say that a path $\rho_s^{s'} : s = s_1 \overset{a_1}{\Rightarrow} s_2 \cdots \overset{a_{m-1}}{\Rightarrow} s_m = s'$ in T conforms to σ if $\forall j : 1 \leq j < m$, $a_j \in \sigma(s_j)$. When the path constitutes a proper play, i.e. when $s = s_0$, we say that the play conforms to σ. The following proposition is easy to see from the definition.

Proposition 1. *Given a strategy $\mu = (S_\mu, \Rightarrow_\mu, s_0, \widehat{\lambda}_\mu)$ for player i along with a specification σ, $\mu \models_i \sigma$ iff for all $s \in S_\mu$ such that $\widehat{\lambda}_\mu(s) = i$ we have $out_\mu(s) \in \sigma(s)$.*

For a game tree T and a node $s \in S$, let T_s denote the tree which consists of the unique path $\rho_{s_0}^s$ and the subtree rooted at s. For a strategy specification $\sigma \in Strat^i(P^i)$, we define $T_s \upharpoonright \sigma = (S_\sigma, \Rightarrow_\sigma, s_0, \widehat{\lambda}_\sigma)$ to be the least subtree of T_s which contains the unique path from s_0 to s and satisfies the property: for every $s_1 \in S_\sigma$,

– if $\widehat{\lambda}_\sigma(s_1) = i$ then for all s_2 with $s_1 \overset{a}{\Rightarrow} s_2$ and $a \in \sigma(s_1)$ we have $s_1 \overset{a}{\Rightarrow}_\sigma s_2$ and $\widehat{\lambda}_\sigma(s_2) = \widehat{\lambda}(s_2)$.
– if $\widehat{\lambda}_\sigma(s_1) = \bar{\imath}$ then for all s_2 with $s_1 \overset{a}{\Rightarrow} s_2$ we have $s_1 \overset{a}{\Rightarrow}_\sigma s_2$ and $\widehat{\lambda}_\sigma(s_2) = \widehat{\lambda}(s_2)$.

The truth of a formula $\alpha \in \Pi$ in a model M and position s (denoted $M, s \models \alpha$) is defined by induction on the structure of α, as usual.

– $M, s \models p$ iff $p \in V(s)$.
– $M, s \models \neg\alpha$ iff $M, s \not\models \alpha$.
– $M, s \models \alpha_1 \vee \alpha_2$ iff $M, s \models \alpha_1$ or $M, s \models \alpha_2$.
– $M, s \models \langle a \rangle\alpha$ iff there exists s' such that $s \overset{a}{\Rightarrow} s'$ and $M, s' \models \alpha$.
– $M, s \models (\sigma)_i : c$ iff $c \in \sigma(s)$.
– $M, s \models \sigma \leadsto_i \beta$ iff for all s' such that $s \Rightarrow_\sigma^* s'$ in $T_s \upharpoonright \sigma$, we have $M, s' \models \beta \wedge (\mathbf{turn}_i \supset enabled_\sigma)$.

where $enabled_\sigma = \bigvee_{a \in \Sigma} (\langle a \rangle \top \wedge (\sigma)_i : a)$ and $\Rightarrow_\sigma^*$ denotes the reflexive, transitive closure of $\Rightarrow_\sigma$.

Figure 4 illustrates the semantics of $\sigma \leadsto_1 \beta$. It says, for any 1 node β is ensured by playing according to σ; for a 2 node, all actions should ensure β.

The notions of satisfiablility and validity can be defined in the standard way. A formula α is satisfiable iff there exists a model M and s such that $M, s \models \alpha$. A formula α is said to be valid iff for all models M and for all nodes s, we have $M, s \models \alpha$.

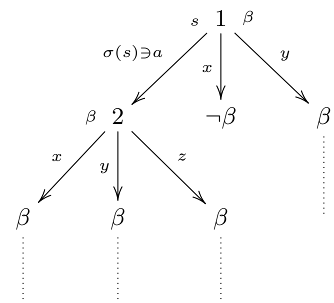


Fig. 4. Interpretation of $\sigma \leadsto_i \beta$

Axiom system: We now present our axiomatization of the valid formulas of the logic. We find the following abbreviations useful:

- $\delta_i^\sigma(a) = \mathbf{turn}_i \wedge (\sigma)_i : a$ denotes that move "a" is enabled by σ at an i node.
- $inv_i^\sigma(a,\beta) = (\mathbf{turn}_i \wedge (\sigma)_i : a) \supset [a](\sigma \leadsto_i \beta)$ denotes the fact that after an "a" move by player i which conforms to σ, $\sigma \leadsto_i \beta$ continues to hold.
- $inv_{\bar{\imath}}^\sigma(\beta) = \mathbf{turn}_{\bar{\imath}} \supset \bigodot(\sigma \leadsto_i \beta)$ says that after any move of $\bar{\imath}$, $\sigma \leadsto_i \beta$ continues to hold.

The axiom schemes

(A0) All the substitutional instances of the tautologies of propositional calculus.

1. (a) $[a](\alpha_1 \supset \alpha_2) \supset ([a]\alpha_1 \supset [a]\alpha_2)$
2. (a) $\langle a\rangle\alpha \supset [a]\alpha$
3. (a) $\langle a\rangle\top \supset ([\psi \mapsto a]^i)_i : a$ for all $a \in \Sigma$
   (b) $(\mathbf{turn}_i \wedge \psi \wedge ([\psi \mapsto a]^i)_i : a) \supset \langle a\rangle\top$
   (c) $\mathbf{turn}_i \wedge ([\psi \mapsto a]^i)_i : c \equiv \neg\psi$ for all $a \neq c$
4. (a) $(\sigma_1 + \sigma_2)_i : c \equiv (\sigma_1)_i : c \vee (\sigma_2)_i : c$
   (b) $(\sigma_1 \cdot \sigma_2)_i : c \equiv (\sigma_1)_i : c \wedge (\sigma_2)_i : c$
5. $\sigma \leadsto_i \beta \supset (\beta \wedge inv_i^\sigma(a,\beta) \wedge inv_{\bar{\imath}}^\sigma(\beta) \wedge enabled_\sigma)$

Inference rules

$$(MP)\ \frac{\alpha, \quad \alpha \supset \beta}{\beta} \qquad (NG)\ \frac{\alpha}{[a]\alpha}$$

$$(Ind \leadsto)\ \frac{\alpha \wedge \delta_i^\sigma(a) \supset [a]\alpha, \quad \alpha \wedge \mathbf{turn}_{\bar{\imath}} \supset \bigodot\alpha, \quad \alpha \supset \beta \wedge enabled_\sigma}{\alpha \supset \sigma \leadsto_i \beta}$$

The axioms are mostly standard, (A3) and (A4) describe the semantics of strategy specifications. The rule $Ind \leadsto$ illustrates the new kind of reasoning in the logic. It says that to infer that the formula $\sigma \leadsto_i \beta$ holds in all reachable states, β must hold at the asserted state and

- for a player i node after every move which conforms to σ, β continues to hold.

– for a player $\bar{\imath}$ node after every enabled move, β continues to hold.
– player i does not get stuck by playing σ.

Note that this notion of strategy composition extends to infinite game trees as well. Such a consideration naturally leads us to think of *temporal logics* on game trees and incorporate strategic reasoning in them, which is the line of work initiated by *Alternating Temporal Logic* (ATL). For a detailed analysis, see **?**.

The notion of strategy composition can be seen as constituting a theory of play whereby rational players observe and respond to play, updating their strategies. These considerations are discussed extensively in papers like Bonanno [2015] and Pacuit [2015]. Our point of departure is in the use of structural composition to represent such rational deliberation.

From strategy composition to subgame composition: We have talked of local partial strategies applicable on a game tree and composing them to build a complete strategy for the entire game. A dual approach is to consider the game to be composed of subgames and combine complete strategies from each to build a strategy for the original game.

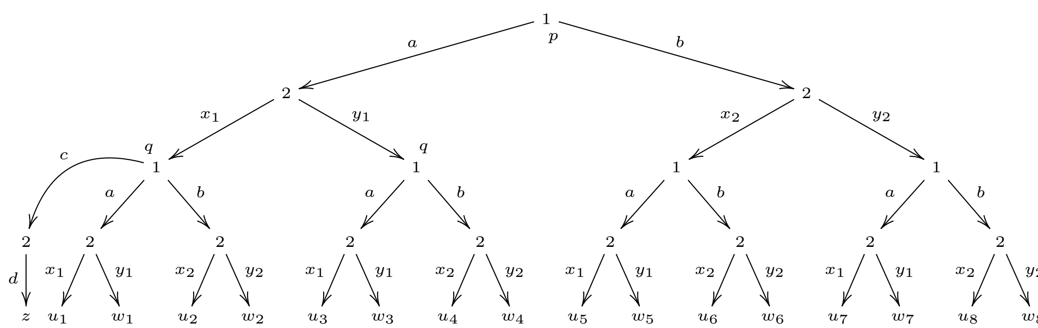


Fig. 5. Game tree T.

Consider the game tree given in Figure 5. The nodes are labelled with turns of players and edges with actions. For propositions p and q let the valuation function be as shown in the figure. Let the valuation at the leaf nodes satisfy the following constraints:

– proposition r_1 holds at states u_1, u_2, u_3, u_4 and w_1, w_2, w_3, w_4.
– proposition r_2 holds at states u_2, u_4, u_6, u_8 and w_2, w_4, w_6, w_8.
– proposition r_3 holds at states u_1, w_1, u_3, w_3.

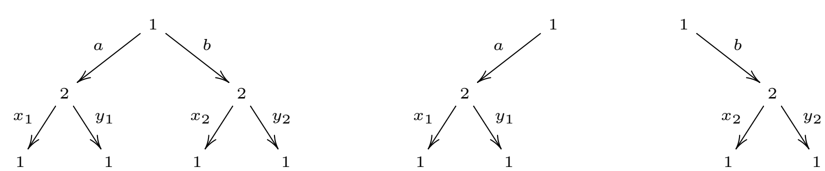
Consider the following strategy specifications:

– $\sigma_1 = [p \mapsto a]^1$.

– $\sigma_2 = [q \mapsto b]^1$.

It is easy to see that if player 1 plays action a at the root followed by action b then she can ensure the outcome r_2 no matter what player 2 does. This can be expressed in the logic as $(\sigma_1 \cdot \sigma_2) \rightsquigarrow r_2$. This however specifies a complete strategy for player 1. Now consider the specification σ_1, this is a partial specification since it does not uniquely dictate player 1's actions at any node other than the root node. It can be easily verified that any (functional) strategy of player 1 which conforms to σ_1 ensures the outcome r_1. This can be expressed as $\sigma_1 \rightsquigarrow r_1$.

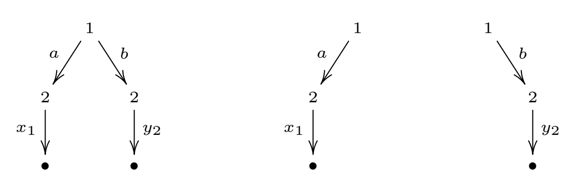
Subgame composition: The dual approach in strategizing is to consider games to be structured, composed of many simple subgames and to retain the functional notion of strategies. In this setting we can define when a simple atomic game h is enabled at a game position s of the game tree T. Intuitively this holds if it is possible to embed the structure h in the game tree T starting at the game position s. The formal definition is presented in Section 2.5.



(a) h_1 (b) μ_1 (c) μ_2

Fig. 6. Atomic game and strategy trees

As an example consider the atomic game tree h_1 given in Figure 6(a), the game h_1 is enabled at the root node of T since the structure h_1 can be embedded in T starting at the root node. There are two valid strategies μ_1 and μ_2 for player 1 in the game h_1, these are represented in Figure 6(b) and 6(c) respectively.



(a) h_2 (b) π_1 (c) π_2

Fig. 7. Atomic game and strategy trees

Similarly, for the game tree h_2 given in Figure 7(a), the valid strategies π_1 and π_2 of player 1 are given in Figure 7(b) and 7(c).

Now consider the composite game h_1 followed by h_2 (denoted by $h_1; h_2$). This corresponds to pasting the tree structure h_2 at all leaf nodes of h_1. It can be easily checked that the game $h_1; h_2$ is enabled at the root node of T. The following assertion then holds:

– In the composite game $h_1; h_2$, player 1 has a strategy to ensure the outcome r_3.

The strategy for player 1 is basically to play according to the strategy μ_1 in the game h_1 and π_1 in the game h_2. We could also consider the repetition of the game h_1 twice, in which case the following assertion holds:

– In the composite game $h_1; h_1$, player 1 has a strategy to ensure the outcome r_2.

Here player 1 needs to play according to strategy μ_1 in the game h_1 and π_2 in h_2. In other words, player 1 needs to follow the strategy which conforms to the specification $\sigma_1 \cdot \sigma_2$ in the game tree T.

A logic for compositional games: The notions of composing games and reasoning about strategies in compositional games suggest a natural logical formalism. For a composite game g, let $\mathcal{A}(g)$ denote the set of all atomic games occurring in g (the composition operators are formally defined in Section 2.5). Consider the following logic:

$$\Phi := p \in P \mid \neg\alpha \mid \alpha_1 \vee \alpha_2 \mid \langle g, \eta, i\rangle\alpha.$$

where g is a composite game and $i \in N$. The map $\eta : \mathcal{A}(g) \times N \to \bigcup_{i \in N} \Omega^i(\mathcal{A}(g))$ specifies for each atomic game $h \in \mathcal{A}(g)$ and a player i, a functional strategy for player i in h. The construct $\langle g, \eta, i\rangle\alpha$ can then interpreted as,

– $\langle g, \eta, i\rangle\alpha$: In the game g, player i can ensure the outcome α by playing according to the strategies provided by the map η.

Composition of game-strategy pairs: The compositional framework in itself is however more powerful and can be employed to capture the notion of strategic response of players. For instance, let h_1 and h_2 be the atomic games given in Figure 6(a) and 7(a). Let τ shown in Figure 8 be a strategy of player 2 in the game h_1 and let π_2 be the strategy of player 1 (shown in Figure 7(c)).

Consider the following assertion:

– If player 2 plays according to strategy τ in game h_1 then player 1 can respond with strategy π_2 in h_2 to ensure the outcome r_2.

In terms of composition of game-strategy pairs, the above assertion can be represented as,

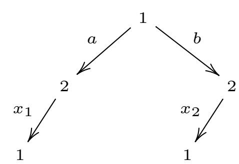


Fig. 8. Strategy τ for player 2

– $\langle (h_1, \tau); (h_2, \pi_2) \rangle r_2$.

It can be verified that the above assertion holds in the game tree T given in Figure 5. This shows that in order to express complex strategizing notions, it is useful to compose game-strategy pairs rather than to treat game composition and strategic analysis as independent entities. We now proceed to formalise this notion of composition. In fact we work with a more general framework of game-outcome pairs. A game-outcome pair in effect defines the functional strategies which ensure the specified outcome.

2.5 Compositional games

For a finite set of action symbols Σ, let $\mathcal{T}(\Sigma)$ be a countable set of finite extensive-form game trees over the action set Σ which is closed under subtree inclusion. That is, if $T \in \mathcal{T}(\Sigma)$ and T' is a subtree of T then $T' \in \mathcal{T}(\Sigma)$. We also assume that for each $a \in \Sigma$, the tree consisting of the single edge labelled with a is in $\mathcal{T}(\Sigma)$. Let $\mathbb{H}$ be a countable set and h, h' range over this set. Elements of $\mathbb{H}$ are referred to in the formulas of the logic and the idea is to use them as names for extensive-form game trees in $\mathcal{T}(\Sigma)$. Formally we have a map $\nu : \mathbb{H} \to \mathcal{T}(\Sigma)$ which given any name $h \in \mathbb{H}$ associates a tree $\nu(h) \in \mathcal{T}(\Sigma)$.

The logic: Let P be a countable set of propositions, the syntax of the logic is given by:

$$\Gamma := (h, \beta) \mid g_1; g_2 \mid g_1 \cup g_2 \mid g^*$$

$$\Phi := p \in P \mid \neg\alpha \mid \alpha_1 \vee \alpha_2 \mid \langle g, i \rangle \alpha$$

where $h \in \mathbb{H}$, $\beta \in Bool(P)$ and $g \in \Gamma$.

Models of the logic are pairs $M = (T, V)$ where $T = (S, \Rightarrow, s_0, \widehat{\lambda})$ is an extensive-form game tree and $V : S \to 2^P$ is a valuation function. The truth of a formula $\alpha \in \Phi$ in a model M and a position s (denoted $M, s \models \alpha$) is defined as follows:

– $M, s \models p$ iff $p \in V(s)$.
– $M, s \models \neg\alpha$ iff $M, s \not\models \alpha$.
– $M, s \models \alpha_1 \vee \alpha_2$ iff $M, s \models \alpha_1$ or $M, s \models \alpha_2$.

- $M, s \models \langle g, i \rangle \alpha$ iff $\exists (s, X) \in R^i_g$ such that $\forall s' \in X$ we have $M, s' \models \alpha$.

A formula α is satisfiable if there exists a model M and a state s such that $M, s \models \alpha$.

For $g \in \Gamma$ and $i \in N$, we want $R^i_g \subseteq W \times 2^W$. To define the relation formally, let us first assume that R^i_g is defined for the atomic case, namely when $g = (h, \beta)$. The semantics for composite games is given as follows:

- $R^i_{g_1;g_2} = \{(u, X) \mid \exists Y \subseteq W$ such that $(u, Y) \in R^i_{g_1}$ and $\forall v \in Y$ there exists $X_v \subseteq X$ such that $(v, X_v) \in R^i_{g_2}$ and $\bigcup_{v \in Y} X_v = X\}$.
- $R^i_{g_1 \cup g_2} = R^i_{g_1} \cup R^i_{g_2}$.
- $R^i_{g^*} = \bigcup_{n \geq 0} (R^i_g)^n$ where $(R^i_g)^n$ denotes the n-fold relational composition.

In the atomic case when $g = (h, \beta)$ we want a pair (s, X) to be in R^i_g if the game h is enabled at state s and there is a strategy for player i to ensure the outcome β such that X is the set of leaf nodes of the strategy. We make this notion precise below.

Enabling of trees: For a game position $s \in S$, let T_s denote the subtree of T rooted at s. We say the game h is enabled at a state s if the structure $\nu(h)$ can be embedded in T_s with respect to the enabled actions and player labelling. Formally this can be defined as follows:

Given a state s and $h \in \mathbb{H}$, let $T_s = (S^s_M, \Rightarrow_M, \widehat{\lambda}_M, s)$ and $\nu(h) = T_h = (S_h, \Rightarrow_h, \widehat{\lambda}_h, s_{h,0})$. The restriction of T_s with respect to the game tree h (denoted $T_s \upharpoonright h$) is the subtree of T_s which is generated by the structure specified by T_h. The restriction is defined inductively as follows: $T_s \upharpoonright h = (S, \Rightarrow, \widehat{\lambda}, s_0, f)$ where $f : S \to S_h$. Initially $S = \{s\}$, $\widehat{\lambda}(s) = \widehat{\lambda}_M(s)$, $s_0 = s$ and $f(s_0) = s_{h,0}$.

For any $s \in S$, let $f(s) = t \in S_h$. Let $\{a_1, \ldots, a_k\}$ be the outgoing edges of t, i.e. for all $j : 1 \leq j \leq k$, $t \stackrel{a_j}{\Rightarrow}_h t_j$. For each a_j, let $\{s^1_j, \ldots, s^m_j\}$ be the nodes in S^s_M such that $s \stackrel{a_j}{\Rightarrow}_M s^l_j$ for all $l : 1 \leq l \leq m$. Add nodes $s^1_j, \ldots, s^m_j$ to S and the edges $s \stackrel{a_j}{\Rightarrow} s^l_j$ for all $l : 1 \leq l \leq m$. Also set $\widehat{\lambda}(s^l_j) = \widehat{\lambda}_M(s^l_j)$ and $f(s^l_j) = t_j$.

We say that a game h is enabled at s (denoted $enabled(h, s)$) if the tree $T_s \upharpoonright h = (S, \Rightarrow, \widehat{\lambda}, s_0, f)$ satisfies the following properties: for all $s \in S$,

- $moves(s) = moves(f(s))$,
- if $moves(s) \neq \emptyset$ then $\widehat{\lambda}(s) = \widehat{\lambda}_h(f(s))$.

For a game tree T, let $\Omega^i(T)$ denote the set of strategies of player i on the game tree T and let $frontier(T)$ denote the set of all leaf nodes of T.

Atomic pair: For an atomic pair $g = (h, \beta)$ and $i \in N$, we define R^i_g as follows:

- $R^i_{(h,\beta)} = \{(u, X) \mid enabled(h, u)$ and $\exists \mu \in \Omega^i(T_u \upharpoonright h)$ such that $frontier(\mu) = X$ and $\forall s \in X$, $s \models \beta\}$.

Axiom system: We present an axiomatization of the valid formulas of the logic. We find it convenient to make use of the following notations.

We call a tree $T = (S, \Rightarrow, s_0, \widehat{\lambda})$ atomic if $|S| = 1$, i.e. the tree consists of a single node. Given an $h \in \mathbb{H}$ such that $\nu(h)$ is a non-atomic tree T and an action $a \in \vec{s_0}$, we denote by h_a the subtree of T rooted at a node s' with $s_0 \xRightarrow{a} s'$. For each $a \in \Sigma$, we define trees T_a^i and $T_a^{\bar{\imath}}$ as,

- $T_a^i = (S, \Rightarrow, s_0, \widehat{\lambda})$ where $S = \{s_0, s_1\}$, $s_0 \xRightarrow{a} s_1$, $\widehat{\lambda}(s_0) = i$ and $\widehat{\lambda}(s_1) \in N$.
- $T_a^{\bar{\imath}}$ is similar to T_a^i except for the player label at game position s_0 where we have $\widehat{\lambda}(s_0) = \bar{\imath}$.

We use h_a^i and $h_a^{\bar{\imath}}$ as names denoting these trees. That is, $\nu(h_a^i) = T_a^i$ and $\nu(h_a^{\bar{\imath}}) = T_a^{\bar{\imath}}$. We can then define $\langle a \rangle \alpha$ with the standard modal logic interpretation as follows:

- $\langle a \rangle \alpha = (\mathbf{turn}_i \supset \langle (h_a^i, \top), i \rangle \alpha) \wedge (\mathbf{turn}_{\bar{\imath}} \supset \langle (h_a^{\bar{\imath}}, \top), \bar{\imath} \rangle \alpha)$.

For $h \in \mathbb{H}$, we use the notation $h^{\checkmark}$ to denote that the tree structure $\nu(h) = (S, \Rightarrow, s_0, \widehat{\lambda})$ is enabled. This can be defined as follows:

- If $\nu(h)$ is atomic then $h^{\checkmark} = \top$.
- If $\nu(h)$ is not atomic and $\widehat{\lambda}(s_0) = i$ then
  - $h^{\checkmark} = \mathbf{turn}_i \wedge (\bigwedge_{a_j \in moves(s_0)} (\langle a_j \rangle \top \wedge [a_j] h_{a_j}^{\checkmark}))$.

The axiom schemes

1. Propositional axioms:
   (a) All the substitutional instances of tautologies of PC.
   (b) $\mathbf{turn}_i \equiv \neg \mathbf{turn}_{\bar{\imath}}$.
2. Axiom for single edge games:
   (a) $\langle a \rangle (\alpha_1 \vee \alpha_2) \equiv \langle a \rangle \alpha_1 \vee \langle a \rangle \alpha_2$.
   (b) $\langle a \rangle \mathbf{turn}_i \supset [a] \mathbf{turn}_i$.
3. Dynamic logic axioms:
   (a) $\langle g_1 \cup g_2, i \rangle \alpha \equiv \langle g_1, i \rangle \alpha \vee \langle g_2, i \rangle \alpha$.
   (b) $\langle g_1 ; g_2, i \rangle \alpha \equiv \langle g_1, i \rangle \langle g_2, i \rangle \alpha$.
   (c) $\langle g^*, i \rangle \alpha \equiv \alpha \vee \langle g, i \rangle \langle g^*, i \rangle \alpha$.
4. $\langle (h, \beta), i \rangle \alpha \equiv h^{\checkmark} \wedge \downarrow_{(h,i,\beta,\alpha)}$.

where for any $h \in \mathbb{H}$ with $\nu(h) = T = (S, \Rightarrow, s_0, \widehat{\lambda})$ we define $\downarrow_{(h,i,\beta,\alpha)}$ as follow:

- $$\downarrow_{(h,i,\beta,\alpha)} = \begin{cases} \beta \wedge \alpha & \text{if } T \text{ is an atomic game.} \\ \bigvee_{a \in \Sigma} \langle a \rangle \langle (h_a, \beta), i \rangle \alpha & \text{if } T \text{ is not atomic and } \widehat{\lambda}(s_0) = i. \\ \bigwedge_{a \in \Sigma} [a] \langle (h_a, \beta), i \rangle \alpha & \text{if } T \text{ is not atomic and } \widehat{\lambda}(s_0) = \bar{\imath}. \end{cases}$$

Inference rules

$$(MP)\ \frac{\alpha, \quad \alpha \supset \beta}{\beta} \qquad (NG)\ \frac{\alpha}{[a]\alpha}$$

$$(IND)\ \frac{\langle g, i\rangle\alpha \supset \alpha}{\langle g^*, i\rangle\alpha \supset \alpha}$$

The axioms and inference rules form an extension of the axiom system for propositional dynamic logic to trees. The difficult part is 'pushing' the enabling condition down into the program structure, which complicates the proof of completeness as well. The details are similar to the one in Ramanujam and Simon [2009].

2.6 Strategy switching and stability

We have argued that resource-limited players do not select complete strategies. Rather, they start initially with a set of possible strategies, knowledge about the game and other players' skills. As the game progresses, they compose/switch to devise new strategies. This can be specified in a syntax for strategy specification that crucially uses a construct for players to play the game with a strategy ν_1 up to some point and then switch to a strategy ν_2.

$$\Omega_i ::= \nu \in \Sigma_i \mid Strat_1 \cup Strat_2 \mid Strat_1 \cap Strat_2 \mid Strat_1 {}^\frown Strat_2 \mid (Strat_1 + Strat_2) \mid \psi?Strat$$

Using the "test operator" $\psi?Strat$, a player checks whether an observable condition ψ holds and then decides on a strategy. We think of these conditions as past time formulas of a simple tense logic over an atomic set of observables.

In the atomic case, ν simply denotes a partial strategy. The intuitive meaning of the operators are given as:

- $Strat_1 \cup Strat_2$ means that the player plays according to the strategy $Strat_1$ or the strategy $Strat_2$.
- $Strat_1 \cap Strat_2$ means that if at a history $t \in T$, $Strat_1$ is defined then the player plays according to $Strat_1$; else if $Strat_2$ is defined at t then the player plays according to $Strat_2$. If both $Strat_1$ and $Strat_2$ are defined at t then the moves that $Strat_1$ and $Strat_2$ specify at t must be the same (we call such a pair $Strat_1$ and $Strat_2$, compatible).
- $Strat_1 {}^\frown Strat_2$ means that the player plays according to the strategy $Strat_1$ and then after some history, switches to playing according to $Strat_2$. The position at which she makes the switch is not fixed in advance.
- $(Strat_1 + Strat_2)$ says that at every point, the player can choose to follow either $Strat_1$ or $Strat_2$.
- $\psi?Strat$ says at every history, the player tests if the property ψ holds of that history. If it does then she plays according to $Strat$.

The following lemma relates strategy specifications to finite state transducers, which are automata that output advice. Below note that *Strat* is a strategy specification, a syntactic object, and μ is a (functional) strategy, defined earlier to be a subtree of the tree unfolding of the game arena.

Lemma 1. *Given game arena $\mathcal{G}$, a player $i \in N$ and a strategy specification $Strat \in \Omega_i$, where all the atomic strategies mentioned in Strat are bounded memory, we can construct a transducer $\mathcal{A}_{Strat}$ such that for all $\mu \in \Omega^i$ we have $\mathcal{G}, \mu \models Strat$ iff $\mu \in Lang(\mathcal{A}_{Strat})$.*

Call a strategy *Strat* switch-free if it does not have any of the $\frown$ or the $+$ constructs.

Given a game arena $\mathcal{G}$ and strategy specifications of the players, we may ask whether there exists some subarena of $\mathcal{G}$ that the game settles down to if the players play according to their strategy specifications. (Note that a play being an infinite path in a finite graph, ind by settling down, we refer to the connected component that the play is eventually confined to.) This subarena is in some sense the equilibrium states of the game. It is also meaningful to ask if the game settles down to such an equilibrium subarena, then whether the strategy of a particular player attains stability with respect to switching.

Theorem 3. *Given a game arena $\mathcal{G} = (W, \rightarrow, w_0)$ a subarena R of $\mathcal{G}$ and strategy specifications $Strat_1, \ldots, Strat_n$ for players 1 to n, the following questions are decidable.*

- *Do all plays conforming to these specifications eventually settle down to R?*
- *Given strategy specifications $Strat_1, \ldots, Strat_n$ for players 1 to n, if all plays conforming to these specifications converge to R, does the strategy of player i become eventually stable with respect to switching?*

For a detailed study of strategy switching, see [Paul et al., 2009a].

2.7 Player types

For finite extensive-form games of perfect information, backward induction (BI) offers a solution that is simple and attractive as prediction of stable play. However, this critically depends on reasoning being backward, or bottom-up on the tree from the leaves to the root. In some games such as the famous example of the centipede game, this solution is somewhat counter-intuitive.

In general, an extensive-form game can have several Nash equilibria apart from the one given by the backward induction solution. If this is the statement we make *about* the game, how does the player reason *in* the game?

Surprise moves and forward induction: The following example is given by Perea [2010]: it is a two-player extensive-form game in which the first player chooses a move a that ends the game or the move b that leads to a normal

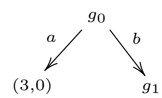


Fig. 9. An extensive-form game

	d_1	d_2	d_3
c_1	(2,2)	(2,1)	(0,0)
c_2	(1,1)	(1,2)	(4,0)

Fig. 10. Normal form game g_1

form game g_1, in which the players concurrently choose between $\{c_1, c_2\}$ and $\{d_1, d_2, d_3\}$, respectively.

The backward induction solution advises player 1 to choose a, so player 2 does not expect to have any role. But suppose player 1 chooses b and the game does reach g_1. How should player 2 reason at this node? Should player 2 conclude that 1 is irrational and choose arbitrarily, or should 2 treat the subgame as a new game *ab initio* expecting rational play in the future?

Note that player 2 can ascribe a good reason for 1 to choose b: the expectation that 2 would choose d_3 in game g_1. (In this case, 1 can be expected to play c_2, and then player 2's best response would be d_2.)

Such issues have been discussed extensively in the literature, and many resolutions have been suggested. Some of them go as follows:

– Players' actions are to be based on substantive stable common belief in future rationality [Baltag et al., 2009, Halpern, 2001].
– Treat the first move of player 1 as a mistake, and either ignore past information or update beliefs accordingly [Hoshi and Isaac, 2011].
– Players come in different *types*, and deviations from expected behaviour are interpreted according to players' knowledge of each others' type [van Benthem and Liu, 2004, van Benthem, 2009].
– Players rationalize each other's behaviour [Pearce, 1984].

Among these the last requires an explanation: according to this view, a player, at a game node, asks what rational strategy choices of the opponent could have led the history to this node. In such a situation, she must also ask whether the node could also have been reached by the opponent who does not only choose rationally herself, but who also believes that the other players choose rationally as well. This argument can be iterated, and leads to a form of forward induction [Battigalli and Siniscalchi, 1999, Perea, 2010]. This leads to an interesting algorithm that can be seen as an alternative to backward induction [Perea, 2012].

A small point is worth noting here: the way forward induction (FI) is formalized as above, both BI and FI yield the same outcome [Perea, 2010] in generic extensive-form games of perfect information (where payoffs at leaves are distinct). The strategies would in general be different, and this is in itself important for a theory of play. van Benthem [2014] suggests an alternative viewpoint: rather than looking for a normal form subgame as above, he suggests that any sufficiently abstract representation of the subgame may result in FI yielding a different outcome. For instance, if the players were computationally limited, they would have only a limited view of a large subgame, and this is a very relevant consideration for a theory of play.

In general, how a player reasons *in* the game involves not only reasoning such as the above, but also computational abilities of the player. As the game unfolds, players have to record their observations, and a memory-restricted player needs to select what to record [van Benthem, 2011, de Bruin, 2010].

Aumann and Dreze [2005] make a strong case for the focus of game theory to shift from equilibrium computation to questions of how rational players should play. For zero-sum games, the value of the game is unique and rational players will play to achieve this value. However, in the case of non-zero-sum games as mentioned above, multiple Nash equilibria can exist. This implies that players cannot extract an advice as to which strategy to employ from the equilibrium values. According to Aumann and Dreze, for a game to be well defined, it is also necessary that players have an expectation on what the other players will do. In estimating how the others will play, a rational player should take into account that others are estimating how he will play. The interactive element is crucial and a rational player should then play so as to maximize his utility, given how he thinks the others will play. The strategy specifications we introduce below are in the same spirit, since such a specification will be interactive in the sense of Aumann and Dreze [2005].

Players matter: van Benthem [2014] offers a masterly analysis of the many issues that distinguish reasoning about games and reasoning in games. Briefly, he points out that even if we consider BI as pre-game deliberation, there are aspects of dynamic belief revision to be considered; then there is the range of events that occur during play: players' observations, information received about other players, etc; then there is post-game reflection. As we move from deliberation to actual play, our interpretation of game theory requires considerable re-examination. We leave the reader to the pleasure of reading [van Benthem, 2014] for more on this, but pick up one slogan from there for discussion here: *the players matter.*

Briefly, reasoning inside games involves reasoning about actual play, and about the players involved. The standard game-theoretic approach uses uniform algorithms (such as BI and FI) to talk of reasoning during play (including the actuality of surprise moves) and *type spaces* encode all hypotheses that players have about each other. However, the latter is again of the pre-game deliberative kind (as in BI), and abstracts all considerations of actual play into the type space. It is in this spirit that Perea [2010] talks of completeness of type spaces for FI, whereas the van Benthem analysis is a (clarion) call for dynamics in both aspects: dynamic decision making during play and a theory of player types that's dynamically constructed as well.

We suggest that this is a critical issue for logical foundations of game theory. A node of a game tree is a history of play, and unless all players have a logical explanation of how play got there, it is hard to see them making rational decisions at that point. The rationale that players employ then critically depends on perceived continuity in other players' behaviour, which needs to be construed during the course of play.

However, while this is easily said, it raises many questions that do not seem to have obvious answers. What would be a logic in which such reasoning as proceeds during play can be expressed? What would we ask of such a logic – that it provides formulas for every possible strategy that a player might employ in every possible game? That it be expressively complete to describe the (bewildering) diversity of player types? That we may derive stable strategy profiles using an inference engine underlying the logic? That we discover new strategies from the axioms and inference rules of the logic?

Several logics have been studied in the context of reasoning about strategic ability. van Benthem [2012] studies strategies in a dynamic logic, and in the context of alternating temporal logics, a variety of approaches have been studied [van der Hoek et al., 2005, Walther et al., 2007, Jamroga and van der Hoek, 2004, Ågotnes and Walther, 2009]. While these logics reason with the functional notion of strategy, a theory of play requires reasoning about the dynamics of player types as well.

Logic and automata for player types: We suggest that the logical language attempt to describe a universe of constructible player types. Therefore, players in this framework are of definable type and considerations of other players are also restricted to definable types. Rationalizability becomes relative to the expressiveness of the underlying formalism; we can perhaps call this notion 'extensive-form *reasonability*'. We are less interested in completeness of the proposed language here, than in expressing interesting patterns of reasoning such as the ones alluded to above.

Our commitment is not only to simple modal logics to describe types, but also to realizing types by automata.[6] A number of reasons underlie this decision: for one, resource limitations of players critically affect course of play and selection of strategies. For another, automata present a nice tangible class of players that require rationale of the kind discussed above[7] and yet restrict the complexity that human players bring in. Further, automata theory highlights memory structure in players, and the selective process of observation and update.

[6] By automata, we refer only to finite state devices here, though probablistic polynomial time Turing machines are a natural class to consider as well [Fortnow and Whang, 1994].

[7] Suprise move by an opponent is perhaps much harder for an automaton to digest than for a human player.

Why is such an approach needed, or indeed relevant, considering that an elegant topological construction of type spaces is already provided by Battigalli and Siniscalchi [1999], Perea [2010] and others, with a completeness theorem as well? A crucial departure lies in the emphasis on constructivity and computability of types and strategies (rather than their existence). Moreover, if our attempt is not only to enrich the type space but also to provide *explanations* of types, logical means seem more attractive. The price to pay lies, of course, in the restrictive simplicity of the logics and automata employed, and it is very likely that such reasoning is much less expressive than the topological type spaces.

Types as formulas: Let N denote the set of players, we use i to range over this set. For technical convenience, we restrict our attention to two player games, i.e. we take $N = \{1, 2\}$. We often use the notation i and $\bar{\imath}$ to denote the players where $\bar{\imath} = 2$ when $i = 1$ and $\bar{\imath} = 1$ when $i = 2$. Let Σ be a finite set of action symbols representing moves of players, we let a, b range over Σ.

Strategizing during play involves making observations about moves, forming beliefs and revising them. Player types are constructed precisely in the same manner:

- Patterns of the form 'when condition p holds, player 2 chooses a' are observations by player 1 and help to assign a basic type to player 2.
- Such a process clearly involves nondeterminism to accommodate apparently contradictory behaviour, so a player needs to assign a disjunction of types to the other.
- The process of reasoning proceeds by case analysis: in situations such as x, the other player is seen to play conservatively whereas in other situations such as y, the type is apparently aggressive. Thus type construction is conjunctive as well.
- The planning of a player also includes how he responds to perceived opponent startegies that lie within this plan. Therefore type definition includes such responses.
- Rationalization: perceived behaviour can be explained by actual play being part of a strategy that involves the future as well, and this is articulated as a belief by the player about the opponent. Moreover, such beliefs include the opponent's beliefs about the player as well, and iterating the process builds a hierarchy of beliefs.

Above, we have spoken of the type of a player as it is ascribed by the opponent. Note that the same reasoning works for ascribing types 'from above' to a player. Such considerations lead us to a syntax for player types, which is again a two-level syntax as we had earlier: we have strategy specifications, and formulas from a simple action-indexed tense logic, enriched with a belief operator. In particular, we have formulas of the form $B_i \pi @ \bar{\imath}$ which is read as: i believes that the opponent is playing a strategy that conforms to π. The use of the @ symbol is to shift location to the opponent. The semantics of the belief operator is based on the rationalizability considerations discussed above. For details, please see **?**.

The construct $B_i\pi@\bar{\imath}$ describes (a kind of) *belief hierarchy*: player i believes that opponent behaviour corresponds to some complete plan π. Note that π, in turn could be referring to some type σ' of player i, and so on. In this sense, a player holds a belief about opponent's strategy choices, about opponents' beliefs about other agents' choices, opponents' beliefs about others' beliefs etc. Since this is essentially how type spaces are defined, these specifications offer a compositional means for structuring type spaces.

The semantics of a player type is given as a set of the player's plan subtrees of the given game tree, based on observables. It is defined at every player i node, specifying player i's beliefs about opponents' strategies that could have resulted in play reaching that node. But since every opponent's type specifies the opponent's beliefs about others' strategy choices, this results in a recursive structure and we can build a hierarchy of types.

Note that the belief assertions can specify different strategic choices based on the past, and thus talk of how a player may, during the game, revise her beliefs, a form of dynamics.

This further suggests that we wish to *derive* types during play. Thus, rather than types as being fixed for the class of games, we consider types as those start perhaps as heuristics, and *grow* during play.

Once we have the notion of types, it induces a notion of *local equilibrium* as follows. Consider player 1's response to player 2's strategy τ: here, 1's best response is not to τ, but to *every* type π that τ satisfies. Symmetrically, 2's best response is not to a straegy μ of player 1 but to every type σ that μ satisfies. Thus we can speak of the type pair (σ, π) being in equilibrium. We merely remark on this induced notion here, one well worth developing further on in future.

We have suggested that our definition of player types has been guided by concerns of constructibility and simplicity. Yet, we need to discuss how types as we have defined relate to the topological type spaces considered by game theorists, especially since forward induction is justified by the completeness of such spaces.

Let $\mathbb{T}(\Sigma) = (S, \Rightarrow, s_0)$ be an extensive-form game. A **type space** over $\mathbb{T}$ is a tuple $G = (U_i, \delta_i)_{i \in N}$ where each U_i is a compact topological space, representing the set of types for player i, and δ_i is a function that assigns to every type $u \in U_i$ and tree node s, a probability distribution $\delta_i(u, s) \in \Delta(\Omega_{\bar{\imath}}(s), U_{\bar{\imath}})$. Note that $\Omega_{\bar{\imath}}(s)$ represents the set of opponent strategies that potentially reach node s, $U_{\bar{\imath}} = \Pi_{j \neq i} U_j$ is the set of opponents' type combinations, and $\Delta(X)$ is the set of probability distributions on X with respect to the Borel σ-algebra.

In game theory, type spaces are typically defined for games of imperfect information, and the definition above coincides with the standard one when the information set for every player is a singleton. A natural question arises whether the concept makes sense for games of perfect information. In the discussions on forward induction, as for instance in [Battigalli and Siniscalchi, 1999] or [Perea, 2010], the BI and FI solutions coincide for generic games, and the analysis differentiates games with nontrivial information sets. However, as van Benthem [2014] argues, there are other interpretations of forward induction that are relevant for a theory of play: when the game tree is large, a player at a tree node s may be able to reason only about a small inital fragment of the subtree issued at s, and the subsequent abstraction may be seen as imperfect information as well.[8] Moreover, rationalizing by the player i does induce an equivalence relation $\sim_i$ on the tree nodes in our analysis.

Note the similarity of our definition of types to the standard notion, without the use of probability distributions. The use of equivalence relations between nodes is an implicit form of qualitative expectations and we choose the simpler formalism as it is more amenable to modal logics. With these observations, consider the type space 'induced' in our framework.

Consider a model $M = (T, V)$ where $(S, \Rightarrow, s_0, \widehat{\lambda})$ is an extensive form game tree and $V : S \to 2^P$ a valuation function. Then we define the **logical type space** over $\mathbb{T}$ to be a tuple $L = (Sat^i(T), \theta_i)_{i \in N}$, $Sat^i(T)$ is the set of player i type specifications satisfiable in the game, and $\theta_i : (Sat^i(T) \times S) \to Sat^{\bar{\imath}}(T)$ was defined earlier. Recall that this set represents the beliefs of player i about the opponent implied by the type σ at the node s.

Now one can see the close correspondence between the two definitions, as well as the differences. The type space G is globally defined, and can be seen as fixing an encoding of all possible beliefs of players about opponent behaviour *a priori*. In contrast, the type space L has more local structure, and is crucially determined by the expressiveness of the logic. The topological structure of the type space in G is replaced by logical structure in L. For instance, the types in L are down-ward closed: if $\sigma_1 \cdot \sigma_2$ is a type, then so are σ_1 and σ_2; it is closed under entailment: if σ_1 is a type and σ_1 entails σ_2, then σ_2 is a type, and so on. There are other symmetries such as: if $\pi \in \theta_i(\sigma, s)$ for some tree node s, then there exists a tree node t such that $\sigma \in \theta_{\bar{\imath}}(\pi, t)$. Characterizing this logical structure by a completeness theorem is an important question, but we do not proceed further on this here.

At this juncture, our claim to 'growing' types can be explained. Consider the root node s_0 in the tree T. Notice that the beliefs of the players about each other refer only to invariant properties in the game tree (as specified by the observables), and hence the only definite assertions are about the present, namely the root node itself. However, as play progresses, we have definite assertions about the past, as well as about the choices thus eliminated, and we have sharper type formulas. This process may be understood as a construction of the type space that proceeds top-down, starting from the root node and enriching players' beliefs based on observations as the game tree gets pruned by play.[9]

While this is a general picture, we focus on a specific question: does this 'construction' of a logical type require an *unbounded* amount of information? We now proceed to show that the required information is in fact *finite state*, and hence can be checked by an automaton. Further, we show that, in principle, a Turing machine can construct the type space.

[8] We refer the reader to [van Benthem, 2014] for a more detailed justification.

[9] A formal characterization of this process as a recursive function on the tree is in progress, but there are many technical challenges.

Once we consider types to be logical, a natural question is whether a given type is consistent: we want a player to be a reasoner whose reasoning is coherent. It is this question we address here.

Note that a type corresponds to a set of plans and beliefs in our framework. We have spoken of a model in which a player records observations during play and rationalizes opponents' behaviour by considering what strategies might have led to opponents playing in a particular way. The meaning we offer for constructibility of such a type is a finite state automaton that 'plays out' such plans and rationalizes course of play.

2.8 Large games

The main strand running through the discussion so far is that we have considered temporally large games (that have episodic structure). We now consider spatially large games (where the number of players is too large for rationality to be based on exhaustive intersubjectivity).

Issues in games with a large number of players Game models of social situations typically involve large populations of players. However, common knowledge of rationality symmetrizes player behaviour and allows us to predict behaviour of any rational player. On the other hand, it is virtually impossible for each player to reason about the behaviour of every other player in such games, since a player may not even know how many players are in the game, let alone how they are likely to play.

What is the technical implication of the number of players being large? In our view, in large games, the payoffs are usually dependent on the 'distribution' of the actions played by the players rather than the action profiles themselves. Moreover, in such games the payoffs are independent of the identities of the players.

An action distribution is a tuple $\mathbf{y} = (y_1, y_2, \ldots, y_{|A|})$ such that $y_i \geq 0,\ \forall i$ and $\sum_{i=1}^{|A|} y_i \leq n$. Let $\mathbf{Y}$ be the set of all action distributions. Given an action profile $\mathbf{a}$, we let $\mathbf{y}(\mathbf{a})$ be its corresponding action distribution, that is, $\mathbf{y}(\mathbf{a})(k)$ gives the number of players playing the kth action in A. Every player i has a rational valued function $f_i : \mathbf{Y} \to \mathbb{Q}$ which can be seen as the payoff of i for a particular distribution.

Why are such distributions interesting? They are used in many social situations. For instance, recently Singapore decided to make the entire city Wi-Fi enabled. How is it decided that a facility be provided as infrastructure? Typically such analysis involves determining when usage crosses a threshold. But then understanding why usage of one facility increases vastly, rather than another, despite the presence of several alternatives, is tricky. But this is what strategy selection is about. However, we are not as much bothered about strategy selection by an individual player but by a significant fraction of the population.

Similar situations occur in the management of the Internet. Policies for bandwidth allocation are not static. They are dynamic, based on studying both volumes of traffic and type of traffic. The popularity of an application like YouTube dramatically changes such usage, calling for changes in Internet policies. Predicting such future requirements is tricky, but much wanted by the engineers. Herd mentality and imitation are common in such situations.

In large games, payoffs are associated not with strategy profiles, but with type distributions. Suppose there are k strategies used in the population. Then the outcome is specified as a map $\mu : \Pi_k(n) \rightarrow P^k$, where $\Pi_k(n)$ is a set of distributions: k-tuples that sum up to n, and P is a payoff. Thus every player playing the j^{th} strategy gets the payoff given by the j^{th} component specified by μ for a given distribution. Typically there is usually a small number t of types such that $t < n$ where n is the number of players. Can one carry out all the analysis using only the t types and then lift the results to the entire game?

Why should such an analysis be possible? When we confine our attention to finite memory players, for n players, the strategy space is the n-fold product of these memory states. What we wish to do is to map this space into a t-fold product, whereby we wish to identify two players of the same type. We can show that in the case of deterministic transducers, such a blow-up is avoidable, since the product of a type with itself is then isomorphic to the type.

A population of 1000 players with only two types needs to be represented only by pairs of states and not 1000-tuples. But we need to determinize transducers, and that leads to exponential blow-up. So one might ask, when is the determinising procedure worthwhile? Suppose we have n players, t types, and p is the maximum size of the state space of any nondeterministic type finite state transducer. It turns out [Paul and Ramanujam, 2011a] that the construction is worthwhile when $n > 0.693 \cdot t \cdot \pi(p)$, where $\pi(p)$ is the number of primes below p. As we are talking about large games, the inequality above can be expected to hold.

In general, while we have spoken only of qualitative outcomes, and this is natural for a logical study, it makes sense to consider **quantitative** objectives as well, especially when outcomes are distribution determined. For infinite play, such outcomes may diverge, and we then need to consider *limit-average* payoffs (or other discounted payoffs). It is still possible to carry out the kind of analysis as we have discussed here, to show existence of equilibria in finite memory strategies, as for instance, in [Paul and Ramanujam, 2011a].

Neighbourhood structures: In large games, it is convenient to think of players arranged in neighbourhoods. A player strategizes locally, observing behaviour and outcomes within her neighbourhood, but may switch to an adjacent neighbourhood.

As an example, consider vegetable sellers in India. In Indian towns, it is still possible to see vegetable sellers who carry vegetables in baskets or pushcarts and set up shop in some neighbourhood. The location of their 'shop' changes dynamically, based on the seller's perception of demand for vegetables in different neighbourhoods in the town, but also on who else is setting up shop near her, and on her perception of how well these (or other) sellers are doing. Indeed, when she buys a lot of vegetables in the wholesale market, the choice of her 'product mix' as well as her choice of location are determined by a complex rationale. While the prices she quotes do vary depending on the general market situation, the neighbourhoods where she sells also influence the prices significantly: she knows that in the poorer neighbourhoods, her buyers cannot afford to pay much. She can be thought of as a small player in a large game, one who is affected to some extent by play in the entire game, but whose strategising is local where such locality is itself dynamic.

In the same town, there are other, relatively better off vegetable sellers who have fixed shops. Their prices and product range are determined largely by the wholesale market situation, and relatively unaffected by the presence of the itinerant vegetable sellers. If at all, they see themselves in competition only against other fixed-shop sellers. They can be seen as big players in a large game.

What is interesting in this scenario is the movement of a large number of itinerant vegetable sellers across the town, and the resultant increase and decrease in availability of specific vegetables as well as their prices. We can see the vegetable market as composed of dynamic neighbourhoods that expand and contract, and the dynamics of such a structure dictates, and is in turn dictated by the strategies of itinerant players.

When we model games with such neighbourhood structures, the central question to study is that of stability of game configurations. When can we guarantee that game dynamics leads to a configuration that does not change from then on, or oscillated between fixed configurations? Do finite memory strategies suffice? What kind of game-theoretic tools are used for such analysis? Paul and Ramanujam [2011b] offer an instance, where a characterization is presented in terms of *potential games* [Monderer and Shapley, 1996]. However, the general question of what stable configurations are of interest and how to strategize to achieve them is of general interest, as well as obtaining bounds on when stability is attained.

Dynamic game forms: Social situations often involve strategies that are generic, (almost) game-independent: threat and punishment; go with the winner / follow the leader; try to take the lead, and if you can't, follow a leader; imitate someone you think well of; and so on. They have some (limited) efficacy in many interaction situations. But when a significant proportion of players use such heuristics, it may affect game dynamics significantly.

In general, we can consider such dynamics as follows. An individual player has to make choices; making choices has a cost. Society provides choices, incurs cost to do so. Society revises choices and costs from time to time based on the history and prediction of the future. This affects individual strategies who switch between the available choices. Then the game arena is not static but changes dynamically.

We can then ask several questions based on eventual patterns dictated by the dynamics:

– Does the play finally settle down to some subset of the game?
– Can a player ensure certain objectives using a strategy that doesn't involve switching?

– Given a subarena, is a particular strategy *live*?
– Does an action profile eventually become part of the social infrastructure?
– Do the rules of the society and the behaviour of other players drive a particular player out of the game?

Paul et al. [2009b] offer a formal model in which such questions are posed and it is shown that these can be checked algorithmically. Therefore, it is possible to compare between game restriction rules in terms of their imposed social cost. For a player, if the game restriction rules are known and the type of the other players are known then she can compare between her strategy specifications.

The more general objective of such study is to explore the rationale of when and how should society intervene, and when such rationale is common knowledge among players, how they should strategize. In this sense, individual rationality and societal rationality are mutually recursive in each other, and the study of such interdependence offers an interesting challenge for logical models.

The imitation heuristic: In a large population of players, where resources and computational abilities are asymmetrically distributed, it is natural to consider a population where the players are predominantly of two kinds: optimisers and imitators. Asymmetry in resources and abilities can then lead to different types of imitation and thus ensure that we do not end up with "herd behaviour". Mutual reasoning and strategising process between optimizers and imitators leads to interesting questions for game dynamics in these contexts.

Is imitation justified? We can say no since it does not achieve optimal in most cases. But we can also say yes, since it saves time, uses less resource and does not do much worse than optimal outcomes in most cases.

The rationality (or otherwise) of imitation has been studied (though perhaps not extensively) in game theory. In [Paul and Ramanujam, 2010], games of unbounded duration on finite graphs are studied, where players may have overlapping objectives, and are divided into players who optimise and others who imitate. In this setting, it is shown that the following questions can be answered algorithmically:

– If the optimisers and the imitators play according to certain specifications, is a global outcome eventually attained?
– What sort of imitative behaviour (subtypes) eventually survive in the game?
– How worse-off are the imitators from an equilibrium outcome?

However, this is a preliminary study, and more sophisticated models would involve randomizing players as well as more nuanced player types in the population.

3 Interaction about preferences

From modelling interactions per se, in form of games and strategies, we now move on to study *preferences* which constitute a motive or incitement for interaction.

Preferences are integral to any decision-making process, be it individual or collective. In general, an agent is said to prefer some option a over another option b if a is more desirable, advantageous, beneficial or choice-worthy than b for the agent. Thus, a motivational attitude like preference in decision-making scenarios is basically a comparative attitude. It can also be considered as an appraisal of matters of value in such situations leading to certain choices, and as such it is different from the more informational attitudes like knowledge which concern facts. These preferences are also subjective in nature as they are attributed to agents under consideration.

In his seminal work [von Wright, 1963] on preferences, von Wright distinguished between two kinds of preferences: extrinsic and intrinsic. The kind of preferences discussed above constitute extrinsic preferences - in his words, 'a judgement of betterness serves as a ground or reason for preference'. On the other hand one might simply like one choice over another for no reason whatsoever, e.g., one might prefer tea over coffee simply because one likes tea more. An extensive amount of work has been done on analyzing both extrinsic and intrinsic preferences from the logic viewpoint. For this chapter, the focus is on the extrinsic preferences to provide an overview of the work done towards situating and modelling certain approaches on (combining) preferences in the collective decision-making scenarios.

The growing importance of decision-making in AI has resulted in a significant increase in focus on representation of preferences and reasoning about preferences, where logic plays an important role. From the computational viewpoint, algorithmic and complexity considerations in various social procedures, e.g., voting [Lang, 2004, Conitzer et al., 2007], fair allocation [Chevaleyre et al., 2006], and others have led to the development of computational social choice [Chevaleyre et al., 2008], a crucial area in current day research. Another relevant issue is automated learning of preferences, where the approaches for preference elicitation are quite varied, e.g., range from asking effective questions [Haddawy et al., 2003] to analyzing others' preferences [Pazzani, 1999].

When we deal with preferences in decision-making scenarios, more often than not we consider the following questions: What are the reasons behind such preferences? Can these preferences be changed/modified in some way? For a detailed logical study that deals with such questions, see [Liu, 2011]. Often, interaction/deliberation between agents influences their respective preferences - the individual decision-making processes of the agents' affect one another's viewpoints towards collective decision-making and in such cases we need ways to consider the individual changes in preferences towards attaining some *uniform* preferences in the group. Even without interaction, one can combine the individual preferences based on some specific rules of combination, e.g., majority rules, and we get the group preference. In what follows we provide a comparative study of some of the logical approaches which are built on similar frameworks. As discussed in the introduction, we focus our study on *preference aggregation* as well as *deliberation about preferences*.

As a running example for this section, let us consider the following paradigmatic situation: Three friends Aniket, Brishti and Chirayu would like to go to a restaurant together and they have shortlisted three restaurants. Out of these three, they all have their individual choices. Now, the question is how to come up with a single choice so that all of them can go together. We now provide a logical framework to ground our current discussion. Since the focus is on multi-agent preferences, we consider A to be a *finite non-empty* set of agents, where $|\mathsf{A}| = n \geq 2$. We first define simple preference structures as follows:

Definition 1 (Preference Frame). *A preference frame F is a tuple $(W, \{\leq_i\}_{i\in\mathsf{A}})$ where* **(1)** *W is a finite non-empty set of worlds;* **(2)** *for all agents i, $\leq_i \subseteq W \times W$ is a preorder (i.e., a reflexive and transitive relation), agent i's* preference relation *among worlds in W ($u \leq_i v$ is read as "world v is at least as preferable as world u for agent i"). We define $u <_i v$ ("u is less preferred than v for agent i") as ($u \leq_i v$ and $v \not\leq_i u$).*

A basic modal language to talk about these preferences is given below. Let P be a countably infinite set of propositional variables. Our intuitive explanations of the syntactic vocabulary follow the definition of formulas.

Definition 2 (Language $\mathcal{L}_p$). Formulas φ *of the language $\mathcal{L}_p$ are given by*

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid \langle\leq_i\rangle\varphi \mid \langle<_i\rangle\varphi \mid E\varphi$$

where, $p \in \mathsf{P}$. Propositional constants $(\top, \bot)$, other Boolean connectives $(\wedge, \rightarrow, \leftrightarrow)$ are defined as usual and the dual modal universal operator $[\leq_i]$ is defined as $[\leq_i]\varphi := \neg\langle\leq_i\rangle\neg\varphi$. The duals $[<_i]\varphi$ of $\langle<_i\rangle\varphi$ and $U\varphi$ of $E\varphi$ are defined analogously.

The intuitive reading of the formula $\langle\leq_i\rangle\varphi$ is that 'there is a preferred world for agent i where φ holds', and similar is that for the formula $\langle<_i\rangle\varphi$ with respect to strict preference. The formula $E\varphi$ describes the existence of a world where φ holds. To provide interpretation of these formulas formally we now define preference models based on the preference frames above.

Definition 3 (Preference Model). *A preference model M is a tuple $(W, \{\leq_i\}_{i\in\mathsf{A}}, V)$, where $(W, \{\leq_i\}_{i\in\mathsf{A}})$ is a preference frame and $V : \mathsf{P} \rightarrow 2^W$ is a valuation function.*

Definition 4 (Truth definition). *The interpretation of the formulas of $\mathcal{L}_p$ are given in terms of pointed preference models (M, w) with $w \in W$:*

$$\begin{array}{rll}
M, w \models p & \textit{iff} & w \in V(p), \\
M, w \models \neg\varphi & \textit{iff} & M, w \not\models \varphi, \\
M, w \models \varphi \vee \psi & \textit{iff} & M, w \models \varphi \textit{ or } M, w \models \psi, \\
M, w \models \langle\leq_i\rangle\varphi & \textit{iff} & \exists v \textit{ s.t. } w \leq_i v \textit{ and } M, v \models \varphi, \\
M, w \models \langle<_i\rangle\varphi & \textit{iff} & \exists v \textit{ s.t. } w <_i v \textit{ and } M, v \models \varphi, \\
M, w \models E\varphi & \textit{iff} & \exists v \textit{ s.t. } M, v \models \varphi.
\end{array}$$

Definition 5 (Satisfiability and validity). *A formula φ is said to be satisfiable in a model M if there is a state w such that $M, w \models \varphi$ and valid if it is true at every state in every model.*

We note here that when we talk about preferences, we generally have a pair of entities in mind, and we consider preference of one over the other. We follow [van Benthem et al., 2009, Girard, 2011] to illustrate some such notions in terms of binary preference operators $\leq_i^{\exists\exists}$ and $\leq_i^{\forall\exists}$:

- $M, w \models \varphi \leq_i^{\exists\exists} \psi$ iff $\exists s, t : M, s \models \varphi$ and $M, t \models \psi$ and $s \leq_i t$.
- $M, w \models \varphi \leq_i^{\forall\exists} \psi$ iff $\forall s, \exists t : M, s \models \varphi$ implies $M, t \models \psi$ and $s \leq_i t$.

The formula $\varphi \leq_i^{\exists\exists} \psi$ describes the existences of a φ-world and a ψ-world such that agent i prefers the ψ-world more than the φ-world. The formula $\varphi \leq_i^{\forall\exists} \psi$ describes that for each φ-world there exists a ψ-world such that agent i prefers the ψ-world more than the φ-world.These operators can be expressed in $\mathcal{L}_p$ in the following way.

$$\varphi \leq_i^{\exists\exists} \psi := E(\varphi \wedge \langle \leq_i \rangle \psi)$$
$$\varphi \leq_i^{\forall\exists} \psi := U(\varphi \rightarrow \langle \leq_i \rangle \psi)$$

Strict binary preference operators can also be defined similarly. For a detailed discussion on such binary preferences and a complete axiomatization of the logic described above, see [van Benthem et al., 2009]. We now move on to main topic of this section, that is, combining individual preferences to provide group preferences. As mentioned earlier, there are two aspects of any such collective decision making processes: aggregation and deliberation.

Social choice theory [Arrow, 1963, Arrow et al., 2002] which deals with studies on preference aggregation in terms of social choice functions is known for its impossibility results. On the other hand, deliberative democracy, which can be considered as a process of open discussion leading to an agreed judgement on different decisions [Elster, 1986, Habermas, 1996], may face fewer difficulties from such issues. However, social choice theorists feel that such issues devastate deliberation as the deliberative democrats prescribe procedures whose structurelessness is more conducive to impossibility in collective choice [Miller, 1992, Knight and Johnson, 1994]. But, according to the deliberative democrats, private desires or even aspirations may not be well aggregated by some 'voting rule', whereas deliberation can lead to certain preference uniformities, e.g., single-peaked preference profiles [Black, 1948] - the agents might agree about the dimension over which they disagree. Without going into this debate, we describe below some logical approaches dealing with the processes of aggregation and deliberation. Our goal is to consider amalgamation of these approaches towards a more effective analysis of collective decision making.

3.1 Expressing aggregation of preferences

Various authors have proposed logics describing aggregation of preferences (e.g., see [Agotnes et al., 2011, Troquard et al., 2011, Girard, 2011, Ghosh and Velázquez-Quesada, 2011]) dealing with various aspects of the aggregation process. We introduce a logic specifically designed to support reasoning about social choice functions. Agotnes et al. [2011] studied judgement aggregation from a logical perspective by proposing a modal logic to reason judgement aggregation scenarios. They considered how multiple sets of logical formulas can be aggregated to a single consistent set. Preference aggregation is considered as a special case of judgement aggregations. Troquard et al. [2011] proposed a logic to reason about social choice functions by considering modal operators to capture agent preferences and strategic abilities of coalitions of agents. A correspondence was established between formulas in the logic and properties of social choice functions. Girard [2011] proposed a (hybrid) group preference logic (*GPL*) to express preference aggregation in terms of a given hierarchy of agents and added a dynamic operator to *GPL* to model effects of a particular change in the hierarchy, namely, agent promotion - an agent in the group under consideration is promoted to a higher rank. Ghosh and Velázquez-Quesada [2011] proposed logics of opinions, beliefs and preferences of agents describing the mutual influences of the agents over themselves as well as the events involved, together with their effect on beliefs and preferences. In what follows we describe one of them, [Girard, 2011] in more details. The said proposal is close in terms of the frameworks on deliberation we discuss below.

Before going any further, we first describe the lexicographic aggregation operators defined in [Andréka et al., 2002] based on which the modal preference operators are proposed in *GPL*. Let V denote a set of variables and let $(\leq_x)_{x \in V}$ denote a collection of binary (preference) relations on some non-empty set, W, say. The task is to combine this collection of preference relations in some way to form a single preference relation. These variables may correspond to agents and the combined relation may provide us with the group preference relation. The concepts that are used to define this combined relation are priority graphs and priority relations, defined below.

Definition 6 (Priority graph). *A priority graph P is a tuple $(\mathsf{A}, <, v)$ where* **(1)** A *is a finite non-empty set of nodes (agents);* **(2)** $<$ *is a strict partial order on* A *(the priority relation), and;* **(3)** v *is a function from* A *to* V*. (Note that, more than one agent can be assigned to a single variable).*

Definition 7 (Priority relation). *A priority relation $\leq_P$ corresponding to a priority graph $P : (\mathsf{A}, <, v)$ is a binary relation on W providing an aggregated preference relation given as follows:*

$$\textit{for all } s, t \in W,\ s \leq_P t \textit{ iff } \forall i \in \mathsf{A}(s \leq_{v(i)} t \vee \exists j \in \mathsf{A}(i < j \wedge s <_{v(j)} t)$$

The definition above of the priority relation is based on the usual lexicographic rule. We note that the priority relation defined here is expressed in terms of a priority operator in [Andréka et al., 2002]. There are certain compositions of these priority operators which are of significant importance, as discussed in [Andréka et al., 2002], in the sense that, every priority operator (relation), defined with respect to a priority graph, is equivalent to one built from these fundamental operators (relations). These compositions are defined as follows:

Definition 8 (Composition of Priority operators). *Given any two priority graphs, P and Q two operators P/Q and $P||Q$, which are called the 'but' and 'on the other hand' operators, respectively, are defined as follows:*

$$P/Q = (P \cap Q) \cup P^{<}$$
$$P||Q = P \cap Q.$$

Here, $P^{<}$ denotes the strict binary relation corresponding to the priority graph P. The *but* operator puts the (priority relation corresponding to the) first graph in priority over (that of) the second (that is, apply P whenever there is conflict between P and Q), and the *on the other hand operator* puts the (priority relations corresponding to the) graphs at an incomparable rank. We are now all set to define *GPL* as proposed in [Girard, 2011]. In addition to a set of propositional variables P, a set nominals N is also considered in the language. The hybrid language given below helps in providing syntactic characterizations of the properties of the group preference relations defined in terms of priority graphs.

Definition 9 (Language $\mathcal{L}_{gp}$). Formulas φ *of the language $\mathcal{L}_g p$ are given by*

$$\varphi ::= p \mid s \mid \neg\varphi \mid \varphi \vee \psi \mid \langle \leq_X \rangle \varphi \mid \langle <_X \rangle \varphi \mid E\varphi$$
$$X ::= i \mid X/Y \mid X||Y$$

where, $p \in \mathsf{P}$ and $s \in \mathsf{N}$. Propositional constants $(\top, \bot)$, other Boolean connectives $(\wedge, \rightarrow, \leftrightarrow)$ are defined as usual and the dual modal universal operator $[\leq_X]$ is defined as $[\leq_X]\varphi := \neg\langle \leq_X \rangle \neg\varphi$. The duals $[<_X]\varphi$ of $\langle <_X \rangle \varphi$ and $U\varphi$ of $E\varphi$ are defined analogously.

The intuitive reading of the formula $\langle \leq_X \rangle \varphi$ is that 'there is a preferred world for a group of agents whose priority graph is given by X where φ holds', and similar is that for the formula $\langle <_X \rangle \varphi$ with respect to strict preference. When X denotes a single agent, the reading is as earlier for the basic language. The formula $E\varphi$ describes the existence of a world where φ holds. To provide interpretation of these formulas we now define group preference models based on the priority graphs given above.

Definition 10 (Group Preference Model). *A group preference model M is a tuple $(W, G, \{\leq_X\}_{X \in G}, V)$, where* **(1)** *$W$ is a non-empty set of worlds;* **(2)** *G is a set of priority graphs based on the agents in A;* **(3)** *$\{\leq_X\}_{X \in G}$ is a collection of priority relations corresponding to the priority graphs in G, and;* **(4)** *$V : \mathsf{P} \cup \mathsf{N} \rightarrow 2^W$ is a valuation function such that $V(s)$ is a singleton for all $s \in \mathsf{N}$.*

Definition 11 (Truth definition). *The interpretation of the formulas of $\mathcal{L}_{gp}$ are given in terms of pointed preference models (M, w):*

$$
\begin{array}{rcl}
M, w \models p & \textit{iff} & w \in V(p), \\
M, w \models s & \textit{iff} & \{w\} = V(s), \\
M, w \models \neg\varphi & \textit{iff} & M, w \not\models \varphi, \\
M, w \models \varphi \vee \psi & \textit{iff} & M, w \models \varphi \textit{ or } M, w \models \psi, \\
M, w \models \langle \leq_X \rangle \varphi & \textit{iff} & \exists v \textit{ s.t. } w \leq_X v \textit{ and } M, v \models \varphi, \\
M, w \models \langle <_X \rangle \varphi & \textit{iff} & \exists v \textit{ s.t. } w <_X v \textit{ and } M, v \models \varphi, \\
M, w \models E\varphi & \textit{iff} & \exists v \textit{ s.t. } M, v \models \varphi.
\end{array}
$$

For more details on *GPL* together a sound and complete axiomatization, and dynamic extensions of the logic, see [Girard, 2011]. Our focus is on reasoning based on aggregation of preferences and the static logic presented here serves the purpose. Let us consider the running example: Three friends Aniket, Brishti and Chirayu would like to go to a restaurant together and they have shortlisted three restaurants, say r_1, r_2 and r_3. Our agent set is $\mathsf{A} = \{a, b, c\}$. Let the propositions p_i denote the fact that r_i is the best restaurant, for $i \in \{1, 2, 3\}$. To model this scenario we consider the following set of worlds: $W = \{w_1, w_2, w_3\}$. Let the valuations be given by, $V(p_i) = \{w_i\}$ for each i. Now, out of these three restaurants, the three friends have their individual choices. To keep it simple let us consider a best choice for each of them: For agent a: $w_3 < w_1; w_2 < w_1$, for agent b: $w_3 < w_2; w_1 < w_2$, for agent c: $w_1 < 3; w_2 < w_3$. If we consider the priority graph given by $a/b/c$, the group preference would give us w_1, i.e. r_1 to be the most preferred restaurant. Again, if we consider the priority graph given by $(a||b)/c$, there will not be any best choice whatsoever.

These preference aggregation operators in *GPL* are based on some hierarchy of agents which is basically ad hoc in nature. Some graphs (hierarchies) will give us a unanimous choice, but some would not. In such kind of a treatment, one does not question where these hierarchies come from, but just accept them as given. Such a process might be beneficial when the number of agents is too high (e.g. in policy-making decisions), and it is simply not possible to bring them on the same table so that they can be influenced by one another to change their preferences so as to come to an agreement. But, for a relatively small number of agents (e.g., in faculty-selection committees) a deliberation process would go a long way towards achieving the purpose.

3.2 Expressing deliberation on preferences

While the process of aggregation focuses on accumulating individual preferences without discussing their origin [Dietrich and List, 2013], deliberation can be seen as a conversation through which individuals justify their preferences, a process that might lead to changes in their opinions as they are influenced by one another. As discussed earlier, even if deliberation may not always lead to unanimity, the discussion can lead to some 'preference uniformity' which might facilitate their eventual aggregation. There have been logic-based proposals on deliberation about preferences (e.g., see [Goldbach, 2015, Ghosh and Velázquez-Quesada, 2015a,b, Velázquez-Quesada, 2017, Ghosh and Sano, 2017]), and we focus on a line of study followed in most of them, where an agent's preferences are modified based on some other agents' preferences whom they consider reliable. We first provide some preliminary definitions.

Definition 12 **(*PR* Frame).** *A PR* (preference/reliability) frame *F is a tuple* $(W, \{\leq_i, \preccurlyeq_i\}_{i \in \mathsf{A}})$ *where* **(1)** *W is a finite non-empty set of worlds;* **(2)** $\leq_i \subseteq W \times W$ *is a preorder (i.e., a reflexive and transitive relation), agent i's* preference relation *among worlds in W ($u \leq_i v$ is read as "world v is at least as preferable as world u for agent i");* **(3)** $\preccurlyeq_i \subseteq \mathsf{A} \times \mathsf{A}$ *is a pre-order, agent i's* reliability relation *among agents in A ($j \preccurlyeq_i k$ is read as "agent k is at least as reliable as agent j for agent i").*

We define the following abbreviations.

- $u <_i v$ ("u is less preferred than v for agent i") means $u \leq_i v$ and $v \not\leq_i u$,
- $u \simeq_i v$ ("u and v are equally preferred for agent i") means $u \leq_i v$ and $v \leq_i u$,
- $j \prec_i k$ ("j is less reliable than k for agent i") is defined as $j \preccurlyeq_i k$ and $k \not\preccurlyeq_i j$,
- $j \approx_i k$ ("j and k are equally reliable for agent i") means $j \preccurlyeq_i k$ and $k \preccurlyeq_i j$.

To describe agent preferences based on their reliabilities we consider the following language. In addition to the usual preference operator for $\leq$, there is a converse operator, $\geq$, and also Boolean combinations of the operators. This facilitates in expressing the different relations described above and also in giving interpretation to the dynamic operators introduced later.

Definition 13 **(Language $\mathcal{L}_{pr}$).** *Formulas φ, ψ and relational expressions π, σ of the language $\mathcal{L}_{pr}$ are given by*

$$\varphi := p \mid j \sqsubseteq_i j' \mid \neg\varphi \mid \varphi \vee \psi \mid \langle\pi\rangle\,\varphi$$
$$\pi := U \mid \leq_i \mid \geq_i \mid ?(\varphi, \psi) \mid -\pi \mid \pi \cup \sigma \mid \pi \cap \sigma$$

with $p \in \mathsf{P}$ and $i, j, j' \in \mathsf{A}$. Propositional constants ($\top, \bot$), other Boolean connectives ($\wedge, \rightarrow, \leftrightarrow$) and the dual modal universal operators $[\pi]$ are defined as usual ($[\pi]\,\varphi := \neg\langle\pi\rangle\,\neg\varphi$ for the latter).

The set of formulas of $\mathcal{L}_{pr}$ contains atomic propositions (p) and formulas describing the agents' reliability relations ($j \sqsubseteq_i j'$), and it is closed under negation ($\neg$), disjunction ($\vee$) and modal operators of the form $\langle\pi\rangle$ with π a relational expression. The set of relational expressions contains the constant U (the global relation), the preference relations ($\leq_i$), their respective converse ($\geq_i$; []) and an additional construct of the form $?(\varphi, \psi)$ (explained after the semantic interpretation) where φ and ψ are formulas of the language, and it is closed under Boolean operations over relations (the so called *boolean modal logic*; []). To interpret these formulas, *PR* models are defined below.

Definition 14 **(*PR* model).** *A PR* model *M is a tuple (F, V) where F is a PR frame and $V : \mathsf{P} \rightarrow 2^W$ is a* valuation function*. A pair (M, w) with M a PR model and w a world in it is called a* pointed *PR* model*.*

Definition 15 (Truth definition). *The truth definition of formulas in $\mathcal{L}_{pr}$ at pointed PR models and the relations R_π for the relational expressions π are given by*

$$\begin{array}{lll} M, w \models p & \textit{iff} & w \in V(p) \\ M, w \models j \sqsubseteq_i j' & \textit{iff} & j \preccurlyeq_i j' \\ M, w \models \neg\varphi & \textit{iff} & M, w \not\models \varphi \\ M, w \models \varphi \vee \psi & \textit{iff} & M, w \models \varphi \quad \textit{or} \quad (M, w) \models \psi \\ M, w \models \langle \pi \rangle \varphi & \textit{iff} & \exists u \in W \ \textit{s.t.}\ R_\pi wu \ \textit{and}\ M, u \models \varphi \end{array}$$

and

$$\begin{array}{llll} R_U & := W \times W & R_{-\pi} & := (W \times W) \setminus R_\pi \\ R_{\leq_i} & := \leq_i & R_{\pi\cup\sigma} & := R_\pi \cup R_\sigma \\ R_{\geq_i} & := \{(v, u) \mid u \leq_i v\} & R_{\pi\cap\sigma} & := R_\pi \cap R_\sigma \\ R_{?(\varphi,\psi)} & := \{(u, v) \mid M, u \models \varphi \ \textit{and}\ (M, v) \models \psi\} & & \end{array}$$

We note that $R_{?(\varphi,\psi)}$ is the set of those pairs $(u, v) \in (W \times W)$ such that u satisfies φ and v satisfies ψ.

For a complete axiomatization of the logic described above see [Ghosh and Velázquez-Quesada, 2015a]. Deliberation is essentially a dynamic process and below we model the effects of such a process leading to preference changes (upgrades). We follow [Ghosh and Velázquez-Quesada, 2015a,b]. in defining relevant preference upgrade operation. For the sake of brevity we assume totality of the reliability orderings. Some proposals of such changes based on less-restrictive reliability orderings can be found at [Velázquez-Quesada, 2017, Ghosh and Sano, 2017].

Deliberation is essentially a dynamic process and the effect of deliberation about agents' preferences is modelled in [] in the following manner: A public announcement of the agents' individual preferences might induce an agent i to adjust her own preferences according to what has been announced and the reliability she assigns to the set of agents. Thus, agent i's preference ordering *after* such announcement, $\leq'_i$, can be defined in terms of the just announced preferences (the agents' preferences *before* the announcement, $\leq_1, \ldots, \leq_n$) and how much i relied on each agent (i's reliability *before* the announcement, $\preccurlyeq_i$): $\leq'_i := f(\leq_1, \ldots, \leq_n, \preccurlyeq_i)$ for some function f. Below, we define a preference upgrade operation based on agent reliabilities from [Ghosh and Velázquez-Quesada, 2015a] which is a general way of describing some natural (lexicographic) upgrade operators. In what follows, we consider the reliability orderings to be total for the sake of simplicity, and assume that each agent has a unique maximally reliable agent. Similar studies based on non-total orderings can be found in [Velázquez-Quesada, 2017].

Definition 16 (General lexicographic upgrade). *A* lexicographic list $\mathcal{R}$ *over W is a finite non-empty list whose elements are indices of preference orderings over W, with $|\mathcal{R}|$ the list's length and $\mathcal{R}[k]$ its kth element ($1 \leq k \leq |\mathcal{R}|$). Intuitively, $\mathcal{R}$ is a priority list of preference orderings, with $\leq_{\mathcal{R}[1]}$ the one with* *the highest priority. Given $\mathcal{R}$, the preference ordering $\leq_{\mathcal{R}} \subseteq (W \times W)$ is defined as*

$$u \leq_{\mathcal{R}} v \text{ iff}_{def} \underbrace{\Big(u \leq_{\mathcal{R}[\,|\mathcal{R}|\,]} v \;\wedge\; \bigwedge_{k=1}^{|\mathcal{R}|-1} u \simeq_{\mathcal{R}[k]} v\Big)}_{1} \;\vee$$
$$\underbrace{\bigvee_{k=1}^{|\mathcal{R}|-1} \Big(u <_{\mathcal{R}[k]} v \;\wedge\; \bigwedge_{l=1}^{k-1} u \simeq_{\mathcal{R}[l]} v\Big)}_{2}$$

Thus, $u \leq_{\mathcal{R}} v$ holds if this agrees with the least prioritised ordering ($\leq_{\mathcal{R}[|\mathcal{R}|]}$) and for the rest of them u and v are equally preferred (part 1), or if there is an ordering $\leq_{\mathcal{R}[k]}$ with a strict preference for v over u and all orderings with higher priority see u and v as equally preferred (part 2).

The key fact in the above definition is that $\mathcal{R}$ includes only the preference relations (strictly speaking, the *indices* of the preference relations) that are actually used when building up the new preference ordering. Any change in one's preference ordering might just be dependent on the preferences of the person who might be the most reliable one according to the concerned agent. It might also be the case that the change occurs based on the preferences of all the agents. All these possibilities are covered by the definition above. Also, the general lexicographic upgrade preserves preorders (and thus the class of semantic models) when every preference ordering in $\mathcal{R}$ satisfies the requirements.

The formal language In order to describe the changes the general upgrade operation brings about, the language $\mathcal{L}_{pr}$ is extended in the following way.

Definition 17. *The language $\mathcal{L}_{\{\mathrm{fx}\}}$ extends $\mathcal{L}$ with a modality $\langle \mathrm{fx}^i_{\mathcal{R}} \rangle$ for every agent $i \in \mathsf{A}$ and every lexicographic list $\mathcal{R}$. Given a PR pointed model (M, w), define*

$$(M, w) \Vdash \langle \mathrm{fx}^i_{\mathcal{R}} \rangle \varphi \quad \textit{iff} \quad \big(\mathrm{fx}^i_{\mathcal{R}}(M), w\big) \Vdash \varphi$$

where the PR model $\mathrm{fx}^i_{\mathcal{R}}(M)$ is exactly as M except in $\leq_i$, which is now given by $\leq_{\mathcal{R}}$ (Definition 16). Observe how, since the general lexicographic upgrade is a total function, the semantic interpretation of $[\mathrm{fx}^i_{\mathcal{R}}]\,\varphi := \neg\langle \mathrm{fx}^i_{\mathcal{R}} \rangle \neg\varphi$ is

$$(M, w) \Vdash [\mathrm{fx}^i_{\mathcal{R}}]\,\varphi \quad \textit{iff} \quad \big(\mathrm{fx}^i_{\mathcal{R}}(M), w\big) \Vdash \varphi$$

that is, $\langle \mathrm{fx}^i_{\mathcal{R}} \rangle \varphi \leftrightarrow [\mathrm{fx}^i_{\mathcal{R}}]\,\varphi$.

Thus, the modality $\langle \mathrm{fx}^i_{\mathcal{R}} \rangle$ allows us to express the effects of upgrading the preference relation $\leq_i$ via the general lexicographic upgrade with a lexicographic list $\mathcal{R}$ while keeping the remaining preference relations as before. Let us consider our example once again, this time with reliability orderings given by, $a : a \prec b \prec c$; $b : b \prec c \prec a$; $c : a \prec b \prec c$. Let us consider the list $\mathcal{R}$ consisting of the unique maximally reliable agent. Then we have that Brishti originally preferred restaurant r_2 but after hearing the others' preferences she prefers restaurant r_1, as according to her, Aniket's opinion is the most reliable one. Thus, in the beginning, at each world of the model, there exists some preferred world for agent b where p_2 holds. Once the model gets updated by an announcement of preferences from all the agents, then there exists some preferred world for agent b in the new model where p_1 holds.

Even though the general lexicographic upgrade covers many natural upgrades, there are also 'reasonable' policies that fall outside its scope. For example, one can also consider changing the old preference orderings in a more constrained way, focusing on a specific part of the ordering of the reliable agent(s). Let us consider the following example from [Ghosh and Velázquez-Quesada, 2015a].

Agent i can upgrade her preferences by placing her most reliable agent's most preferred worlds above the rest, then using her old ordering within each zone. Thus, for example, if agent a is agent b's most reliable agent and the individual preferences are as below

$$a : w_2 < w_1 < w_3 \simeq w_4, \qquad b : w_3 < w_4 < w_1 < w_2$$

then such upgrade on b's preferences will create two zones, the upper one with a's most preferred worlds (w_3 and w_4), and the lower one with the remaining worlds (w_1 and w_2). Within each zone, b's old preferences will apply, thus producing the revised preference ordering:

$$b : w_1 <' w_2 <' w_3 <' w_4$$

One can show that no lexicographic list can produce this outcome [Ghosh and Velázquez-Quesada, 2015a]. To model such upgrades, as mentioned in [Ghosh and Velázquez-Quesada, 2015b], we provide the following preference upgrade definition.

Definition 18 (General layered upgrade). *A* layered list $\mathcal{S}$ *over W is a finite (possibly empty) list of pairwise disjoint subsets of W together with a default preference ordering over W. The list's length is denoted by $|\mathcal{S}|$, its kth element is denoted by $\mathcal{S}[k]$ (with $1 \leq k \leq |\mathcal{S}|$), and $\leq^{\mathcal{S}}_{ondef}$ is its default preference ordering. Intuitively, $\mathcal{S}$ defines layers of elements of W in the new preference ordering $\leq_{\mathcal{S}}$, with $\mathcal{S}[1]$ the set of worlds that will be in the topmost layer and $\leq^{\mathcal{S}}_{ondef}$ the preference ordering that will be applied to each individual set and to those worlds not in $\bigcup_{k=1}^{|\mathcal{S}|} \mathcal{S}[k]$. Formally, given $\mathcal{S}$, the ordering $\leq_{\mathcal{S}} \subseteq (W \times W)$ is defined as*

$$u \leq_{\mathcal{S}} v \quad \textit{iff}_{def} \quad \underbrace{\Big(u \leq^{\mathcal{S}}_{ondef} v \;\wedge\; (\{u,v\} \cap \bigcup_{k=1}^{|\mathcal{S}|} \mathcal{S}[k] = \emptyset \vee \bigvee_{k=1}^{|\mathcal{S}|} \{u,v\} \subseteq \mathcal{S}[k])\Big)}_{1}$$
$$\vee \quad \underbrace{\bigvee_{k=1}^{|\mathcal{S}|} \Big(v \in \mathcal{S}[k] \;\wedge\; u \notin \bigcup_{l=1}^{k} \mathcal{S}[l]\Big)}_{2}$$

Thus, $u \leq_{\mathcal{S}} v$ holds if this agrees with the default ordering $\leq^{\mathcal{S}}_{ondef}$ and either neither u nor v are in any of the specified sets in $\mathcal{S}$ or else both are in the same *set (part 1), or if there is a set $\mathcal{S}[k]$ in which v appears and u appears neither in the same set (a case already covered in part 1) nor in one with higher priority (part 2).*

We note that if $|\mathcal{S}| = 0$, then $u \leq_{\mathcal{S}} v$ iff $u \leq^{\mathcal{S}}_{ondef} v$. On the other hand, if $\mathcal{S}$'s sets form a partition of W (i.e., the sets are not only mutually exclusive but also collectively exhaustive), then $\bigcup_{k=1}^{|\mathcal{S}|} \mathcal{S}[k] = W$ and

$$u \leq_{\mathcal{S}} v \quad \text{iff} \quad \underbrace{\Big(u \leq^{\mathcal{S}}_{ondef} v \;\wedge\; \bigvee_{k=1}^{|\mathcal{S}|} \{u, v\} \subseteq \mathcal{S}[k]\Big)}_{1} \vee \underbrace{\bigvee_{k=1}^{|\mathcal{S}|} \Big(v \in \mathcal{S}[k] \;\wedge\; u \notin \bigcup_{l=1}^{k} \mathcal{S}[l]\Big)}_{2}$$

In fact, since $\leq^{\mathcal{S}}_{ondef}$ is used to break ties not only within each $\mathcal{S}[k]$ but also among those worlds not appearing in any such set, the provided definition of a layered list actually just 'abbreviates' (but still it is equivalent to) a list that requires a full partition of W by not writing explicitly the set with the least priority.

Definition 19. *Let $M = (W, \{\leq_i, \preccurlyeq_i\}_{i\in\mathsf{A}}, V)$ be a PR model.*

- *Let $\mathcal{S}$ be a layered list whose default ordering is reflexive, transitive and total; let $j \in \mathsf{A}$ be an agent. The PR model $\mathrm{gy}^j_{\mathcal{S}}(M) = (W, \{\leq'_i, \preccurlyeq_i\}_{i\in\mathsf{A}}, V)$ is such that, for every agent $i \in \mathsf{A}$, $\leq'_i := \leq_{\mathcal{S}(M)}$ if $i = j$, and $\leq'_i := \leq_i$ otherwise.*
- *Let $\boldsymbol{\mathcal{S}}$ be a list of $|\mathsf{A}|$ layered lists whose default ordering are reflexive, transitive and total, with $\boldsymbol{\mathcal{S}}_i$ its ith element. The PR model $\mathrm{gy}_{\boldsymbol{\mathcal{S}}}(M) = (W, \{\leq'_i, \preccurlyeq_i\}_{i\in\mathsf{A}}, V)$ is such that, for every agent $i \in \mathsf{A}$, $\leq'_i := \leq_{\boldsymbol{\mathcal{S}}_i(M)}$*

The formal language In order to describe the changes the general layered upgrade operation brings about, the language is extended in the following way.

Definition 20. *The language $\mathcal{L}^{PR}_{\{\mathrm{gy}\}}$ extends $\mathcal{L}^{PR}$ with a modality $\langle \mathrm{gy}^i_{\mathcal{S}} \rangle$ for every agent $i \in \mathsf{A}$ and every layered list $\mathcal{S}$ whose default ordering is reflexive, transitive and total. Given a PR model M, define*

$$[\![\langle \mathrm{gy}^i_{\mathcal{S}} \rangle\, \varphi]\!]^M := [\![\varphi]\!]^{\mathrm{gy}^i_{\mathcal{S}}(M)}$$

with $\mathrm{gy}^i_{\mathcal{S}}(M)$ as in Definition 19. Note how, by defining $[\mathrm{gy}^i_{\mathcal{S}}]\, \varphi := \neg\langle \mathrm{gy}^i_{\mathcal{S}} \rangle\, \neg\varphi$, then $[\![[\mathrm{gy}^i_{\mathcal{S}}]\, \varphi]\!]^M := [\![\varphi]\!]^{\mathrm{gy}^i_{\mathcal{S}}(M)}$ so $\langle \mathrm{gy}^i_{\mathcal{S}} \rangle\, \varphi \leftrightarrow [\mathrm{gy}^i_{\mathcal{S}}]\, \varphi$ is valid.

The modality $\langle \mathrm{gy}^i_{\mathcal{S}} \rangle$ allows to describe the effects of upgrading agent i's preferences via the general layered upgrade with $\mathcal{S}$, keeping the preferences of the remaining agents as before. This definition can be extended to simultaneous upgrades by asking for a list $\boldsymbol{\mathcal{S}}$ of layered lists and using a modality $\langle \mathrm{gy}_{\boldsymbol{\mathcal{S}}} \rangle$ whose semantic interpretation uses the operation $\mathrm{gy}_{\boldsymbol{\mathcal{S}}}(\cdot)$ of Definition 19.

The process of changing preferences provide in Definition 16 is quite similar to the aggregation model described in Section 3.1 in the sense that group preference is induced by a hierarchy of agents. The main difference being that in the former case, the agent hierarchy is more objective in nature and the preference relations are parametized by these hierarchies, whereas in the latter case, the preference changes are handled in a more subjective way based on agents' reliabilities on one another. The process defined in Definition 18 is much more general in nature as shown in [Ghosh and Velázquez-Quesada, 2015b], and needs more in-depth investigation.

4 Conclusion

There are several interesting research directions that we have not touched upon, and we conclude this chapter with a brief discussion of some of them.

4.1 Strategic reasoning

Perhaps the most important area that we have not touched on at all in our discussion of games is that of games of imperfect information. This is especially important since *communication structure* in strategies is important and worthwhile for study. What to communicate and when it is strategic and when players have only a partial view of the game state and must communicate to learn more as well as coordinate to achieve desired goals, this dictates considerable structure in strategies. For an instance of such reasoning, see [Ramanujam and Simon, 2010].

A more restricted setting is that of *concurrency*, where several games proceed in parallel, and there is no communication across games. Still, players can use information gained from one game in another. This is discussed in Van Benthem et al. [2008] and Ghosh et al. [2010].

An important motivation for endowing strategies with structure is so that we can develop an *algebraic theory* of strategies, similar to algebras of games developed by Goranko [2003], Venema [2003], and van Benthem et al. [2019]. In general, equational theories of strategies await strong theoretical foundations.

Below we list several questions that come up in the course of a search for strategy structure.

- We have discussed games with a fixed number of players (albeit unknown perhaps). How is strategizing affected when the set of players is *unbounded*, and hence potentially infinite? This is the case in games such as the Internet.
- We discussed neighbourhood structures, but in the model, we have merely replaced a flat structure on players by one which has one level depth. It is natural to consider a hierarchical structure of neighbourhoods, and a topological study would be more useful.
- We have talked only of player behaviour in games. A closely related question is one that keeps the strategy space fixed, but asks for *incentive mechanisms* that achieve desired outcomes. Mechanism design in the context of structured strategies is unclear.
- We have suggested that heuristics such as imitation are important in large games. It would be interesting to offer such analysis for a study of herd behaviour and runaway phenomena.

– We talked of game - strategy pairs, to show that they are dependent notions. A theory in which games and strategies are mutually recursive in the other is needed for offering foundations to such reasoning.
– Finite state transducers provide a natural *complexity measure* for strategies: the size of the minimal deterministic finite state machine that can play that strategy. Developing a nuanced complexity theory of strategies based on such notions is a definite need. This requires notions of *strategy reductions* that await further exploration.
– A most critical lacuna in our discussion has been the omission of *randomized* strategies. Logical theories that admit strategy structure as well as randomization are essential for applicability.

There is much more to reasoning about strategies. Perhaps the biggest issue that we have not discussed is that of learning: such learning of strategies may arise during course of play, or in the form of automated statistical learning of strategies from large records of play.

4.2 Preferences

In defense of logical studies on preferences, one can always say that such studies help in a better understanding of the motivational attitude of *preference*. Furthermore, logic frameworks can be considered as a step towards automation and they can be used to specify and verify properties of social algorithms, e.g., procedures involving choices of preferences, combination of individual choices towards forming a group choice and similar other phenomena. From the logic perspective, by expressing properties of preferences in different languages one can study the expressive powers of different logics towards formulating different results encompassing the study on preferences. We list some related questions below that come up in the course of such studies.

– From the aggregation point of view, Girard [2011] considers a certain hierarchy of agents based on which the group preference relations are defined. Where do these hierarchies come from? More importantly, in what other ways can we model the aggregation process so as to avoid the Arrovian impossibility results?
– A more detailed study of the dynamics would also be noteworthy. How would the changes in the hierarchy of agents affect the group preferences? Under what conditions would the group preferences remain unchanged? How could we bring about some nice structures (uniformities) in the preference profiles through logical dynamics?
– From the deliberative view point, we have discussed several policies of preference upgrades based on agent reliabilities. For such kind of studies, one can never be sure whether all possible/reasonable cases were covered. Logic allows us to come up with suitable axioms that all such processes should satisfy, and then one can verify whether the proposed policies are indeed reasonable policies.

- The interaction between knowledge, belief and preferences have not been studied at all in these proposals. One notable exception is the work of Goldbach [2015]. Consideration of these issues would lead to better models of strategic behavior of agents, and notions like manipulability can be studied.
- The deliberative models presented here in terms of preference upgrades are based on the notions of reliability. What about other possibilities? How do we study the deliberative models in a uniform way? Can we combine these notions with the theory of argumentation? How do we study the interplay?
- We have studied the effect of agent reliability over preference changes. What about the opposite effect? Can the agent reliabilities change based on their preferences? If yes, on what basis can we model these changes? A preliminary work to this effect has been done in [Ghosh and Sano, 2017], and there is a lot more to investigate in this respect.
- As mentioned earlier, deliberation may not always lead to consensus on issues, but, as empirical studies suggest, it might lead to certain structured preferences (e.g., single-peaked profiles), which might facilitate in aggregation later (being devoid of the Arrovian impossibility criteria). Can a formal language express this combination of deliberation followed by aggregation in a reasonable manner, so that we can specify and verify the properties of such combinations. This would lead to well-structured social algorithms for automated social processes.

As in the case of strategies, there is much more to reasoning about preferences as well. We have not at all discussed the notion of *learning* about preferences: even though the aggregation process may not lead to learning per se, the deliberation process may well lead to preference elicitation based on others' preferences [Pazzani, 1999].

Bibliography

T. Ågotnes and D. Walther. A logic of strategic ability under bounded memory. *Journal of Logic Language and Information*, 18:55–77, 2009.

T. Agotnes, W. van der Hoek, and M. Wooldridge. On the logic of preference and judgment aggregation. *Autonomous Agents and Multiagent Systems*, 22 (1):4–30, 2011.

R. Alur, T. A. Henzinger, and O. Kupferman. Alternating-time temporal logic. *Lecture Notes in Computer Science*, 1536:23–60, 1998.

H. Andréka, M. D. Ryan, and P.-Y. Schobbens. Operators and laws for combining preference relations. *Journal of Logic and Computation*, 12(1):13–53, 2002.

K. J. Arrow. *Social Choice and Individual Values*. John Wiley & Sons, Inc., New York, London, Sydney, 2nd edition, 1963. First edition published in 1951.

K. J. Arrow, A. K. Sen, and K. Suzumura, editors. *Handbook of Social Choice and Welfare*. Elsevier, 2002. ISBN 978-0-444-82914-6. Two volumes.

R. J. Aumann and J. H. Dreze. When all is said and done, how should you play and what should you expect? http://www.ma.huji.ac.il/raumann/pdf/dp_387.pdf, March 2005.

A. Baltag, S. Smets, and J. Zvesper. Keep hoping for rationality: a solution to the backward induction paradox. *Synthese*, 169:301–333, 2009.

P. Battigalli and M. Siniscalchi. Hierarchies of conditional beliefs and interactive epistemology in dynamic games. *Journal of Economic Theory*, 88:188–230, 1999.

D. Berwanger. Game logic is strong enough for parity games. *Studia Logica*, 75 (2):205–219, 2003.

D. Black. On the rationale of group decision-making. *Journal of Political Economy*, 56:23–34, 1948.

G. Bonanno. Branching time logic, perfect information games and backward induction. *Games and Economic Behavior*, 36(1):57–73, 2001.

G. Bonanno. Reasoning about strategies and rational play in dynamic games. In J. van Benthem, S. Ghosh, and R. Verbrugge, editors, *Models of Strategic Reasoning*, pages 34–62. Springer, 2015.

R. Bredereck, J. Chen, and G. J. Woeginger. Are there any nicely structured preference profiles nearby? In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence (IJCAI)*, pages 62–68. AAAI Press, 2013.

N. Bulling, V. Goranko, and W. Jamroga. Logics for reasoning about strategic abilities in multi-player games. In J. van Benthem, S. Ghosh, and R. Verbrugge, editors, *Models of Strategic Reasoning*, pages 93–136. Springer, 2015.

Y. Chevaleyre, P. E. Dunne, U. Endriss, J. Lang, M. Lemaítre, N. Maudet, J. A. Padget, S. Phelps, J. A. Rodríguez-Aguilar, and P. Sousa. Issues in multiagent resource allocation. *Informatica* ., 30(1):3–31, 2006.

Y. Chevaleyre, U. Endriss, J. Lang, and N. Maudet. Preference handling in combinatorial domains: From ai to social choice. *AI Magazine*, 29(4):37–46, 2008.

V. Conitzer, T. Sandholm, and J. Lang. When are elections with few candidates hard to manipulate? *Journal of the ACM.*, 54(3):37–71, 2007.

B. de Bruin. *Explaining Games: The Epistemic Program in Game Theory.* Synthese Library, Springer, 2010.

F. Dietrich and C. List. Where do preferences come from? *International Journal of Game Theory*, 42(3):613–637, 2013.

J. S. Dryzek and C. List. Social choice theory and deliberative democracy: A reconciliation. *British Journal of Political Science*, 33:1–28, 2003.

H.-D. Ebbinghaus, J. Flum, and W. Thomas. *Mathematical Logic.* Springer, 1996.

J. Elster. The market and the forum. In J. Elster and A. Hylland, editors, *Foundations of Social Choice Theory.* Cambridge University Press, Cambridge, 1986.

U. Endriss. Logic and social choice theory. In A. Gupta and J. van Benthem, editors, *Logic and Philosophy Today*, volume 2, pages 333–377. College Publications, 2011.

S. Enqvist, H. H. Hansen, C. Kupke, J. Marti, and Y. Venema. Completeness for game logic. In *2019 34th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 1–13. IEEE, 2019.

L. Fortnow and D. Whang. Optimality and domination in repeated games with bounded players. In *Proceedings of the twenty-sixth annual ACM symposium on Theory of computing*, pages 741–749. ACM, 1994.

S. Ghosh. Strategies made explicit in dynamic game logic. In *Proceedings of the Workshop on Logic and Intelligent Interaction, ESSLLI 2008*, pages 74–81, 2008.

S. Ghosh and R. Ramanujam. Strategies in games: A logic-automata study. In *Lectures on Logic and Computation*, pages 110–159. Springer, 2011.

S. Ghosh and K. Sano. Valuing others' opinions: Preference, belief and reliability dynamics. In A. R. J. van den Herik and J. Filipe, editors, *Proceedings of the 9th International Conference on Agents and Artificial Intelligence - Volume 2 (ICAART 2017)*, pages 615–623. Scitepress, 2017.

S. Ghosh and F. R. Velázquez-Quesada. Merging information. In J. van Benthem, A. Gupta, and E. Pacuit, editors, *Games, Norms and Reasons: Logic at the Crossroads*, pages 77–96. Springer Netherlands, Dordrecht, 2011.

S. Ghosh and F. R. Velázquez-Quesada. Agreeing to agree: Reaching unanimity via preference dynamics based on reliable agents. In R. Bordini, E. Elkind, G. Weiss, and P. Yolum, editors, *AAMAS 2015*, pages 1491–1499, 2015a.

S. Ghosh and F. R. Velázquez-Quesada. A note on reliability-based preference dynamics. In W. van der Hoek, W. H. Holliday, and W. fan Wang, editors, *LORI 2015*, pages 129 –142, 2015b.

S. Ghosh, R. Ramanujam, and S. Simon. Playing extensive form games in parallel. In *International Workshop on Computational Logic in Multi-Agent Systems*, pages 153–170. Springer, 2010.

P. Girard. Modal logic for lexicographic preference aggregation. In J. van Benthem, A. Gupta, and E. Pacuit, editors, *Games, Norms and Reasons: Logic at the Crossroads*, pages 97–117. Springer Netherlands, Dordrecht, 2011.

R. Goldbach. Modelling democratic deliberation. Master's thesis, University of Amsterdam, 2015.

V. Goranko. Coalition games and alternating temporal logics. *Proceedings of 8th conference on Theoretical Aspects of Rationality and Knowledge (TARK VIII)*, pages 259–272, 2001.

V. Goranko. Basic algebra of game equivalences. *Studia Logica*, 75(2):221–238, 2003.

T. Grüne-Yanoff and S. O. Hansson, editors. *Preference Change*, volume 42 of *Theory and Decision Library*. Springer, 2009. ISBN 978-90-481-2593-7.

J. Habermas. *Between Facts and Norms: Contributions to a Discursive Theory of Law and Democracy*. MIT Press, Cambridge, 1996.

P. Haddawy, V. Ha, A. Restificar, B. Geisler, and J. Miyamoto. Preference elicitation via theory refinement. *Journal of Machine Learning Research*, 4: 317–337, 2003.

J. Halpern. Substantive rationality and backward induction. *Games and Economic Behaviour*, 37:425–435, 2001.

D. Harel, D. Kozen, and R. Parikh. Process logic: Expressiveness, decidability, completeness. *Journal of Computer and System Sciences*, 25(2):144–170, October 1982.

D. Harel, D. Kozen, and J. Tiuryn. *Dynamic Logic*. The MIT Press, October 2000.

P. Harrenstein, W. van der Hoek, J.-J. Meyer, and C. Witteveen. A modal characterization of Nash equilibrium. *Fundamenta Informaticae*, 57:2–4:281–321, 2003.

J. Hintikka. Language-games for quantifiers. In *Studies in Logical Theory*, volume 2 of *American Philosophical Quarterly Monograph Series*, pages 46–72. Blackwell, 1968.

T. Hoshi and A. Isaac. Taking mistakes seriously: Equivalence notions for game scenarios with off equilibrium play. In *LORI-III*, pages 111–124, 2011.

W. Jamroga and W. van der Hoek. Agents that know how to play. *Fundamenta Informaticae*, 63(2-3):185–219, 2004.

J. Knight and J. Johnson. Aggregation and deliberation: On the possibility of democratic legitimacy. *Polotical Theory*, 42:277–296, 1994.

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

J. Lang. Logical preference representation and combinatorial vote. *Annals of Mathematics and Artificial Intelligence.*, 42:37–71, 2004.

C. Lemke and J. Howson. Equilibrium points of bimatrix games. *Journal of the Society of Industrial and Applied Mathematics*, 12:413–423, 1964.

F. Liu. A two-level perspective on preference. *Journal of Philosophical Logic*, 40:421–439, 2011.

D. Miller. Deliberative democracy and social choice. *Political Theory*, 40:54–67, 1992.

D. Monderer and L. Shapley. Potential games. *Games and Economic Behaviour*, 14:124–143, 1996.

J. Nash. Equilibrium points in n-person games. *Proceedings of the National Academy of Sciences*, 36:89–93, 1950.

M. Osborne and A. Rubinstein. *A course in game theory*. MIT Press, 1994.

E. Pacuit. Dynamic models of rational deliberation in games. In J. van Benthem, S. Ghosh, and R. Verbrugge, editors, *Models of Strategic Reasoning*, pages 3–33. Springer, 2015.

R. Parikh. The logic of games and its applications. *Annals of Discrete Mathematics*, 24:111–140, 1985.

R. Parikh and M. Pauly. Game logic - an overview. *Studia Logica*, 75(2):165–182, 2003.

S. Paul and R. Ramanujam. Imitation in large games. In *Proceedings of the first international symposium on games, automata, logics and verification (GandALF)*, Electronic proceedings in theoretical computer science, pages 162–172, 2010.

S. Paul and R. Ramanujam. Dynamic restriction of choices: Synthesis of societal rules. In H. van Ditmarsch, J. Lang, and S. Ju, editors, *LORI*, volume 6953 of *Lecture Notes in Computer Science*, pages 28–50. Springer, 2011a. ISBN 978-3-642-24129-1.

S. Paul and R. Ramanujam. Neighbourhood structure in large games. In *Proceedings of the 13th conference on the Theoretical Aspects of Rationality and Knowledge (TARK 2011)*, pages 121–130, 2011b.

S. Paul, R. Ramanujam, and S. Simon. Stability under strategy switching. In K. Ambos-Spies, B. Löwe, and W. Merkle, editors, *Proceedings of the 5th Conference on Computability in Europe (CiE)*, volume 5635 of *LNCS*, pages 389–398, 2009a.

S. Paul, R. Ramanujam, and S. Simon. Dynamic restriction of choices: A preliminary logical report. In *Proceedings of the 12th Conference on Theoretical Aspects of Rationality and Knowledge (TARK)*, pages 218–226, 2009b.

S. Paul, R. Ramanujam, and S. Simon. Automata and compositional strategies in extensive form games. In J. van Benthem, S. Ghosh, and R. Verbrugge, editors, *Models of Strategic Reasoning*, pages 174–201. Springer, 2015.

M. Pauly. Game logic for game theorists. Technical report, CWI, 2000.

M. Pauly. *Logic for Social Software*. PhD thesis, University of Amsterdam, October 2001.

M. Pazzani. A framework for collaborative, content-based and demographic filtering. *Artificial Intelligence Review*, 13(5-6):393–408, 1999.

D. G. Pearce. Rationalizable strategic behaviour and the problem of perfection. *Econometrica*, 52:1029–1050, 1984.

A. Perea. Backward induction versus forward induction reasoning. *Games*, 1: 168–188, 2010.

A. Perea. *Epistemic Game Theory: Reasoning and Choice*. Cambridge University Press, 2012.

J. Perote-Pena and A. Piggins. A model of deliberative and aggregative democracy. *Economics and Philosophy*, 31(1):93–121, 2015.

R. Ramanujam. Logical player types for a theory of play. In A. Baltag and S. Smets, editors, *Johan van Benthem on Logic and Information Dynamics*, pages 509–528. Springer, 2014.

R. Ramanujam and S. Simon. Dynamic logic on games with structured strategies. In *Proceedings of the 11th International Conference on Principles of Knowledge Representation and Reasoning (KR-08)*, pages 49–58. AAAI Press, 2008a.

R. Ramanujam and S. Simon. A logical structure for strategies. In *Logic and the Foundations of Game and Decision Theory (LOFT 7)*, volume 3 of *Texts in Logic and Games*, pages 183–208. Amsterdam University Press, 2008b.

R. Ramanujam and S. Simon. Dynamic logic of tree composition. In *Perspectives in Concurrency Theory*, pages 408–430. CRC Press, 2009.

R. Ramanujam and S. Simon. A communication based model for games of imperfect information. In P. Gastin and F. Laroussinie, editors, *CONCUR*, volume 6269 of *Lecture Notes in Computer Science*, pages 509–523. Springer, 2010. ISBN 978-3-642-15374-7.

N. Troquard, W. van der Hoek, and M. Wooldridge. Reasoning about social choice functions. *Journal of Philosophical Logic*, 40:473–498, 2011.

J. van Benthem. Games in dynamic-epistemic logic. *Bulletin of Economic Research*, 53(4):219–248, 2001.

J. van Benthem. Extensive games as process models. *Journal of Logic, Language and Information*, 11:289–313, 2003a.

J. van Benthem. Logic games are complete for game logics. *Studia Logica*, 75: 183–203, 2003b.

J. van Benthem. Decisions, actions, and games: a logical perspective. In *Third Indian Conference on Logic and Applications*, pages 1–22, 2009.

J. van Benthem. Exploring a theory of play. In *TARK*, pages 12–16, 2011.

J. van Benthem. In praise of strategies. In J. van Eijck and R. Verbrugge, editors, *Games, Actions and Social Software*, volume 7010 of *Lecture Notes in Computer Science*, pages 96–116. Springer, 2012.

J. van Benthem. *Logic in Games*. The MIT Press, 2014.

J. van Benthem and F. Liu. Diversity of logical agents in games. *Philosophia Scientiae*, 8:163–178, 2004.

J. Van Benthem, S. Ghosh, and F. Liu. Modelling simultaneous games in dynamic logic. *Synthese*, 165(2):247–268, 2008.

J. van Benthem, P. Girard, and O. Roy. Everything else being equal: A modal logic for *ceteris paribus* preferences. *Journal of Philosophical Logic*, 40:83–125, 2009.

J. van Benthem, E. Pacuit, and O. Roy. Toward a theory of play: A logical perspective on games and interaction. *Games*, 2(1):52–86, 2011.

J. van Benthem, N. Bezanishvili, and S. Enqvist. A new game equivalence, its logic and algebra. *Journal of Philosophical Logic*, 48:649–684, 2019.

W. van der Hoek, W. Jamroga, and M. Wooldridge. A logic for strategic reasoning. *Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multi-Agent Systems*, pages 157–164, 2005.

F. R. Velázquez-Quesada. Reliability-based preference dynamics: lexicographic upgrade. *Journal of Logic and Computation*, 27(8):2341–2381, 2017.

Y. Venema. Representation of game algebras. *Studia Logica*, 75(2):239–256, 2003.

J. von Neumann and O. Morgenstern. *Theory of games and economic behaviour*. Princeton University Press, 1947.

B. von Stengel. Computing equilibria for two-person games. *Handbook of Game Theory*, 3:1723–1759, 2002.

G. H. von Wright, editor. *The logic of preference.* Edinburgh University Press., 1963.

D. Walther, W. van der Hoek, and M. Wooldridge. Alternating-time temporal logic with explicit strategies. In *Proceedings of the 11th Conference on Theoretical Aspects of Rationality and Knowledge (TARK-2007)*, pages 269–278, 2007.

E. Zermelo. Über eine Anwendung der Mengenlehre auf die Theorie des Schachspiels,. In *Proceedings of the Fifth Congress of Mathematicians*, pages 501–504. Cambridge University Press, 1913.