A Logical Study of the Improvement Graphs formed from Games

By

Ramit Das

MATH10201604001

The Institute of Mathematical Sciences, Chennai

A thesis submitted to the

Board of Studies in Mathematical Sciences

In partial fulfillment of requirements

for the Degree of

DOCTOR OF PHILOSOPHY

of

HOMI BHABHA NATIONAL INSTITUTE

March, 2022

Homi Bhabha National Institute

Recommendations of the Viva Voce Committee

As members of the Viva Voce Committee, we certify that we have read the dissertation prepared by Ramit Das entitled "A Logical Study of the Improvement Graphs formed from Games" and recommend that it may be accepted as fulfilling the thesis requirement for the award of Degree of Doctor of Philosophy.

_______________________________________ Date:
Chairman - Venkatesh Raman

_______________________________________ Date:
Guide/Convenor - R Ramanujam

_______________________________________ Date:
Examiner - Sujata ghosh

_______________________________________ Date:
Member 1 - SP Suresh

_______________________________________ Date:
Member 2 - CRS

_______________________________________ Date:
Member 3 - Vikram

Final approval and acceptance of this thesis is contingent upon the candidate's submission of the final copies of the thesis to HBNI.

I hereby certify that I have read this thesis prepared under my direction and recommend that it may be accepted as fulfilling the thesis requirement.

Date:

Place: R Ramanujam (Guide)

STATEMENT BY AUTHOR

Ramit Das

DECLARATION

I hereby declare that the investigation presented in the thesis has been carried out by me. The work is original and has not been submitted earlier as a whole or in part for a degree / diploma at this or any other Institution / University.

Ramit Das

LIST OF PUBLICATIONS ARISING FROM THE THESIS

Journal

1. Title, All authors in the same order as appeared in the paper, Journal, Year, Vol., starting page-ending page.

2.

Chapters in books and lecture notes

1.

2.

Conferences

1.

2.

Others

1.

2.

Ramit Das

DEDICATIONS

To the people who bleed for their craft

ACKNOWLEDGEMENTS

Will Fill In

Contents

Summary

This is the summary. Minimum one page and maximum two pages on key findings of the research described in the thesis. Should not provide chapter wise descriptions as done in Synopsis

List of Figures

List of Tables

Chapter 1

Introduction

This is the beginning.

1.1 Literature Survey

Chapter 2

Improvement Graphs

Blah blah blah...

Ramit

Here I will introduce the notion of Improvement graphs.

The objectives of game theory would involve being able to come up with systems that optimise the interactions among the people and achieve cooperative goals that it sets out to achieve even when the people are non cooperative among themselves. It would not be far fetched to say that through the use and understanding of what we call as *g*ames we hope to achieve a futuristic civilisation where human beings can become a global collective form just like the beehive is thought of an organism in of itself.

Towards such lofty goals we get to the mathematical structure of what a noncooperative normal form one shot game is.

2.1 Definitions

Definition 2.1.1: Normal Form Game

$\mathcal{G} \;=\; < \Sigma_{i\in[n]}, \pi_{i\in[n]} >$ for n- players where $n > 1$.

Σ_i Represents a set of choice of strategies that can be chosen by player i.

Set of all Strategy profiles $\Sigma = \Sigma_1 \times \Sigma_2 \times \ldots \Sigma_n$

Payoff Each $\pi_i : \Sigma \mapsto \mathbb{R}$ is the payoff function for each player i.

Each element $\sigma \in \Sigma$, $\sigma = (\sigma_1, \sigma_2, \ldots, \sigma_n)$ is known as a **strategy profile** and represents the ordered list of the one shot choices, σ_i, by each player i . Suppose the payoff function π_i at $\sigma = (\sigma_1, \sigma_2, \ldots, \sigma_n)$ is 5, this means that player i gets a payoff of value 5 when the entire group of players have the collection of one shot choices represented by σ.

Notation $\sigma_{-i} = (\sigma_1, \sigma_2, \ldots, \sigma_{i-1}, \sigma_{i+1}, \ldots, \sigma_n)$ or $(\sigma_j)_{j\neq i}$.

Definition 2.1.2: Extended Normal Form Game

$\mathcal{G} \;=\; < \Sigma_{i\in[n]}, \pi_{i\in[n]}, (\Omega, \preceq) >$ for n- players where $n > 1$.

Generalised notion of Payoffs $\pi_i : \Sigma \mapsto \Omega$.

Intuition Behind Improvement Graphs

We wish to study the individual, rational, strategic decision making between presented alternatives (in the non-cooperative setting). One potential form of reasoning in such a situation is to envisage all possible strategic choices by others, consider one's own response to each, then others' response to it in their turn, and so on *ad infinitum*, with Nash equilibrium representing fixed-points of such an iteration..

Such reasoning, which we might call *improvement dynamics*, is similar to but distinct from rational decision making under uncertainty; it is also similar to but distinct from epistemic reasoning. The former is about optimization, selecting the 'best' option in light of one's information; the latter is about 'higher order information' involving information about others' information etc.

Improvement dynamics intends to yield the same end results as these, but operates at a more operational, computational level, and reasoning about it can be seen as reasoning at the level of computations searching for equilibria.

Formalisation

We now formalise our ideas above by presenting the definition of an Improvement Graph for a n-player game.

Definition 2.1.3: Improvement Graph

$I_{\mathcal{G}} = (\Sigma, \rightarrow_1, \rightarrow_2, \ldots, \rightarrow_n)$.

Notion of Improvement Given a strategy profile, $\sigma = (\sigma_1, \sigma_2, \ldots, \sigma_n)$, player i is said to **improve** in a strategy profile, σ', if $\sigma'_{-i} = \sigma_{-i}$ and $\sigma'_i \neq \sigma_i$ and $\pi_i(\sigma) \preceq \pi_i(\sigma')$.

Improvement Edge The $\rightarrow_i \subseteq \Sigma \times \Sigma$ represents an edge joining two strategy profiles which are related by the notion of improvement for player i and labelled by i.

Now that we have introduced what the Improvement Graph is. it will be easier to explain what we intend to do with the thesis here. We will try to capture properties of the games either on this Improvement Graph or try to capture the properties directly from the game description without forming this behemoth (it is exponential in the number of player n) of a combinatorial structure.

2.1.1 Examples

We take the prisoner's dilemma game and show the improvement graph associated with it. We have two players 1 and 2 denoted by blue and red colours respectively. The improvement arrows of each player are also color coded by their respective colors. Strategy A is basically to maintain silence and not reveal the details, whereas strategy B is to betray the other guy involved.

		Player 2	
		Strategy A	Strategy B
Player 1	Strategy A	$(-1,-1)$	$(-3,0)$
	Strategy B	$(0,-3)$	$(-2,-2)$

Table 2.1: Payoff Matrix For Prisoner's Dilemma

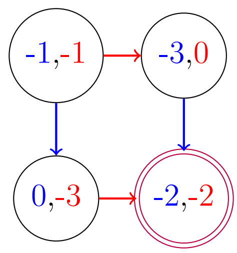

Figure 2.1: Improvement Graph for Prisoner's Dilemma with payoffs given in Table 2.1

Next we take the Matching Pennies and show the improvement graph associated with it. This game doesn't have a nash equilibrium. We preserve the same colors for each players. The strategies here are Heads (H) and Tails(T).

		Player 2	
		H	T
Player 1	H	$(1, 0)$	$(0, 1)$
	T	$(0, 1)$	$(1, 0)$

Table 2.2: Payoff Matrix For Matching Pennies

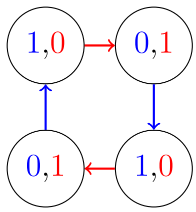

Figure 2.2: Improvement Graph for Matching Pennies with payoffs given in Table 2.2

2.2 Variants of Games

2.2.1 Social Network Games

Definition 2.2.1: Social Network Games

$\mathcal{G} \;=\; < [n], E, \Sigma_{i\in[n]}, \pi_{i\in[n]}, (\Omega, \preceq) >$

Structure over players In place of a flat set of n-players now you got a player relationship graph $([n], E)$ where E is the directed edges of the player graph.

Payoff Now, π_i depends only on the neighbourhood of the player i $\mathcal{N}_i j$ Let's define, $\Sigma^i = \times_{\{j\in\mathcal{N}_i\}}\Sigma_j$ and $\pi_i : \Sigma^i \mapsto \Omega$

Succinct Representation Since the payoff now depends only on the neighbourhood of the given player graph, we can now have polynomial representation of the game in the size of the strategy and the player set.

Will have to fill in the idea of thresholding in the payoff

2.2.2 Priority Separable Games

Definition 2.2.2: Separable Games

$\mathcal{G} = \ < \Sigma_{i\in[n]}, \pi_{i,j\in[n]}, (\Omega, \preceq, , \oplus) >$ for n- players where $n > 1$.

Here we have an even more succinct representation of the game.

Partial payoffs $\pi_{i,j} : \Sigma_i \times \Sigma_j \mapsto \Omega$ focus on a particular player and it's relation with another player.

Actual Payoff From these partial payoffs the payoff for the particular player i is computed in a form of an operation defined by $\oplus$. Hence, the actual payoff for a particular player is $\pi_i(\sigma) \ = \ \oplus_{\{j\in[n]\}} \ \pi_{i,j}(\sigma_i, \sigma_j)$.

Priority separable games. Let N be the set of players and $(S_i)_{i\in N}$ be the set of strategies for each player. Let O be a finite set of outcomes and for all $i \in N$, let $\ll_i \subseteq O \times O$ be a strict total ordering over the outcome set. Let $G = (N, E)$ be a directed graph (without self loops) and for each $i \in N$, let $R(i) = \{j \mid (j, i) \in E\}$ be the neighbourhood of i in G. We also associate a priority ordering within the neighbourhood for each node $i \in N$ and denote this by the relation $\triangleright_i \subseteq R(i) \times R(i)$. For $i, j \in N$, let $p_{i,j} : S_i \times S_j \to O$ be a partial payoff function. A priority separable game is specified by the tuple $\mathcal{G} = (G, (S_i)_{i\in N}, O, (\ll_i)_{i\in N}, (\triangleright_i)_{i\in N}, (p_{i,j})_{i,j\in N})$. Observe that $\mathcal{G}$ has a compact representation that is polynomial in $|N|$, $\max_{i\in N} |S_i|$.

Given a strategy profile s, the payoff for player $i \in N$ is then defined as $p_i(s) = \times_{j\in R(i)} p_{i,j}$. Let $p_i^*(s)$ denote the reordering of the tuple $p_i(s)$ in decreasing order of the priority of neighbours of i. That is, if $R(i) = \{i_1, \ldots, i_k\}$ and $i_1 \triangleright_i i_2 \triangleright_i \cdots \triangleright_i i_k$, then for $j \in \{1, \ldots, k\}$, $(P_i^*(s))_j = p_{i,i_j}(s_i, s_{i_j})$. In order to analyse the strategic aspect of the game, we need to define how players compare between strategy profiles.

For $i \in N$, we define the relation $\preceq_i \subseteq S \times S$ as follows: $s \preceq_i s'$ if $p_i^*(s) \preceq^{lex} p_i^*(s')$ where $\preceq^{lex}$ denotes the lexicographic ordering. All the basic notions in games that we introduced earlier, like better response, best response, improvement graph and Nash equilibrium remain the same and is based on the ordering of strategy profiles $(\preceq_i)_{i \in N}$.

Priority separable games form a qualitative subclass of *polymatrix games* [**?**], a class of strategic form games where payoffs are quantitative and pairwise separable. A strategic form game $(N, (S_i)_{i \in N}, (u_i)_{i \in N})$ with $u_i : S \to \mathbb{R}$ is a polymatrix game if for every $i, j \in N$ there exist a partial utility function $u_{i,j}$ over the domain $S_i \times S_j$ such that for all $s \in S$, $u_i(s) = \Sigma_{j \in R(i)} u_{i,j}(s_i, s_j)$. Priority separable games are qualitative versions of strategic games with pairwise separable payoffs where players' preferences are defined using a lexicographic ordering over their local neighbourhoods. Note that priority separable games allow $\ll_i$ to be different for each player $i \in N$.

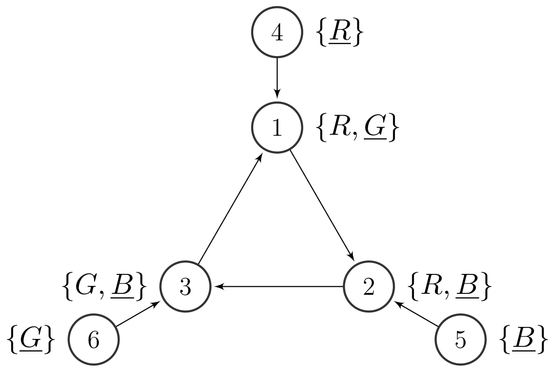

Figure 2.3: A priority separable game

Example 1. Consider the game where $N = \{1, \ldots, 6\}$ and the graph G is as given in figure 2.3. For $i \in N$ the set of strategies S_i is specified in figure 2.3 as a label next to each node in G. Let $O = \{0, 1\}$ with $0 \ll_i 1$ for all $i \in N$. For $i, j \in N$, let $p_{i,j} = 1$ if $s_i = s_j$ and $p_{i,j} = 0$ if $s_i \neq s_j$. Let $3 \rhd_1 4$, $1 \rhd_2 5$ and $2 \rhd_3 6$. For $j \in \{4, 5, 6\}$, $|S_j| = 1$ and $R(j) = \emptyset$. Consider the strategy profile $s = (G, B, B, R, B, G)$ which is denoted with an underline in figure 2.3. Note that in s player 1 is not playing its best response and has a profitable deviation to R.

A natural question is whether priority separable games always have a pure Nash equilibrium. Below we show that the class of priority separable games need not always have a pure Nash equilibrium using an example which is similar to the one given in [?] for polymatrix games.

Example 2. Consider the game given in example 1 along with the neighbourhood graph given in figure 2.3. For players $i \in \{4, 5, 6\}$, $R_i = \emptyset$ and since $|S_i| = 1$, for all $s \in S$, s_i is a best response to s_{-i}. Thus in each strategy profile s only the choices made by players $1, 2$ and 3 are relevant. Below we enumerate all such strategy profiles and underline a strategy which is not a best response for each strategy profile. It then follows that this game does not have a Nash equilibrium. $(R, R, \underline{B})$, $(\underline{R}, R, G)$, $(R, \underline{B}, B)$, $(R, \underline{B}, G)$, $(\underline{G}, R, B)$, $(G, \underline{R}, G)$, $(\underline{G}, B, B)$, $(G, B, \underline{G})$.

Given that priority separable games need not always have a Nash equilibrium, an immediate question is whether there is an efficient procedure to check if a Nash equilibrium exists in this class of games. We show that checking for the existence of a Nash equilibrium is NP-complete. While the upper bound is straightforward, to show NP-hardness we give a reduction from 3-SAT using an argument similar to the one in [?].

Theorem 2.2.1: Complexity of Deciding Nash Equilibria for Priority Separable Games

Given a priority separable game $\mathcal{G}$, deciding if $\mathcal{G}$ has a Nash equilibrium is NP-complete.

Proof. Given a priority separable game $\mathcal{G}$ and a strategy profile s, in polynomial time we can verify if s is a Nash equilibrium in $\mathcal{G}$. It follows that the problem is in NP. We show NP-hardness by giving a reduction from 3-SAT.

Suppose the instance of the 3-SAT formula is $\varphi = (a_1 \vee b_1 \vee c_1) \wedge (a_2 \vee b_2 \vee c_2) \wedge \ldots \wedge (a_k \vee b_k \vee c_k)$ with k clauses and m propositional variables $x_1, \ldots, x_m$. For $j \in \{1, \ldots, k\}$, a_j, b_j and c_j are literals of the form x_l or $\neg x_l$ for some $l \in \{1, \ldots, m\}$. We construct a priority separable game $\mathcal{G}_\varphi$ with the neighbourhood graph structure $G = (N, E)$ such that $\mathcal{G}_\varphi$ has a Nash equilibrium iff φ is satisfiable.

For every propositional variable x_l where $l \in \{1, \ldots, m\}$, we add a player X_l in $\mathcal{G}_\varphi$ with $S_{X_l} = \{\top, \bot\}$. With each clause $a_j \vee b_j \vee c_j$ for $j \in \{1, \ldots, k\}$, we associate 9 players whose neighbourhood is specified by the graph given in figure 2.4. The strategy set for each such node (or player) in the graph is specified as a label next to the node. We use x, y, z as variables where $x, y, z \in \{\top, \bot\}$ whose values are specified as part of the reduction. We denote this graph by $F_j(x, y, z)$ indicating that x, y and z are parameters.

For a literal d, let $\lambda(d) = \top$ if d is a positive literal and $\lambda(d) = \bot$ if d is a negative literal. For each clause with literals a_j, b_j and c_j, which is of the form x_l or $\neg x_l$, we add to $\mathcal{G}_\varphi$ the subgraph $F_j(\lambda(a_j), \lambda(b_j), \lambda(c_j))$ and an edge from X_l to the node A_j, B_j or C_j. Let $O = \{0, 1\}$ with $0 \lll_i 1$ for all $i \in N$. For all $i, i' \in N$, we define $p_{i,i'} = 1$ if $s_i = s_{i'}$ and $p_{i,i'} = 0$ if $s_i \neq s_{i'}$. For each subgraph $F_j(x, y, z)$ corresponding to the clause $(a_j \vee b_j \vee c_j)$ and nodes A_j, B_j and C_j let $X_{l[A_j]}, X_{l[B_j]}, X_{l[C_j]}$ denote the nodes such that $(X_{l[A_j]}, A_j) \in E$, $(X_{l[B_j]}, B_j) \in E$ and $(X_{l[C_j]}, C_j) \in E$ respectively for $l[A_j], l[B_j], l[C_j] \in \{1, \ldots, m\}$. We specify the priority ordering for all players i with $|R(i)| > 1$ as follows. For each subgraph $F_j(x, y, z)$ we have,

- $X_{l[A_j]} \rhd_{A_j} 6_j \rhd_{A_j} 7_j \rhd_{A_j} C_j$.
- $X_{l[B_j]} \rhd_{B_j} 4_j \rhd_{B_j} 8_j \rhd_{B_j} A_j$.
- $X_{l[C_j]} \rhd_{C_j} 5_j \rhd_{C_j} 9_j \rhd_{C_j} B_j$.

The crucial observation used in the reduction is the following. Consider the subgraph H_j induced by the nodes in $F_j(x, y, z)$ for $j \in \{1, \ldots, k\}$ along the nodes

$X_{l[A_j]}, X_{l[B_j]}, X_{l[C_j]}$. Consider the priority separable game $\mathcal{G}(H_j)$ induced by nodes in H_j and the neighbourhood structure specified by H_j. Observe that a strategy profile t in $\mathcal{G}(H_j)$ is a Nash equilibrium iff at least one of the following conditions hold: $t_{A_j} = t_{X_{l[A_j]}}$ or $t_{B_j} = t_{X_{l[B_j]}}$ or $t_{B_j} = t_{X_{l[B_j]}}$. Using this observation, we can argue that $\mathcal{G}_\varphi$ has a Nash equilibrium iff φ is satisfiable.

Suppose s is a Nash equilibrium in $\mathcal{G}_\varphi$. Consider the valuation $v_s : \{x_1, \ldots, x_m\} \to \{\top, \bot\}$ defined as follows: $x_l = s_{X_l}$. From the observation above, it follows that for every $F_j(x, y, z)$ for $j \in \{1, \ldots, k\}$ at least one of the following conditions hold: $s_{A_j} = s_{X_{l[A_j]}}$ or $s_{B_j} = s_{X_{l[B_j]}}$ or $s_{B_j} = s_{X_{l[B_j]}}$. Assume without loss of generality that $s_{A_j} = s_{X_{l[A_j]}}$. By the definition of $\mathcal{G}_\varphi$, we have $S_{A_j} \cap S_{X_{l[A_j]}} = \{\lambda(a_j)\}$. By the definition of v_s we have $v_s(x_{l[A_j]}) = \lambda(a_j)$. This implies that $v_s \models a_j$ and therefore $v_s \models a_j \vee b_j \vee c_j$. Since this holds for all clauses, it follows that $v_s \models \varphi$.

Conversely, suppose φ is satisfiable and let $v \models \varphi$ for some valuation $v : \{x_1, \ldots, x_m\} \to \{\top, \bot\}$. Consider the partially defined strategy profile s^v where $s^v_{X_l} = v(x_l)$ for all $l \in \{1, \ldots, m\}$. Since $v \models \varphi$, for all clauses $a_j \vee b_j \vee c_j$, for $j \in \{1, \ldots, k\}$ we have $v \models a_j$ or $v \models b_j$ or $v \models c_j$. Without loss of generality suppose $v \models a_j$. By definition of $\mathcal{G}_\varphi$ we have $S_{A_j} \cap S_{X_{l[A_j]}} = \{\lambda(a_j)\}$. Therefore, the unique best response for node A_j in the game $\mathcal{G}_\varphi$ is the strategy $\lambda(a_j)$. This holds for all clauses and therefore, it is possible to extend s^v to a strategy profile which is a Nash equilibrium in $\mathcal{G}_\varphi$. $\square$

2.2.3 Large Games

have to fill in

Game theory provides elegant models of strategic interaction between players. Solution concepts predict stable play by rational players. When players make choices, reason about how others make choices and consider their own response,

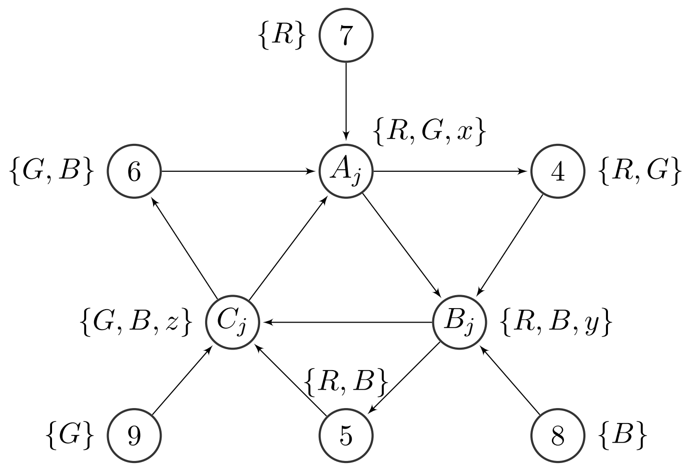

Figure 2.4: Gadget $F_j(x, y, z)$

and further strategise how others would respond to their own choices, mutual intersubjectivity plays an essential role in strategisation. Common knowledge of, or stable common belief in, players' rationality offers an epistemic justification for equilibrium notions. Central to such reasoning is the modelling of outcomes, assigning an outcome to each player for every strategy profile.

But, in practise many social situations involve a large number of players involved in interactive and strategic decision making, but the outcome does not depend on who chooses what, but *how many players* make a particular choice.

The common knowledge assumption does not stand in these scenario and is a more realistic modelling of social phenomena than most of what game theory literature offers

For instance, consider a city (such as Singapore) deciding 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 all about. In such analyses, we are really interested in what fraction of the population chooses each option.

Elaborate further on examples like - a world having to design policies with the covid emergency, market behaviour of humans, congestion games etc.

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 Twitter or YouTube dramatically changes traffic, 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.

These are the situations studied under the rubric of **large games** or *anonymous games*, so-called since player identity does not matter for deciding the outcomes in such games. Payoffs are associated not with strategy profiles, but with choice distributions. Suppose that there are k strategies used in the population. Then the outcome is specified as a map $\mu : \Pi_k(n) \to P^k$, where $\Pi_k(n)$ is a set of distributions: k-tuples that sum up to n, and P is the set of payoffs. Thus every player playing the j^{th} strategy gets the payoff given by the j^{th} component specified by μ for a given distribution.

The advantage of working with large games is that problems of mutual intersubjectivity are eliminated, as 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. As a result, the strategy spaces are much "smoother", though discontinuities in payoff functions can be a serious issue.

Large anonymous games have been extensively studied by game theorists. See, for instance, [**?**, **?**, **?**] and the references therein. What is of interest in such games is that pure strategy equilibria exist for many subclasses, and the fixed-point computation is carried out in a discrete space ([**?**]).

Definition 2.2.3: Large Games

$\mathcal{G} \;=\; <[n], \Sigma, \Pi_{s\in\Sigma}>$ for n- players where $n > 1$

Distribution Given an instance of play in a large game, each member of the population chooses a particular strategy from the fixed strategy set. This induces a distribution of the strategies across the population of players involved. If $\sigma \in \Sigma^n, \sigma = (\sigma_1, \sigma_2, \ldots, \sigma_n)$ is a strategy profile, then consider $\rho = (\rho_1, \ldots, \rho_{|\Sigma|})$ where $\rho_i = |\{\ j \in N \mid \sigma(j) = i\}|$ Such ρ are the distributions of strategies in strategy profiles, let us call them $Y[n]$.

utility $\Pi_s : Y[n] \to \Omega$

Payoff Let f be the transformation function that converts strategy profiles to distributions. Payoff for the particular player p, $\pi_{i\in[n]}$, for the strategy profile σ, is $\pi_i(\sigma) = \Pi_{\sigma_i}(f(\sigma))$

2.3 Properties of the Improvement Graphs

Here I will list out the interesting properties we are after! he properties of interest are as follows.

Nash Equilibrium First we define *Best response*. We say, σ_i is the *best response* to σ_{-i} by player i **if** $\forall \sigma_i' \in \Sigma_i\ \pi_i(\sigma_i, \sigma_{-i}) \geq \pi_i(\sigma_i', \sigma_{-i})$. We call a joint **strategy profile** σ a Nash Equilibrium if each σ_i is a *best response* to σ_{-i}.

This corresponds to a sink vertex in the $I_{\mathcal{G}}$.

Finite Improvement Property First we define a path in the space of all strategy profile Σ. It is a sequence of strategy profiles $(\sigma^1, \sigma^2, \ldots)$ such that for every $k > 1$ there is a player i such that $\sigma^k = (\sigma_i', \sigma_{-i}^{k-1})$ for some $\sigma_i' \neq \sigma_i^{k-1}$. A path is called an **improvement path** if it is maximal and $\forall k, \pi_i(\sigma^{k-1}) \preceq \pi_i(\sigma^k)$ where i is the particular player where a better response could be made to go to σ^k from σ^{k-1}.

These correspond to maximal paths in $I_{\mathcal{G}}$.

The game is said to have the Finite Improvement Property if all such paths are finite. This corresponds to a DAG like structure on $I_{\mathcal{G}}$.

Weak Finite Improvement Property

Squares Let $\mathcal{G}$ be a social network game and players i, j such that $\mathcal{N}_i \cap \mathcal{N}_j = \emptyset$.

Then, for all $\sigma_0, \sigma_1, \sigma_2 \in \Sigma^n$ we have:

- If $\sigma_0 \rightarrow_i \sigma_1$ and $\sigma_0 \rightarrow_j \sigma_2$, then there exists $\sigma_3 \in \Sigma^n$ such that $\sigma_1 \rightarrow_j \sigma_3$ and $\sigma_2 \rightarrow_i \sigma_3$.
- If $\sigma_0 \rightarrow_i \sigma_1$ and $\sigma_1 \rightarrow_j \sigma_2$, then there exists $\sigma_3 \in \Sigma^n$ such that $\sigma_0 \rightarrow_j \sigma_3$ and $\sigma_3 \rightarrow_i \sigma_2$.

Chapter 3

Logical Description of Improvement Graphs

Here I will discuss the preliminaries of the main logics whose variants we will use for our purposes.

3.1 Fist Order Logic with the least fixed point operator

The central ideas and challenges of using FO(lfp). Currently only MLFP is written this will be generalised.

3.1.1 Syntax and Semantics

Definition 3.1.1: Syntax-n-Semantics of MLFP

Induction parameter Given relational vocabulary σ and an additional unary relational symbol $R \notin \sigma$ put explicitly as a parameter we can form $\phi(R, \vec{x})$ formulas of vocabulary $\sigma \cup \{R\}$

Operator For each $\mathfrak{A} \in \sigma$-structures, the formula ϕ induces the following operator, $F_\phi : 2^D \mapsto 2^D$ where

$$F_\phi(X) \;=\; \{\vec{a} \mid \mathfrak{A} \models \phi(X/R, \vec{a})\}$$

lfp operation Now to the syntax of FO definable formulas over $\sigma \cup \{R\}$ we add in another formula of the form $\psi(y) \;=\; [\mathbf{lfp}_{R,\vec{x}}\ \phi(R, \vec{x})](y)$

Semantics The semantics will be as follows

$$\mathfrak{A} \models \psi(y/a) \quad \Longleftrightarrow \quad a \in \mathbf{lfp}(F_\phi)$$

3.1.2 Examples

3.2 Modal Logic

The power of modal logics and the propositional dynamic variant that we will get to.

3.2.1 Syntax and Semantics

We have a countably infinite set of atomic propositions $\mathcal{P} = \{p_0, p_1, \ldots\}$ and the logical connectives $\vee, \wedge, \neg$ and unary modality known as $\Diamond$.

The set Φ of formulas of modal logic will be the smallest set satisfying the following :

- Every atomic propostion p is a member of Φ.
- If α is a member of Φ, so is $\neg\alpha$.
- If α, β be members of Φ, then so is $(\alpha \vee \beta)$ and $(\alpha \wedge \beta)$
- If α is a member of Φ so is $\Diamond\alpha$

bf Notation We represent the dual of $\Diamond$ as the modality $\Box$ which is defined as $\Box\alpha := \neg\Diamond\neg\alpha$

Semantics

We interpret the propositional logic formulas in a model $M = (W, R, V)$ where W is a set of worlds, $R \subseteq W \times W$ is the edge relation between the worlds and $V : W \implies 2^{\mathcal{P}}$ is the valuation function.

We want to define the notion of truth at a world $w \in W$ for the model M by the inductive definition as follows.

$$
\begin{array}{lll}
M, w & \models p & \text{iff } p \in V(w) \text{ for } p \in \mathcal{P} \\
M, w & \models \neg\alpha & \text{iff } M, w \not\models \alpha \\
M, w & \models \alpha \vee \beta & \text{iff } M, w \models \alpha \text{ or } M, w \models \beta \\
M, w & \models \Diamond\alpha & \text{iff there exists } w' \text{ such that } (w, w') \in R \text{ and } M, w' \models \alpha
\end{array}
$$

3.2.2 Examples

Chapter 4

Game Dynamics in MLFP

the order as always merciless

Song: A Quiet Life by Teardo & Bargeld

We start our foray into the main aspects of the thesis. We begin with the ideas of trying to express game theoretic properties. This is done in the hope that we can find a uniform way to express properties among related areas, in our context, central ideas from the field of computational social choice theory - Fair Divisions and Voting!

We go about this exercise by choosing a fairly powerful logic - FO with least fixed points. The advantages of First order logic are immediately seen with the string of expressibility results we get for almost all our different settings. We are also able to show how we would go about lifting the ideas with minimal tweaks to our vocabulary to be able to reap the powers of expressibility granted to us by the logic chosen.

The chapter should serve as a work bridging formal expressibility of reasoning in game theory and allied areas like fair division and voting systems.

4.1 Introduction

A logical study of game theory aims at exposing the assumptions and reasoning that underlies the basic concepts of game theory. This involves the study of individual, rational, strategic decision making between presented alternatives (in the non-cooperative setting).

The idea of **improvement graph** 2.1 is closely connected to optimizing the 'best' option in light of one's information. Improvement dynamics operates at a more operational, computational level, and reasoning about it can be seen as reasoning at the level of **computations searching for equilibria**.

In this sense, logic is seen as a succinct language for describing computational structure, rather than as a deductive system of reasoning by agents. In spirit, the role of such logics is similar to that of logics in **descriptive complexity theory**.

If we were to talk of the descriptive complexity of game theoretic equilibrium notions, it would need to account for the implicit improvement dynamics embedded in the solution concept.

Interestingly, contexts such as fair division and voting in social choice theory, embed such improvement dynamics as well.

When we aggregate individual choices or preferences into social choices / preferences, or decide on social action (like resource allocation) based in individual preferences, once again we see implicit improvement dynamics.

If a particular profile of voter preferences yields a specific electoral outcome, one can consider a voter announcing a revised (and altered) preference to force a different outcome.

Two agents might exchange their allocated goods to move to a new allocation, if they perceive advantage in doing so. Again, these can be seen as offers and counter-offers, perhaps leading to an equilibrium, or not. Some of these situations involve individual improvements, some (like pairs of agents swapping goods) involve coalitions, but they have the same underlying computational structure.

As it turns out, the properties of interest in those phenomena are seen to lie on the hard to compute complexity in the reasonable representations they are provided in.

4.1.1 Our Results and organisation of this chapter

To this purpose we use the data structure called the **improvement graph** 2.1.3 that would help in a uniform analysis of all of them.

We suggest that a suitable language for reasoning about this computational structure underlying games and social choice contexts can be found with variants of logics based on **monadic fixed-point logic**.

This is an extension of first order logic with monadic least fixed-point operators. In this, we follow the spirit of descriptive complexity, where extensions of first order logics describe complexity classes. Formulas offer concise descriptions of reasoning embedded in improvement dynamics.

We then have a model checking algorithm for verifying the truth and falsity of the properties expressed in our logic in the improvement graph model. The time complexity of such a procedure is polynomial in the size of the model.

One natural obstacle in reasoning about improvement dynamics is that the strategy space is exponential in the size of the game description. A natural question arises whether we can reason directly with game descriptions, and similarly in the other contexts, with the resource allocation system description, and so on.

In this context, we study a subclass of *separable games with priorities* 2.2.2, where we can show some restricted validities in the logic: for instance, we can assert that acyclicity of the game presentation ensures existence of Nash equilibrium. The *separable games with priorities* let's us restrict our attention from arbitrary general normal form games to a setting where the utilities are of a certain form (*pairwise separable*). This allows for a succinct representation of games.

In the above attempts we keep the logical description fixed to a finite number of players. But, in the game theory literature we hardly find payoff functions dependent on the number of players. Hence it makes sense that we try to emulate our results over games with arbitrary number of players. We introduce second order set quantifiers over the player sets, a minimal extension from the previous logical extension of mlfp we were working with and go about doing the same exercise as before. Such changes made to the setting raise different complexity issues and we mention them in the passing.

We end the chapter by lifting the results from the game setting to Fair Division and Voting.

4.1.2 Related Work

Various logical formalisms have been used in the literature to reason about games and strategies. Action indexed modal logics have often been used to analyse finite extensive form games where the game representation is interpreted as models of the logical language [?, ?, ?]. A dynamic logic framework can then be used to describe games and strategies in a compositional manner [?, ?, ?] and encode existence of equilibrium strategies [?].

Alternating temporal logic (ATL) [?] and its variants [?, ?, ?] constitute a popular framework to reason about strategic ability in games, especially infinite game structure defined by unfoldings of finite graphs. These formalisms are useful to analyse strategic ability in terms of existence of strategies satisfying certain properties (for example, winning strategies and equilibrium strategies). Some of these logical formalisms are also able to make assertions about partial specifications that strategies have to conform to in order to constitute a stable outcome.

In this work we suggest a framework to reason about the dynamics involved in iteratively updating strategies and to analyse the resulting convergence properties. [?] consider dynamics in reasoning about games in the same spirit as ours and describe it in fixed-point logic. But crucially, the dynamics is on iterated announcements of players' rationality, and belief revision in response to it. Moreover, they discuss extensive form games rather than normal form games. However, they do advocate the use of the fixed-point extension of first order logic for reasoning about games.

4.2 Working on the model of Improvement graphs

Our preliminaries are already worked out in Chapters 2 and 3. In particular you can find out about the logic and its motivation in Section These need to be filled in when the above chapters get written.

Monadic least fixed point logic (MLFP) is an extension of first-order logic which is well studied in finite model theory [?]. It is a restriction of first order logic with least fixed point in which only unary relation variables are allowed. MLFP is an expressive logic for which, on finite relational structures, model checking can be solved efficiently [?].

It is also known that MLFP is expressive enough to describe various interesting properties of games on finite graphs.

MLFP can also naturally describe transitive closure of a binary relation which makes it an ideal logical framework to analyse the dynamics involved in updating strategies and its convergence properties.

When α is a formula with one first order free variable, $C_x\alpha \leq k$ asserts that the number of elements in the domain satisfying α is at most k. Clearly, this is expressible in first order logic with equality, but at the expense of succinctness.

In the literature on first order logic with arithmetical predicates [**?**], it is customary to consider two sorted structures to distinguish between domain elements and the counts. When we deal with the *Improvement Graph* whose domain elements are always profiles, there is no need for such caution. In subsequent sections when we start focussing on *pairwise separable game with priorities* we take into consideration the different sorts in our disposal.

We start off with the definition of the syntax for the MLFPC logic.

4.2.1 MLFPC Syntax

Let σ be a first order relational vocabulary. Let $(S_i)_{i\in\mathbb{N}}$ be a sequence of monadic relation symbols, such that for each i, $S_i \notin \sigma$. These are the second order fixed-point variables of the logic.

The set of all MLFPC formulas, Φ_{MLFPC}, is defined inductively as follows, where $fv^1(\alpha)$ = the set of first order free variables in α; $fv^2(\alpha)$ = the set of all relation symbols S_i occurring in α; $fv(\alpha) = fv^1(\alpha) \cup fv^2(\alpha)$.

- An MLFPC atomic formula if of the form $R(x_1, \dots, x_k)$, or $x = y$ or $S_i(x)$ where R is a k-ary relation symbol in σ, S_i is a second order fixed-point variable.

- If α, β are MLFPC formulas then so are, $\sim \alpha$, $\alpha \wedge \beta$ and $\alpha \vee \beta$.

  $fv(\sim \alpha) = fv(\alpha)$ and $fv(\alpha \wedge \beta) = fv(\alpha \vee \beta) = fv(\alpha) \cup fv(\beta)$.

- If α is a MLFPC formula, $x \in fv^1(\alpha)$ and $k \in \mathbb{N}$, then so are $\exists x\alpha$, $\forall x\alpha$ and $C_x\alpha \leq k$.

  $fv(\exists x\alpha) = fv(\forall x\alpha) = fv(C_x\alpha \leq k) = fv(\alpha) \setminus \{x\}$.

- If α is an MLFPC formula, $S_i \in fv^2(\alpha)$, $x \in fv^1(\alpha)$, and $u \notin fv^1(\alpha)$ and S_i occurs positively in α, then $[\mathbf{lfp}_{S_i,x}\ \alpha](u)$ is an MLFPC formula.

  $fv([\mathbf{lfp}_{S_i,x}\ \alpha](u)) = fv(\alpha) \setminus \{S_i, x\}\ \cup \{u\}$.

The restriction to positive second order variables in the lfp operator is essential to provide an effective semantics to the logic. It is a standard way of ensuring monotonicity, given that we do not have an effective procedure to test whether a given first-order formula is monotone on the class of finite σ-structures [**?**].

It should be noted that the use of positive second order variables in no way restricts us to contexts where equilibria are guaranteed to exist.

Equilibria are given by graph properties, and these variables allow us to collect sets of vertices monotonically. Also note that in a formula $[\mathbf{lfp}_{S_i,x}\ \alpha](u)$ there can be free variables other than x (these are often called parameters).

It is well known that it is possible to remove the parameters in the formula by increasing the arity of the fixed-point variables (see for instance, [**?**][Lemma 1.28]).

4.2.2 MLFPC Semantics

To interpret formulas, we extend σ-structures with interpretations for the free first order and second order variables (the latter from the given sequence $(S_i)_{i \in \mathbb{N}}$). Let $\mathfrak{A}$ be a σ-structure, which has domain A. The notion

$$\mathfrak{A}, u_1 \mapsto a_1, \ldots u_m \mapsto a_m, S_1 \mapsto A_1, \ldots S_n \mapsto A_n \models \alpha(u_1, \ldots, u_m, S_1, \ldots, S_n)$$

is defined in the standard fashion, where $a_i \in A$ and $A_j \subseteq A$, $u_i \in fv^1(\alpha)$ and $S_j \in fv^2(\alpha)$. We abbreviate this by: $\mathfrak{A}, \vec{a}, \vec{A} \models \alpha(\vec{u}, \vec{S})$. To simplify notation, we also use ρ to denote the interpretation, where, for every first order variable u, $\rho(u) \in A$ and for every second order variable S_i, $\rho(S_i) \subseteq A$. The semantics are then given as follows.

- $\mathfrak{A}, \vec{a}, \vec{A} \models R(x_1, \ldots, x_k)$ iff the tuple $\vec{a} \in R^{\mathfrak{A}}$.
- $\mathfrak{A}, \vec{a}, \vec{A} \models S_j(x_i)$ iff $a_i \in A_j$.
- $\mathfrak{A}, \vec{a}, \vec{A} \models x_i = x_j$ iff $a_i = a_j$.
- $\mathfrak{A}, \vec{a}, \vec{A} \models \sim \alpha$ iff $\mathfrak{A}, \vec{a}, \vec{A} \not\models \alpha$.
- $\mathfrak{A}, \vec{a}, \vec{A} \models \alpha \vee \beta$ iff $\mathfrak{A}, \vec{a}, \vec{A} \models \alpha$ or $\mathfrak{A}, \vec{a}, \vec{A} \models \beta$.
- $\mathfrak{A}, \vec{a}, \vec{A} \models \exists x.\alpha$ iff for some $a \in A$, $\mathfrak{A}, \vec{a}, x \mapsto a, \vec{A} \models \alpha$.
- $\mathfrak{A}, \vec{a}, \vec{A} \models C_x\alpha \leq k$ iff $|\{a \in A \mid \mathfrak{A}, \vec{a}, x \mapsto a, \vec{A} \models \phi\}| \leq k$.
- $\mathfrak{A}, \vec{a}, u \mapsto a, \vec{A} \models [\mathbf{lfp}_{S_i,x}\alpha](u)$ iff $a \in \mathrm{lfp}(f_\alpha)$ where for any formula β with $x \in fv^1(\beta)$, $f_\beta : \wp(A) \mapsto \wp(A)$ is defined by: $f_\beta(B) = \{a \in A \mid \mathfrak{A}, S_i \mapsto B, x \mapsto a \models \beta(x)\}$.

The lfp quantifier induces an operator on the powerset of elements on the structure ordered by inclusion. The positivity restriction ensures that the operator is monotone and hence least fixed-points exist.

4.2.3 Illustration

We need to first illustrate a working example of how the lfp operator works so as to make our readers comfortable with the logic. In the following, we present a formula on a particular directed graph in the example below and iterate over the fixed point computation highlighting the effects on the graph as well. The yellow color on the vertices are supposed to highlight the vertices chosen in a particular round of the fixed point computation. The green edges reaching out to further vertices denote the vertices that satisfy the formula and the red ones are to show when the formula isn't satisfied. Hopefully the sequence of diagrams will be helpful in relaying the operational dynamics of the logic we use in this paper.

Given the graph below 4.1 we want to be computing whether $[\mathbf{lfp}_{\{S,x\}}\ \psi](c)$ and $[\mathbf{lfp}_{\{S,x\}}\ \psi](g)$ is true or false, where $\psi(S,x) \;=\; (x=a) \vee \exists y(S(y) \wedge (C_zE(y,z) \leq 2) \wedge E(y,x))$, given below as the sample formula! The fixed point operator f operates on a sequence of sets beginning from the $\emptyset$ and iterating over $f(\emptyset) \rightarrow f(f(\emptyset)) \rightarrow \ldots$ till it reaches a fixed point where $f^n(\emptyset) = f^{n+1}(\emptyset)$, which happens to be our fixed point that the induction operator S stores.

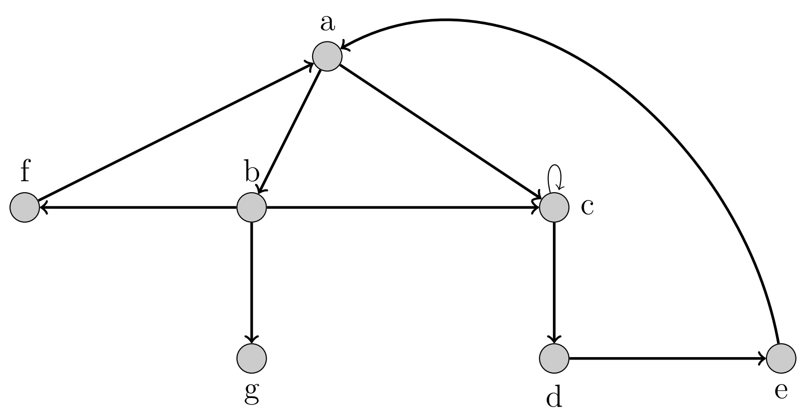

Figure 4.1: A Graph over which the LFP formula computation will be done

Sample formula over the directed graph

$$\psi(S,x) \;=\; (x=a) \vee \exists y(S(y) \wedge (C_zE(y,z) \leq 2) \wedge E(y,x))$$

The intended meaning of this formula is either $x = a$ or x is some vertex connected to a vertex already present in the set S which has the property that it is connected to at max 2 other vertices.

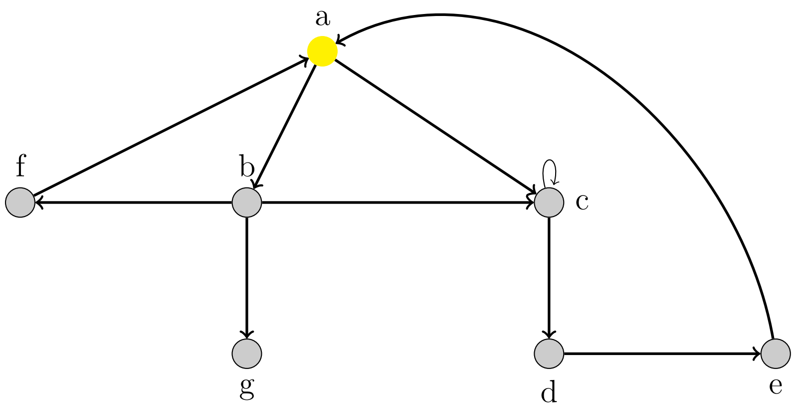

Figure 4.2: The first iteration

Computing the fixed point

In the first iteration when S is mapped to the $\emptyset$, only $x = a$ is a solution to f_ψ. It is coloured yellow to mark it. Therefore, $f_\psi(\phi) = \{a\}$

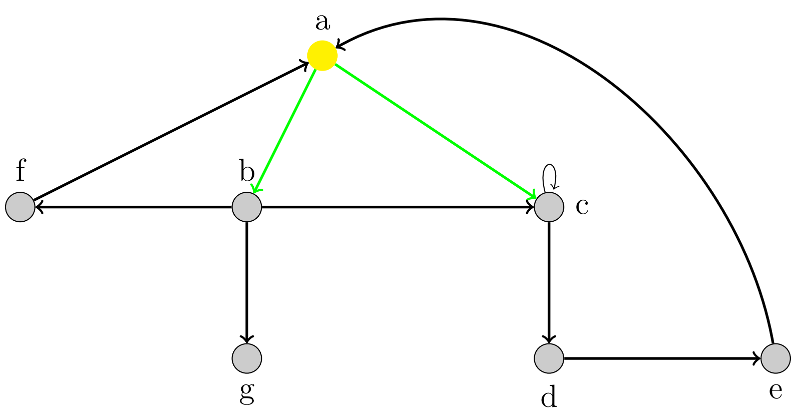

Figure 4.3: Second iteration

Computing the fixed point

In the second iteration we can see that apart from $x = a$, two other vertices connected to vertex a, ie, b, c are the solutions as well. So we have $f_\psi(\{a\}) = \{a, b, c\}$

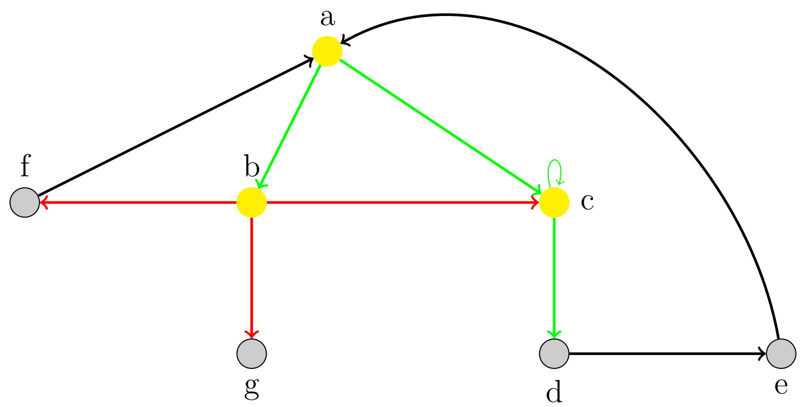

Figure 4.4: The third iteration

Computing the fixed point

In the third iteration we can see that apart from a and the solutions due to a, which are b and c, we have a new solution d due to the vertex c belonging to S. d is a vertex that is connected to c which belongs to S, and c has the property that it is connected to at most two other vertices, namely c and d. We also show that even though b is in S, it doesn't give us any new solution because it is connected to three vertices. Shown with red edges. Therefore, $f_\psi(\{a, b, c\}) = \{a, b, c, d\}$

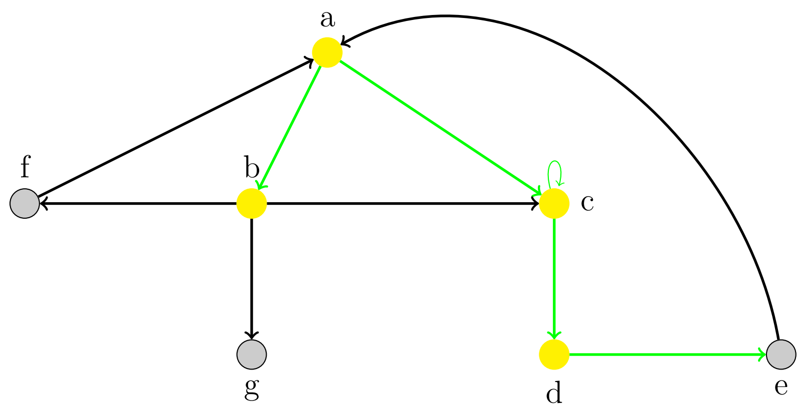

Figure 4.5: Fourth iteration

Computing the fixed point

Similarly, by now we can see that in the fourth iteration we get a new vertex e due to d and thus, $f_\psi(\{a,b,c,d\}) = \{a,b,c,d,e\}$

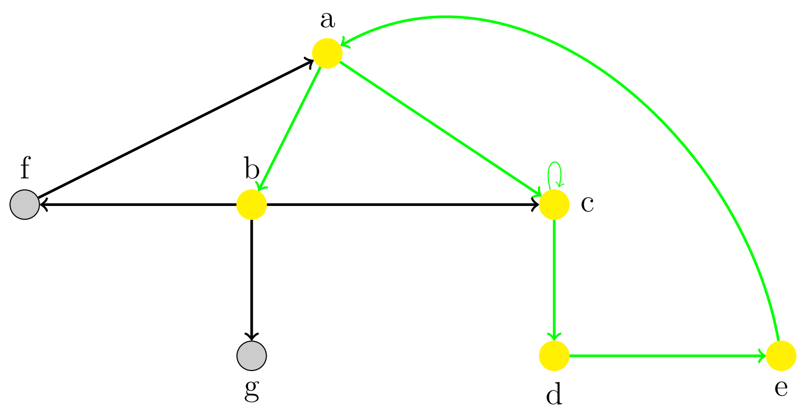

Figure 4.6: Fifth iteration - The fixed point is reached!

Computing the fixed point

And, finally we reach the fixed point of the computation. $f_\psi(\{a,b,c,d,e\}) = \{a,b,c,d,e\}$

Some formula examples

Therefore we can now answer the following lfp formulas.

- $[\mathbf{lfp}_{\{S,x\}}\ \psi](c)$ is true.
- $[\mathbf{lfp}_{\{S,x\}}\ \psi](g)$ is false.

4.2.4 Expressivity

Since the models of interest are improvement graphs, first order variables range over nodes in the graph, monadic second order variables range over subsets of nodes and the vocabulary consists of binary relations E_u, where $u \subseteq [n]$. When $|u| = 1$ and

$u = \{i\}$, we will simply write the relation as E_i. We write formulas of the form $E(x, y)$ to denote $\bigvee_{i \in [n]} E_i(x, y)$.

We now write special formulas that will be of interest in the sequel.

- $sink(x) = \forall y. \sim E(x, y)$
- $trap(S, x) = \forall y.(E(y, x) \implies S(y))$
- $acyclic = \forall u.[\mathbf{lfp}_{S,x}\ trap](u)$
- $reach(S, x) = sink(x) \vee \exists y(E(y, x) \wedge S(y))$
- $weakly - acyclic = \forall u.[\mathbf{lfp}_{S,x}\ reach](u)$

Now consider the formulas interpreted over improvement graphs of normal form games. *sink* refers to the set of sink nodes, and these are exactly the Nash equilibria of the associated game. The sentence *acyclic* is true exactly when the improvement graph is acyclic and hence such games have the finite improvement property (as every improvement path is finite). To see that the sentence captures acyclicity, note the action of the lfp operator: at the zeroth iteration, we get all nodes with in-degree 0; we then get all nodes which have incoming edges from nodes whose in-degree is 0; and so on. Eventually it collects all nodes through which no path leads to a cycle. Since the sentence applies to every node, we infer that the graph does not contain any cycle.

For weak acyclicity, we require that there exists a finite improvement path starting from every node. Again the lfp operator picks up sink nodes at the zeroth iteration, then all nodes that have a sink node as successor, and so on. Eventually it collects all nodes that start finite improvement paths. The sentence asserts that every node has this property.

We can easily generalize these ideas to improvement graphs where the edge relation denotes deviation by a subset of players, rather than a single player. This leads us naturally to a *concurrent setting*, and we get what are called k-equilibria in *coordination games*.

Let s and s' be strategy profiles in a normal form game, $u \subseteq [n]$. Define $s \rightarrow_u s'$ when $s[j] = s'[j]$ for $j \notin u$, and for all $i \in u$, $s'[i] \succ_i s[i]$. That is, with the choices of the other agents fixed, the coalition of agents in u can coordinate their choices and deviate to get a better outcome. Thus k-equilibria are nodes from which no coalition of at most k agents can profitably deviate. We can then define a coalitional k improvement path, where at each step a coalition of at most k agents deviate, which leads us further to a k-FIP. [**?**] shows that a class of *uniform* coordination games has this property.

- $sink_u(x) = \forall y. \sim E_u(x, y)$
- $sink_k(x) = \bigvee_{u \subseteq [n], |u| \leq k} sink_u(x)$
- $k - edge(x, y) = \bigvee_{u \subseteq [n], |u| \leq k} E_u(x, y)$
- $k - trap(S, x) = \forall y.(k - edge(y, x) \implies S(y))$
- $k - FIP = \forall u.[\mathbf{lfp}_{S,x}\ k - trap](u)$

Note that the disjunctions are large, exponential in k. Since we have a counting operator, we could add further structure to nodes, prising out the individual strategies of players and then use the counting quantifier over these to get a succinct formula, linear in k.

In general, we might want to specify reachability of a set of distinguished nodes satisfying some property. For instance, we might want to assert that a particular node is reachable from any node. Note that the lfp operator is sufficient to specify reachability of such nodes.

- $reach_\phi(S, x) = \phi(x) \vee \exists y(E(y, x) \wedge S(y))$

- $\phi - reachable = \forall u.[\mathbf{lfp}_{S,x}\ reach](u)$

Further, the counting quantifier can give us interesting relaxations of improvement dynamics. For instance consider the following specifications:

- $reach(S, x) = sink(x) \lor \exists y(E(y, x) \land S(y))$
- $path - count = C_u([\mathbf{lfp}_{S,x}\ reach](u)) < 5$

This specifies that at most 5 nodes have finite improvement paths originating from them. Here the lfp operator is in the scope of the counting quantifier. In the following specification, we have them the other way about.

- $count - trap(S, x) = C_y E(y, x) < k \implies (\forall z.(E(z, x) \implies S(z)))$
- $special = \forall u.[\mathbf{lfp}_{S,x}\ count - trap](u)$

4.2.5 Model Checking Algorithm

The model checking problem for MLFPC is as follows: given a finite structure $\mathfrak{A}$, a MLFPC formula ϕ along with an interpretation ρ decide whether $\mathfrak{A}, \rho \models \phi$. We show that the recursive procedure given below solves the model checking problem for MLFPC efficiently (in the size of the structure).

MC$(\mathfrak{A}, \rho, \phi)$

Input. An MLFPC formula ϕ, a σ structure $\mathfrak{A}$ and an interpretation ρ.

Output. If $\mathfrak{A}, \rho \models \phi$ then 1 and 0 otherwise.

switch *type of* ϕ

1. **case** ϕ is an atomic formula

   if $\mathfrak{A} \models \phi[\rho]$ **return** 1 **else return** 0

2. **case** $\phi = \sim \beta$

   if $\mathrm{MC}(\mathfrak{A}, \rho, \beta)$ **return** 0 **else return** 1

3. **case** $\phi = \beta_1 \vee \beta_2$

   Let $\rho_1 \coloneqq \rho|_{fv(\beta_1)}$, $\rho_2 \coloneqq \rho|_{fv(\beta_2)}$

   if [$\mathrm{MC}(\mathfrak{A}, \rho_1, \beta_1)$ or $\mathrm{MC}(\mathfrak{A}, \rho_2, \beta_2)$] **return** 1 **else return** 0

4. **case** $\phi = \exists y \beta$

   for $a \in A$

    if $\mathrm{MC}(\mathfrak{A}, \rho[x \mapsto a], \beta)$ **return** 1

   return 0

5. **case** $\phi = C_x \beta \leq k$

   count $= 0$

   for $a \in A$

    if $\mathrm{MC}(\mathfrak{A}, \rho[x \mapsto a], \beta)$ count ++

   if count $\leq k$ **return** 1 **else return** 0

6. **case** $\phi = [\mathbf{lfp}_{S_i,x}\ \beta](u)$

    iter $= \emptyset$, $f_\beta = \emptyset$

    do

     iter $= f_\beta$

     $f_\beta = \{b \in A \mid \mathrm{MC}(\mathfrak{A}, \rho[S_i \mapsto f_\beta, x \mapsto b], \beta)\}$

    while $f_\beta \neq$ iter

    if $\rho(u) \in f_\beta$ **return** 1 **else return** 0

For a formula $\phi \in \Phi_{MLFPC}$, we denote :

Table 4.1: Definitions of the properties of a formula, ϕ, of MLFPC

$cl(\phi)$	the set of all subformulae of ϕ
$width(\phi)$	$= \max \{|fv(\psi)| : \psi \in cl(\phi)\}$. which is the width of ϕ
$qr(\phi)$	the maximal nesting depth of quantifiers in ϕ.

Formally $qr(\phi)$ is defined as follows:

- If ϕ is atomic, then $qr(\phi) = 0$.
- $qr(\sim \phi) = qr(\phi)$.
- $qr(\phi_1 \vee \phi_2) = qr(\phi_1 \wedge \phi_2) = \max(qr(\phi_1),\ qr(\phi_2))$.
- $qr(\exists x.\phi) = qr(\forall x.\phi) = qr(C_x.\phi) = qr([\mathbf{lfp}_{S_i,x}\ \phi](u)) = qr(\phi) + 1$.

Theorem 4.2.1: Correctness of the Model checking Algorithm

Given $\phi \in \Phi_{MLFPC}$, a σ-structure $\mathfrak{A}$ and an interpretation ρ: $\mathfrak{A}, \rho \models \phi$ iff algorithm MC returns 1 in time $|A|^{O(qr(\phi))} + O(|\phi|)$ and space $O(\log|\phi| + width(\phi)\log|A|)$.

Proof. We first argue the correctness of the procedure by induction on the structure of the formula ϕ and then analyse the complexity.

Base case. When ϕ is an atomic formula, the procedure does a direct lookup given the structure and the interpretation. Correctness follows from the semantics.

Induction step.

- For boolean connectives, the correctness follows from induction hypothesis and the semantics.

- $\alpha \equiv \exists x\beta$: $\mathfrak{A}, \sigma \models \exists x\beta$ iff by semantics, $\exists a \in A$ such that $\mathfrak{A}, \rho[x \mapsto a] \models \alpha$. By induction hypothesis, for each $a \in A$, we have $\mathfrak{A}, \rho[x \mapsto a] \models \beta$ iff $\mathrm{MC}(\mathfrak{A}, \rho[x \mapsto a], \beta)$ returns 1. By case 4, in MC it follows that the algorithm returns 1 iff $\exists a \in A$ such that $\mathfrak{A}, \vec{a}, x \mapsto a, \vec{A} \models \alpha$.

- $\phi \equiv C_x\beta \leq k$: $\mathfrak{A}, \sigma \models C_x\beta \leq k$ iff by semantics, $|\{a \in A \mid \mathfrak{A}, \rho[x \mapsto a] \models \beta\}| \leq k$. By induction hypothesis, for each $a \in A$, we have $\mathfrak{A}, \rho[x \mapsto a] \models \beta$ iff $\mathrm{MC}(\mathfrak{A}, \rho[x \mapsto a], \beta)$ returns 1. In the algorithm MC, by definition of case 5, the algorithm maintains a local variable *count* which registers the number of different interpretations of variable x for which $\mathfrak{A}, \rho[x \mapsto a] \models \beta$. Thus from the definition of the algorithm it follows that MC returns 1 iff $|\{a \in A \mid \mathfrak{A}, \rho[x \mapsto a] \models \beta\}| \leq k$.

- $\phi \equiv [\mathbf{lfp}_{S_i,x}\ \beta](u)$: By semantics, $\mathfrak{A}, \rho \models [\mathbf{lfp}_{S_i,x}\beta](u)$ iff $\rho(u) \in \mathrm{lfp}(f_\beta)$ where $f_\beta(B) = \{a \in A \mid \mathfrak{A}, S_i \mapsto B, x \mapsto a \models\ \beta(x)\}$. By induction hypothesis, $\mathfrak{A}, \rho[S_i \mapsto B, x \mapsto a] \models\ \beta(x)$ iff $\mathrm{MC}(\mathfrak{A}, \rho[S_i \mapsto f_\beta, x \mapsto a], \beta)$ returns 1. Now consider case 6, in the algorithm MC. The *do-while* loop computes the least fixed point $f_\beta\ : \wp(A) \mapsto \wp(A)$. The *if* conditional statement checks if $\rho(u) \in f_\beta$. Thus from the definition of the algorithm it follows that MC returns 1 iff $\mathfrak{A}, \rho \models [\mathbf{lfp}_{S_i,x}\beta](u)$.

Complexity analysis. We analyse the running time by induction on the structure of the formula ϕ. For atomic formulas, we perform a direct lookup. Thus the time taken is $O(|A|^k)$ where k is the maximum arity of a predicate symbol in σ. The boolean connectives are straightforward. For $\phi \equiv \exists\psi$, $\phi \equiv \forall\psi$ and $\phi \equiv C_x\psi$, if $\mathfrak{A}, \rho \models \psi$ can be decided in time $O(|A|^p)$ then $\mathfrak{A}, \rho \models \phi$ can be decided in time $O(|A|^{p+1})$. And for $\phi \equiv \mathbf{lfp}_{S_i,x}\ \psi$, whether $\mathfrak{A}, \rho \models \phi$ can be decided in time $O(|A|^{2p})$. Thus the total time taken is $|A|^{O(qr(\phi))} + O(|\phi|)$.

To bound the space required, note that algorithm needs to maintain a pointer to the current subformula of ϕ and to store the current interpretation, which needs

$fv(\phi) \times \log|A|$ bits. Hence the space needed by the algorithm is $O(\log|\phi| + width(\phi)\log|A|)$.

$\square$

The above result implies that for formulas with bounded quantifier depth, the model checking procedure runs in time polynomial in both the size of the structure and the formula. All the formulas expressing interesting properties in games and social choice theory that we have presented in this paper have quantifier depth 1. Therefore, all these properties can be verified in polynomial time.

In the context of improvement graphs, if there are n agents and at most m choices for each agent, the size of the associated improvement graph is $O(m^n)$. Since it is possible to have a compact representation for certain subclasses of strategic form games, for instance, polymatrix games [**?**], the size of the improvement graph structure can be exponential in the representation of the game. Thus the model checking procedure, while polynomial on the size of the underlying improvement graph, can in principle, be exponential in the size of the game representation. This observation may not be very surprising since even for restricted classes of games like 0/1 polymatrix games, checking for the existence of Nash equilibrium is known to be NP-complete [**?**].

4.3 Working on the model of the games itself

While the improvement graph is a natural structure to reason about the dynamic properties of strategic interaction, explicitly representing the improvement graph has the disadvantage that the structure is exponential in the number of strategies in the underlying game. Reasoning about properties of games without explicitly constructing the improvement graph is therefore of obvious interest,

Strategic form games constitute a rich model allowing arbitrary utility functions over strategy profiles, in general, the representation of a game can be exponential in the number of strategies.

In this section we identify a subclass of strategic form games, known as priority separable games where the utility functions are restricted to be pairwise separable.

We show that the logical framework introduced in the section 4.2.1 can be effectively used to analyse such games by interpreting formulas on the (compact) game description rather than the associated improvement graph.

We thus turn our focus to Priority separable games mentioned in 2.2.2. The definitions are given in 2.2.2.

4.3.1 MLFPC and priority separable games

Instead of considering improvement graphs, we take the description of the priority separable game as the model along with a relational vocabulary for the logic. We present here the relational vocabulary.

Vocabulary for Priority Separable Games

The underlying domain consists of three components: the player set N, the strategy set T and the outcome set O. In addition, we fix the following relations to interpret the game model:

- Monadic relation P which identifies the players.
- Monadic relation O which identifies the outcomes.
- Monadic relations $S_i \subset T$ indicating the strategy belonging to the particular player.

- Relation $\mathcal{S} \subseteq N \times T$ that specifies which strategy belongs to the strategy set of a player.

- Binary relation $E \subseteq N \times N$ that specifies the neighbourhood for each player.

- Relation $u \subseteq N \times T \times T \times O$ that specifies the pairwise payoffs.

- Binary relations $\ll_i$ specifying the ordering in the outcome set for each of the player.

- Binary relation $\triangleright_i$ that specifies the priority ordering for each player $i \in N$.

Expressivity

Let σ denote the language consisting the above vocabulary. We call a σ-structure *priority separable* if the interpretation on relational symbols confirm to the above definition. Thus there is a correspondence between priority separable games and priority separable σ-structures.

We first show that we can write a formula to characterise one step improvement of a player. We use $\vec{x}$ and $\vec{y}$ to denote tuple of strategies, $\vec{a}$ and $\vec{b}$ denotes the corresponding tuple of outcomes and p, v, w as player variables. We make use of the auxiliary formulas given below. Given two outcome vectors $\vec{a}$ and $\vec{b}$ we can compare the outcome for player i using the formulas $\psi_1(p, v, \vec{x}, \vec{a}, \vec{y}, \vec{b})$ and $\psi_2(p, \vec{x}, \vec{a}, \vec{y}, \vec{b})$.

- $\text{chkStrat}(\vec{x}) = \bigwedge_j S_j(x_j)$ - states that $\vec{x}$ is a valid strategy profile.

- $\text{chkOut}(p, \vec{x}, \vec{a}) = \bigvee_{i \in N} (\mathcal{S}(p, x_i) \implies (\bigwedge_{j \neq i} u(p, x_i, x_j, a_j)))$ - states that the outcome vector $\vec{a}$ is consistent with the pairwise utility relation u and the tuple of strategies $\vec{x}$.

- $\text{first}(p, v, \vec{x}) = E(v, p) \wedge \bigvee_{i \in N} (\mathcal{S}(p, x_i) \implies \forall w(E(w, p) \implies w \triangleright_i v))$ - states that $z \in R(p)$ is the player in the neighbourhood of p with the highest priority.

- $\text{priority}_<(p, \vec{x}, w, v) = \bigvee_{i \in N} (\mathcal{S}(p, x_i) \implies w \rhd_i v)$ - states that u, v are players in the neighbourhood of i and u has higher priority than v.

- $\text{nxt}(p, w, v, \vec{x}) = \sim \exists v'(\text{priority}_<(p, \vec{x}, w, v') \wedge \text{priority}_<(p, \vec{x}, v', v))$ states that in the priority preference for player p the player w comes before v.

- $\text{1-step}(p, \vec{x}, \vec{y}) = \text{chkStrat}(\vec{x}) \wedge \text{chkStrat}(\vec{y}) \wedge \bigvee_{i \in [n]} (\mathcal{S}(p, x_i) \implies x_i \neq y_i \wedge \bigwedge_{j \neq i} x_j = y_j)$ - states that $\vec{x}, \vec{y}$ are tuple of strategies and the only difference between them is in the strategy of player p.

- $\psi_1(p, u, \vec{x}, \vec{a}, \vec{y}, \vec{b}) = \bigvee_{i \in N} [(\mathcal{S}(p, x_i) \wedge \mathcal{S}(u, x_j)) \implies a_j \ll_i b_j]$.

- $\psi_2(p, \vec{x}, \vec{a}, \vec{y}, \vec{b}) = \bigvee_{i \in N} [\mathcal{S}(p, x_i) \implies a_i = b_i]$.

We can then write the formula $\text{Dev}(p, \vec{x}, \vec{a}, \vec{y}, \vec{b})$ that states that $\vec{y}$ is an improvement for player p from $\vec{x}$ by a 1-step deviation.

$$
\begin{aligned}
\text{Dev}(p, \vec{x}, \vec{y}) = {} & \exists \vec{a}, \vec{b} \ (\bigwedge_i (O(a_i) \wedge O(b_i)) \\
& \wedge \ \text{1-step}(p, \vec{x}, \vec{y}) \\
& \wedge \text{chkOut}(p, \vec{x}, \vec{a}) \wedge \text{chkOut}(p, \vec{y}, \vec{b}) \\
& \wedge \exists v([\mathbf{lfp}_{M,w}\alpha](v, p, \vec{x}, \vec{a}, \vec{y}, \vec{b}) \\
& \quad \wedge \psi_1(p, v, \vec{x}, \vec{a}, \vec{y}, \vec{b}))) \\
\alpha(M, w, p, \vec{x}, \vec{a}, \vec{y}, \vec{b}) = {} & P(p) \wedge P(w) \wedge \text{first}(p, w, \vec{x}) \\
& \vee \exists v[M(v) \wedge \text{nxt}(p, w, v, \vec{x}) \\
& \quad \wedge \psi_2(v, \vec{x}, \vec{a}, \vec{y}, \vec{b})]
\end{aligned}
\tag{4.1}
$$

Now, existence of Nash equilibrium can be characterised using the following formula:

- $\text{Imp}(\vec{x}, \vec{y}) = \exists p(P(p) \wedge (\ \exists \vec{b}, \ \exists \vec{a}(\bigwedge_i (O(a_i) \wedge O(b_i)) \ \wedge \ \text{Dev}(p, \vec{x}, \vec{a}, \vec{y}, \vec{b}))))$.

- $\mathrm{NE}(\vec{x}) = \forall \vec{y} \sim \mathrm{Imp}(\vec{x}, \vec{y})$.

- $\mathbb{G} \models \exists \vec{x}\ \mathrm{NE}(\vec{x})$

Restricted Validities

It is easy to see that priority separable games need not always have a (pure) Nash equilibrium. 2 Thus, a natural question is to identify subclasses where Nash equilibrium is guaranteed to exists.

The main aspect behind priority separable games is the fact that the final utility for each player constitutes a combination of the outcomes in each pairwise interaction.

Thus to identify interesting subclasses, we can restrict both the graphical structure that represents the neighbourhood relation as well as the pairwise utility relation.

In this section, we show that the logical framework can be effectively used to describe such restrictions. We show that both properties describing the neighbourhood relation as well as restrictions on the pairwise utility relation can be characterised over the class of priority separable σ-structures.

Below we provide a few formulas characterising properties of the neighbourhood relation in the underlying game.

- $\mathrm{trap}(M, w) = \forall v(P(v) \wedge P(w) \wedge E(v, w) \implies M(v))$.

- $\mathrm{acyclic} = \forall w(P(w) \wedge [\mathbf{lfp}_{M,u}\ \mathrm{trap}](w))$. As we saw in section 4.2.4, this formula captures the property of acyclicity. By using the monadic relation P, we restrict the domain to the set of players. Thus this formula characterises the absence of a cycle in the player graph (N, E).

- $\text{uniq}(p, v) = E(p, v) \wedge \forall w(P(w) \wedge E(p, w) \implies w = v)$ - characterises all nodes u which has a unique outgoing edge (to node v).

- $\text{scycle} = \forall p \exists v \exists w(P(p) \wedge P(v) \wedge P(w) \wedge \text{uniq}(p, v) \wedge \text{uniq}(w, p))$ - states that the underlying player graph (N, E) is a simple cycle.

Some properties of the utility function that can be characterised in the logic are given below:

- $\text{top}_i(o) = \forall a(O(a) \implies a \ll_i o)$ - states that outcome o is the most preferred outcome for player i.

- $\text{bot}_i(o) = \forall a(O(a) \implies o \ll_i a)$ - states that outcome o is the least preferred outcome for player i.

- $\text{coord}_i(\vec{x}) = \exists a \exists b(O(a) \wedge O(b) \wedge \text{top}_i(a) \wedge \text{bot}_i(b) \wedge S_i(x_i) \wedge (\bigwedge_{j \neq i}((E(j, i) \wedge x_i = x_j \implies u(i, x_i, x_j, a)) \wedge (\sim E(j, i) \vee x_i \neq x_j \implies u(i, x_i, x_j, b)))))$ - states that for each pairwise interaction, if player i coordinates its strategy with that of j (where $j \in R(i)$), then player i's utility is its most preferred outcome.

- $\text{anticoord}_i(\vec{x}) = \exists a \exists b(O(a) \wedge O(b) \wedge \text{top}_i(a) \wedge \text{bot}_i(b) \wedge S_i(x_i) \wedge (\bigwedge_{j \neq i}((E(j, i) \wedge x_i \neq x_j \implies u(i, x_i, x_j, a)) \wedge (\sim E(j, i) \vee x_i = x_j \implies u(i, x_i, x_j, b)))))$ - captures anti-coordination. It states that if player i chooses a strategy different from its neighbour j then player i's utility in the pairwise interaction is its most preferred outcome.

- $\text{coordall}_i = \forall \vec{x}(\text{chkStrat}(\vec{x}) \implies \text{coord}_i(\vec{x}))$.

- $\text{anticoordall}_i = \forall \vec{x}\,(\text{chkStrat}(\vec{x}) \implies \text{anticoord}_i(\vec{x}))$.

- $\text{coord}_G = \bigwedge_{j \in N} \text{coordall}_j$ - states that the game is a coordination game.

- $\text{anticoord}_G = \bigwedge_{j \in N} \text{anticoordall}_j$ - states that the game is an anti-coordination game.

- $\text{dummy}_i = \bigwedge_{j \neq i} (\forall x_j \exists a \forall x_i (S_i(x_i) \wedge S_j(x_j) \wedge O(a) \implies u(i, x_i, x_j, a)))$ - states that the utility of player i is independent of his choice of strategy. That is, player i does not have any influence on his utility function in the pairwise interaction.

- $\text{dummy}_G = \bigwedge_{j \in N} \text{dummy}_j$ - states that the game is a dummy game for every player.

Theorem 4.3.1: Validities over the class of priority separable σ-structures.

The following are validities over the class of priority separable σ-structures.

1. $\varphi_1 = \text{acyclic} \implies \exists \vec{x}\ (\text{chkStrat}(\vec{x}) \wedge \text{NE}(\vec{x}))$.

2. $\varphi_2 = \text{scycle} \wedge \text{coord}_G \implies \exists \vec{x}(\text{chkStrat}(\vec{x}) \wedge \text{NE}(\vec{x}))$.

Proof. (1): Consider any structure $\mathfrak{A}$ such that $\mathfrak{A} \models \varphi_1$ (note that φ_1 is a sentence - it has no free variables) and suppose $\mathfrak{A} \models$ acyclic. By the argument given earlier, this implies that the player neighbourhood graph $G = (N, E)$ is acyclic. We need to show that there exists a tuple of strategies $\vec{x}$ such that $\forall \vec{y} \sim \text{Imp}(\vec{x}, \vec{y})$. In other words, $\vec{x}$ is a tuple of strategies such that no player has an improvement from $\vec{x}$. We argue this by showing a stronger property: for priority separable games whose underlying player neighbourhood graph is acyclic, starting at any arbitrary tuple of strategies $\vec{z}$ there exists a finite improvement path terminating in the tuple of strategies $\vec{x}$ in the induced improvement graph (which implies that no player has an improvement from $\vec{x}$). Suppose the neighbourhood graph (N, E) is acyclic. Let λ be the ordering of players corresponding to the topological sorting of (N, E). Consider an arbitrary strategy profile $\vec{z}$, we construct a finite improvement path σ starting at

$\vec{z}$ as follows: we use the ordering λ and update the strategy of players to their best response according to this ordering. Note that, by definition of λ, when a player j is chosen in the above procedure, it is guaranteed that the strategy s_k for all $k \in R(j)$ is defined. Clearly σ is finite by definition. Suppose the last strategy profile in σ is $\vec{x}$ and there is a player i and strategy profile $\vec{x}'$ such that $\mathfrak{A} \not\models \phi(p, \vec{x}, \vec{a}, \vec{x}', \vec{b})$. But this contradicts the fact that the choice of x_i was the best response when the strategy of player i was updated. Note that for all $j \in R(i)$, j occurs before i in the ordering λ.

(2): Consider any structure $\mathfrak{A}$ such that $\mathfrak{A} \models \varphi_2$ and suppose $\mathfrak{A} \models \text{scycle} \wedge \text{coord}_G$. This implies that $\mathfrak{A} \models \forall u \exists v \exists w (P(u) \wedge P(v) \wedge P(w) \wedge \text{uniq}(u, v) \wedge \text{uniq}(w, u))$. Therefore, for every player i in the player graph there is exactly one incoming and one outgoing edge. We also have that $\mathfrak{A} \models \text{coord}_G$ and therefore, for all players i, there is an outcome o such that $\text{top}_i(o)$ holds and for all $j \in R(i)$ utility of i in the pairwise interaction with j is o when $x_i = x_j$. Again, we show that in this case, starting at any arbitrary tuple of strategies $\vec{z}$ there exists a finite improvement path terminating in the tuple of strategies $\vec{x}$ in the induced improvement graph (which implies that no player has an improvement from $\vec{x}$). The argument is very similar to the one in [**?**][Lemma 6]. Since the underlying player graph forms a simple cycle, assume without loss of generality that it is of the form $1 \to 2 \cdots \to n-1 \to n \to 1$ (otherwise, we can simply relabel the players). Consider an arbitrary strategy profile $\vec{z}$, we construct a finite improvement path σ starting at $\vec{z}$ as follows: we perform two rounds of updates in the cyclic order starting at player 1. In the first round, we update players 1 to n in each step let them switch to their best response. Let x' be the strategy at the end of round 1, every player is playing its best response in x' except possible player 1 (since player n might have changed its strategy in round 1). In round two we let the player update again in the order $1, \ldots, n$. Note that in this round if a player updates then it update to the same strategy as its unique predecessor. Therefore player n does not update its strategy and we terminate in a strategy profile x such that $\mathfrak{A} \models \forall \vec{y} \sim \mathrm{Imp}(\vec{x}, \vec{y})$. $\square$

4.3.2 Generalising to Arbitrary Number of Players

In all these previous attempts we had fixed the number of players in a game. Thus we saw a vector of variables to represent strategy choices by each of the players.

In this section we try to further push our results by removing the constraint of a fixed number of players. We therefore, generalise our attempt at being able to logicise pure nash equilibria.

Allowing arbitrary number of players as part of our model gets us closer to describing the algorithmic problem statements on these pure equilibria games.

In this setting the number of players n becomes part of the input. Hence, now our model checking algorithm can be shown to be NP-hard by Theorem **??** .

The rest of this section will work on the same priority separable games and elicit the same kind of formulas for Nash equilibria. And finally we will also remark on our attempt at finding a tight complexity bound for the model checking procedure.

Vocabulary for generalised priority separable games

The definition of the priority separation games stays the same as in Section **??**.

What we need to do is instead change the vocabulary we worked with. The previous vocabulary basically required the number of players fixed because depending on the number of players we had an ensemble of relations indexed by each player.

In the current treatment we will go away from this need to index to a situation where we can relate a player variable with the (appropriately modified) similar relations as present before. We also need to realise the strategy profiles are second order structures in such a setting. They are functions from the player set N to the union of all possible strategies.

Strategy Profile A sequence $(p, s_p)_{p \in N}$ where each player p chooses a strategy $s_p \in S_p$. We will represent strategy profiles as second order binary variables **X, Y, Z**.

Modified Vocabulary We present a minimally modified vocabulary. The underlying domain continues to contain these three components: the player set N, the strategy set T and the outcome set O. We have the following relations to interpret the game model.

- Monadic relation P which identifies the players.
- Monadic relation O which identifies the outcomes.
- Monadic Relation $S \subset T$ indicating the strategy.
- Relation $\mathcal{S} \subseteq N \times T$ that specifies which strategy belongs to the strategy set of a player.
- Binary relation $E \subseteq N \times N$ that specifies the neighbourhood for each player.
- Relation $u \subseteq N \times T \times T \times O$ that specifies the pairwise payoffs.
- $\ll \subseteq N \times O \times O$ which basically is, $\ll (i, o_1, o_2) = \ll_i (o_1, o_2)$ when compared to the previous section 5 formulas. But while writing are formulas we will be using the syntactic sugar : $\ll_p$ as an infix operator. Example : $o_1 \ll_p o_2$ should be read as $\ll (p, o_1, o_2)$
- $\triangleright \subseteq N \times N \times N$ that specifies the priority ordering for each player $i \in N$. Basically it's $\triangleright(i, p_1, p_2) = \triangleright_i(p_1, p_2)$. Similar to the above scenario here also we will add the syntactic sugar $\triangleright_p$ as an infix operator such that $p_1 \triangleright_p p_2$ means $\triangleright(p, p_1, p_2)$.

Now we are set to write down the formulas in our current setting that would essentially be able to capture the nash equilibrium for a priority separable structure. Before that we just need to write down the syntax of the *extended* **MLFP** which varies slightly from the previous definitions.

Syntax of *extended* MLFP Logic

Let σ be a first order relational vocabulary. Let $(S_i)_{i\in\mathbb{N}}$ be a sequence of monadic relation symbols, such that for each i, $S_i \notin \sigma$. These are the second order fixed-point variables of the logic. Let $(\mathbf{X}_i)_{i\in\mathbb{N}}$ be a set of binary second order variables. We will usually want to express the other binary second order variables as **X,Y,Z**. These are supposed to represent the strategy profiles. Semantically, they are meant to capture $\mathbf{X} \subseteq N \times T$. Since we have three different types of domains, we usually represent them by three different types of variables - $p, u.v.w$ for player variables, o, a, b - for outcome variables and s, t - for strategy variables.

The set of all formulas, $\Phi_{extended\ MLFP}$, is defined inductively as follows, where $fv^1(\alpha)$ = the set of first order free variables in α; $fv^2(\alpha)$ = the set of all relation symbols S_i occurring in α; $fv(\alpha) = fv^1(\alpha) \cup fv^2(\alpha)$.

- A **MLFP** atomic formula is of the form $R(x_1, \ldots, x_k)$, or $x = y$ or $S_i(x)$ or X(p,s) where R is a k-ary relation symbol in σ, S_i is a monadic second order fixed-point variable and where X_i, Y_i are second order variables denoting strategy profiles.

- If α, β are **MLFP** formulas then so are, $\sim\alpha$, $\alpha\wedge\beta$ and $\alpha\vee\beta$. $fv(\sim\alpha) = fv(\alpha)$ $fv(\alpha\wedge\beta) = fv(\alpha\vee\beta) = fv(\alpha)\cup fv(\beta)$.

- If α is a **MLFP** formula, $x \in fv^1(\alpha)$, then so are $\exists x\alpha$, $\forall x\alpha$ $fv(\exists x\alpha) = fv(\forall x\alpha) = fv(\alpha)\setminus\{x\}$.

- If α is a **MLFP** formula, $S_i \in fv^2(\alpha)$, $x \in fv^1(\alpha)$, and $u \notin fv^1(\alpha)$ and S_i occurs positively in α, then $[\mathbf{lfp}_{S_i,x}\ \alpha](u)$ is an **MLFP** formula.
  $fv([\mathbf{lfp}_{S_i,x}\ \alpha](u)) = fv(\alpha) \setminus \{S_i, x\}\ \cup \{u\}$.

- Now that we have defined **MLFP** formulas.
  If α is a MLFP formula , then
  $\Psi\ :=\ \alpha\ |\ \exists X_1 \exists X_2 \ldots \exists X_m \forall Y_1 \forall Y_2 \ldots \forall Y_n \alpha(X_1, \ldots, X_m, Y_1, \ldots, Y_n)$ is an *extended* **MLFP** formula.

Expressing Nash Equilibrium for the generalised setting

Let σ denote the *extended* **MLFP** language consisting of the above vocabulary. *Priority Separable structures* are the σ-structures where the interpretation on the relational symbols conform to the above definition.

- func($\mathbf{X}$) $= \forall p(P(p) \implies \sim \exists s_1, s_2(S(s_1) \wedge S(s_2) \wedge \mathbf{X}(p, s_1) \wedge \mathbf{X}(p, s_2)))$ - checks if $\mathbf{X}$ is a function from the player to strategy set.

- chkStr($\mathbf{X}$) $= \forall p \forall s(P(p) \wedge S(s) \wedge \mathbf{X}(p, s) \rightarrow \mathcal{S}(p, s)) \wedge$ func($\mathbf{X}$) - checks if $\mathbf{X}$ is a strategy profile.

- first(p,z) $= E(z, p) \wedge \forall p_1(P(p_1) \wedge E(p_1, p) \wedge (z \rhd_p p_1))$ - states that z is the first player is the priority ordering of player p.

-

$$\begin{aligned} nxt(p, u, v) =& E(u, p) \wedge E(v, p) \wedge (u \rhd_p v) \\ & \wedge \Big(\sim \exists p_1(P(p_1) \wedge E(p_1, p) \\ & \wedge (u \rhd_p p_1) \wedge (p_1 \rhd_p v)) \Big) \end{aligned}$$

This formula states that v is the immediate next player to player u among the priority ordering of player p.

- P-Pay$(p, u, \mathbf{X}, o) = \exists s_1, s_2(S(s_1) \wedge S(s_2) \wedge \mathbf{X}(p, s_1) \wedge \mathbf{X}(u, s_2) \wedge u(p, s_1, s_2, o))$ - this captures the partial payoff o, of player p against player u in the strategy profile $\mathbf{X}$.

- notEq(p,$\mathbf{X}$,$\mathbf{Y}$) $= \exists x, y(S(x) \wedge S(y) \wedge x \neq y \wedge X(p, x) \wedge Y(p, y))$ - states that the two strategy profiles $\mathbf{X}$ and $\mathbf{Y}$ differ at the coordinate index for player p.

- It states that player p has a 1-step improvement from $\mathbf{X}$ to $\mathbf{Y}$.

$$\begin{aligned}\text{1-step}(\text{p},\mathbf{X},\mathbf{Y}) =& \text{chkStr}(\mathbf{X}) \wedge \text{chkStr}(\mathbf{Y}) \wedge \text{notEq}(p, \mathbf{X}, \mathbf{Y})\\ &\wedge \forall u\Big(P(u) \wedge u \neq p \implies \exists z(S(z) \wedge \mathbf{X}(u, z) \wedge \mathbf{Y}(u, z)\Big)\end{aligned}$$

- $$\begin{aligned}\psi_1(p, u, \mathbf{X}, \mathbf{Y}) = \exists a, b\Big(&O(a) \wedge O(b) \wedge a \neq b\\ &\wedge \text{P-Pay}(p, u, \mathbf{X}, a) \wedge \text{P-Pay}(p, u, \mathbf{Y}, b)\\ &\wedge (a \ll_p b)\Big)\end{aligned}$$

- $$\begin{aligned}\psi_1'(p, u, \mathbf{X}, \mathbf{Y}) = \exists a, b\Big(&O(a) \wedge O(b) \wedge a \neq b\\ &\wedge \text{P-Pay}(p, u, \mathbf{X}, a) \wedge \text{P-Pay}(p, u, \mathbf{Y}, b)\\ &\wedge (b \ll_p a)\Big)\end{aligned}$$

- $\psi_2(p, u, \mathbf{X}, \mathbf{Y}) = \exists a(O(a) \wedge \text{P-Pay}(p, u, \mathbf{X}, a) \wedge \text{P-Pay}(p, u, \mathbf{Y}, a))$

We can then write the formula Dev$(p, \mathbf{X}, \mathbf{Y})$ that states that the strategy profile $\mathbf{Y}$ is an improvement for player p from the strategy profile $\mathbf{X}$ by a 1-step deviation.

$$\begin{aligned}
\mathrm{Dev}(p, \mathbf{X}, \mathbf{Y}) &= \text{1-step}(p, \mathbf{X}, \mathbf{Y}) \\
&\quad \wedge \exists v \left(P(v) \wedge [\mathbf{lfp}_{M,w}\alpha](v, p, \mathbf{X}, \mathbf{Y}) \wedge \psi_1(p, v, \mathbf{X}, \mathbf{Y}) \right) \\
\alpha(M, w, p, \mathbf{X}, \mathbf{Y}) &= P(p) \wedge P(w) \wedge \min(p, w) \\
&\quad \vee \exists v [M(v) \wedge \mathrm{nxt}(p, v, w) \wedge \psi_2(p, v, \mathbf{X}, \mathbf{Y})]
\end{aligned}$$

Now, existence of Nash equilibrium can be characterised using the following formula:

- Imp($\mathbf{X}$,$\mathbf{Y}$) $= \exists p(P(p) \wedge$ Dev(p,$\mathbf{X}$,$\mathbf{Y}$))
- NE($\mathbf{X}$) $= \forall$ $\mathbf{Y}$ $\sim$ Imp($\mathbf{X}$,$\mathbf{Y}$)
- $$\begin{aligned}
\mathrm{NE}(\mathbf{X}) = \forall p \forall s\ \exists \mathbf{Y} \Bigg(& Y(p, s) \wedge \text{1-step}(\mathrm{p}, \mathbf{X}, \mathbf{Y}) \\
& \wedge \Bigg(\\
& \exists v \left(P(v) \wedge [\mathbf{lfp}_{M,w}\alpha](v, p, \mathbf{X}, \mathbf{Y}) \wedge \psi_1'(p, v, \mathbf{X}, \mathbf{Y}) \right) \\
& \vee \\
& \forall v \left(P(v) \wedge E(v, p) \wedge [\mathbf{lfp}_{M,w}\alpha](v, p,\ \mathbf{X}, \mathbf{Y}) \right) \\
& \Bigg) \\
\Bigg) &
\end{aligned}$$

- $\mathbb{G} \models \exists$ $\mathbf{X}$ NE($\mathbf{X}$)

Remark on the Complexity

The complexity has change to the first level of $\exists\forall$ hierarchy. Which is where it should be!

We can see that even though the formulas become succinct in size when compared to the previous setting where it was a function of n, the number of players, but we are now having to tackle a strange fragment of second order logic coupled with **MLFP** - $\exists X \forall Y($ mlfp $)$, which we call *extended* **MLFP**, where the second order quantification is a binary relation and not a monadic one. From a logic standpoint, we know that for the NP-hardness of model checking in second order logic over graph structures, one only needs three monadic quantifiers to encode the three colorability problem. And, we know about the complexity of Nash Equilibrium for polymatrix games to be NP-complete from Theorem 5 of [**?**], therefore the model checking for formulas in this fragment is at least NP-hard. The more interesting question is what tight upper bounds can we get in this fragment. The **MLFP** portion of the formula can surely be unrolled out into a quantifier alternated second order fragment.

4.4 Improvement dynamics in social choice theory

In this section we will show how we can extend our reasoning for games to ideas in social choice theory. We can see that we can hope for a common language across these differing contexts!

We can employ a form of reasoning common in one (say normal form games) in another (say fair resource allocations) and thus transfer results and techniques. We show that the idea of improvement under swaps corresponds to certain form of strong equilibria and coalitional improvement in games.

Dynamics in iterated voting again correspond to improvement dynamics in games. In such cases when the structures studied possess interesting properties such as the finite improvement property or weak acyclicity we get certificates of existence of equilibria.

Interesting subclasses of games (such as potential games) possess such properties and by "transfer" we can look for similar subclasses in social choice contexts, and vice versa.

4.4.1 Allocation of indivisible goods

An important problem often studied in economics and computer science is the allocation of resources among rational agents. This problem is fundamental and has practical implications in various applications including college admissions, organ exchange and spectrum assignment. In this paper, we consider the setting where there are $[n]$ agents and a set $A = \{a_1, \ldots, a_m\}$ of m indivisible items. An allocation $\pi : N \to 2^A$ such that $\cup_{i \in [n]} \pi(i) = A$ and for all $i, j \in [n]$, $i \neq j$, $\pi(i) \cap \pi(j) = \emptyset$. In the most general setting, each agent i has a preference ordering $\prec_i$ over the allocations. Thus an instance of an allocation problem can be specified as a tuple $H = ([n], A, \{\prec_i\}_{i \in [n]})$. Let Π denote the set of all allocations. In this setting, each allocation π can be viewed as defining an outcome and agents have a preference ordering over such outcomes.

There are natural notions of fairness in the context of allocation systems; for instance, we are interested in nodes that are *envy free*. An agent i envies an agent j at node x if there exists a node y such that $y \succ_i x$ and the allocation for i at y is the same as the allocation for j at x. A node is envy-free if no player envies another at that node. We might then want to assert that an envy free node is reachable from any node. Note that we only need to enrich the first order vocabulary to speak of $x[i], y[j]$ etc to express envy-freeness, and the lfp operator is sufficient to specify reachability of such nodes.

- $reach_\phi(S, x) = \phi(x) \vee \exists y(E(y, x) \wedge S(y))$
- $\phi - reachable = \forall u.[\mathbf{lfp}_{S,x}\ reach](u)$

When agents are allowed to exchange items with each other, stability of an allocation is a natural solution concept to study. Core stable outcomes are defined as allocations in which no group of agents have an incentive to exchange their items as part of an internal redistribution within the coalition. The improvement graph structure can capture the dynamics involved in such a sequence of item exchange in a natural manner. The associated improvement graph can be defined over the set of vertices Π. Since the deviation involves exchange of goods among a subset of players (rather than a unilateral deviation by a single player), the edge relation is indexed with a subset $u \subseteq [n]$. That is, for $\pi, \pi' \in \Pi$ and $u \subseteq [n]$, we have $\pi \to_u \pi'$ if for all $i \in u$, $\pi(i) \prec_i \pi'(i)$, $\pi(i) \neq \pi'(i)$ and for all $j \notin u$, $\pi(j) = \pi'(j)$.

Note that the improvement graph here is different from the ones we discussed earlier in a crucial sense. When agents in u swap goods, the allocation for other players outside $[u]$ is unaffected. If each agent's preference ordering depends only on the valuation of the bundle of items that the agent is allocated then their satisfaction is unchanged. However, agent 1 may swap goods with 2 and then use some the goods acquired to make a swap deal with 3 thus leading to interesting causal chains. In effect the entire space of allocations may be tentatively explored by the agents. A finite path in the improvement graph corresponds to a finite sequence of exchanges that converge to a stable outcome. An important question is whether stable allocations always exist and whether a finite sequence of exchanges can converge to such an allocation.

A common assumption that is made in allocation problems, is that the preference ordering for each agent i depends only on the bundle of items assigned to agent i. A special case of the above setting is when $n = m$ (i.e. the number of agents and the items are the same) and π is required to be a bijection. An instance of such an allocation problem A along with an initial allocation π_0 defines the well known Shapley-Scarf housing market [**?**].

When we allow exchange of goods in the housing market, it is known that a simple and efficient procedure termed as Gale's Top Trading Cycle can compute an allocation that is core stable. The allocation constructed in this manner also satisfies desirable properties like strategy-proofness and Pareto optimality.

Housing market with priority separable externalities

There are many instances where an agent's utility depends not only on the items that are allocated but also the allocation received by other agents - possibly within a social context. This is particularly natural in the housing market where the utility depends not just on the house that an agent is allocated but also on who is allocated the neighbouring houses. Formally, this implies that the utility functions depend on agent externalities. When these externalities are separable, we can use the framework developed in priority separable games to reason about the stability of such allocations without having to explicitly construct the improvement graph.

Given an allocation π, a pair of agents (i, j) is called a *blocking pair* if there exists π' such that $\pi'(i) = \pi(j)$, $\pi'(j) = \pi(i)$, $\pi \prec_i \pi'$, $\pi \prec_j \pi'$ and for all $k \in N \setminus \{i, j\}$, $\pi'(k) = \pi(k)$. An allocation π is *stable* if π does not have a blocking pair.

We can use the MLFPC framework to characterise stable allocations in housing markets using the formulas given below. To keep the presentation simple, we retain the notation we used in the previous section. We view the strategy profile as the profile of allocations. While we don't equate the outcomes with the choice of strategies, in typical resource allocation setting, these two parameters are the same. The main difference from the previous section is that unilateral deviation is not possible. The only way a strategy profile can change is when two players agree to swap their items.

- $\mathrm{swap}_{i,j}(\vec{x},\vec{y}) = \bigwedge_{k \in N \setminus \{i,j\}} x_k = y_k \wedge x_i = y_j \wedge y_i = x_j$ - states that the difference between $\vec{x}$ and $\vec{y}$ is that players i and j have swapped their items.

- $\mathrm{chkvOut}(\vec{a}) = \bigwedge_j O(a_j)$ - checks if it is an outcome vector.

-
$$\begin{aligned}\mathrm{chkPay}(p,\vec{x},\vec{a},\vec{y},\vec{b}) = \;& \mathrm{chkOut}(p,\vec{x},\vec{a}) \\ & \wedge \mathrm{chkOut}(p,\vec{y},\vec{b}) \\ & \wedge \exists v \Big([\mathbf{lfp}_{M,x}\alpha](v,p,\vec{x},\vec{a},\vec{y},\vec{b}) \\ & \wedge \psi_1(p,v,\vec{x},\vec{a},\vec{y},\vec{b}) \Big)\end{aligned}$$

-
$$\begin{aligned}\phi_{\{i,j\}}(\vec{x},\vec{a},\vec{y},\vec{b}) = \;& \exists w, v (P(w) \wedge P(v) \wedge \mathcal{S}(u,x_i) \wedge \mathcal{S}(v,x_j) \\ & \wedge \mathrm{swap}_{i,j}(\vec{x},\vec{y}) \wedge \mathrm{chkPay}(w,\vec{x},\vec{a},\vec{y},\vec{b}) \wedge \mathrm{chkPay}(v,\vec{x},\vec{a},\vec{y},\vec{b}))\end{aligned}$$

  This formula is similar to the one we saw in the previous section.

  It says that it is possible to move from an allocation $\vec{x}$ to $\vec{y}$ where players i and j swap their items in which the utility of both the players strictly increase. Due to the inability of players to make unilateral deviation, we have two instances of the **lfp** operator in this formula. **Note** that the quantifier rank of this formula remains 1.

- $\phi(\vec{x},\vec{a},\vec{y},\vec{b}) = \bigvee_{i<j} \phi_{\{i,j\}}(\vec{x},\vec{a},\vec{y},\vec{b})$.

- $$\begin{aligned}\text{swapstep}(\vec{x},\vec{y}) =& \exists \vec{a},\vec{b}\Big(\text{chkStrat}(\vec{x}) \wedge \text{chkStrat}(\vec{y}) \\ & \wedge \text{chkvOut}(\vec{a}) \wedge \text{chkvOut}(\vec{b}) \wedge \phi(\vec{x},\vec{a},\vec{y},\vec{b})\Big)\end{aligned}$$

- $\text{stablealloc}(\vec{x}) = \sim \exists \vec{y}\ \text{swapstep}(\vec{x},\vec{y})$ - states that $\vec{x}$ is a stable allocation.

The algorithmic properties of allocation problems where the reference orderings depend on externalities are studied in[**?**, **?**]. Apart from stability, notions of fairness like envy-freeness, propotionality and maximin share guarantee are also well studied in the context of allocation of indivisible items [**?**, **?**]. Analysis of the improvement graph is also useful in the context of fairness notions. Existence of a finite improvement path terminating in a fair allocation would indicate the possibility of convergence to a fair allocation in terms of an exchange dynamics.

4.4.2 Voting systems

Consider an electorate consisting of a set $[n] = \{1, \ldots, n\}$ of n voters and a set C of m candidates. Let $\mathfrak{R}$ be a voting rule that considers the preference of each voter over the candidates and chooses k winning candidates. The choice sets for all voters are the same $S = \mathfrak{L}(C) = \{\pi | \pi \text{ is a permutation of } C\}$. The outcome set is $O = \binom{C}{k}$. The voting rule $\mathfrak{R} : S^n \mapsto O$ specifies which candidates win given the complete preferences of all voters. We assume that each voter i has a preference ordering $\prec_i$ over the outcome set O. Thus the voting system can be given by the tuple $L = (n, m, \prec_1, \ldots, \prec_n, \mathfrak{R})$.

The improvement graph G_L associated with L is as before: $G_E = (V, E)$ where $V = S^n$, the set of strategy profiles of voters; $E \subseteq (V \times [n] \times V)$ is the improvement relation for voter i, given by: $s \rightarrow_i s'$ if $\mathfrak{R}(s') \succ_i \mathfrak{R}(s)$, $s_i \neq s'_i$ and for all $j \neq i$, $s_j = s'_j$.

Voting equilibria have been studied by Myerson and Weber [**?**]. In general, one speaks of the *bandwagon effect* in an election if voters become more inclined to vote for a given candidate as her standing in pre-election polls improve, or the *underdog effect*, if voters become less inclined to vote for a candidate as her standing improves. Myerson and Weber suggest that equilibrium arises when the voters, acting in accordance with both their preferences for the candidates and their perceptions of the relative chances of candidates in contention for victory, generate an election result that justifies their perceptions. Note that the improvement path again gives us the possibility of 'interaction' arising from voter preferences, and we can analyse this in the context of specific voting rules.

Given that agents may have incentive to strategically misreport their preferences, it is natural to study the convergence dynamics when voting is modelled as a game. Iterative voting [**?**, **?**, **?**] is a formalism that is useful to analyse the strategic dynamics when at each turn a voter is allowed to alter her vote based on the current outcome until it converges to an outcome from which no voter wants to deviate. In general, the outcome of iterative voting may depend on the order of voters' changes. Again, voters act myopically, without knowing the others' preferences. This dynamics is again reflected by the improvement path as discussed here and sink nodes correspond to Nash equilibria. Thus given a voting rule, it is natural to ask what equilibria are reachable from a given vote profile.

Once again, we can use the MLFPC framework to specify convergence dynamics in iterative voting.

4.4.3 Remark

We have suggested that the improvement graph is an important structure for the logical study of mathematical social sciences. A natural alternative to consider would be to translate all the models into that of games, and then induce the improvement graph over the defined model. This is certainly possible, but in general this can lead to an increase in the size of the graph. Moreover since we hope to use MLFPC not only to unify these contexts but also differentiate them (in terms of logical resources needed), such a reduction would not be helpful.

4.5 Discussion

We see in this chapter a preliminary investigation, hopefully leading to a descriptive complexity theoretic study of fundamental notions in games and interaction. It is clear that fixed-point computations underlie the reasoning in a wide variety of such contexts, and logics with least fixed-point operators are natural vehicles of such reasoning. We expect that this is a minimal language for improvement dynamics, but with further vocabulary restrictions that need to be worked out. Proceeding further, we would like to delineate bounds on the use of logical resources for game theoretic reasoning. For instance, one natural question is the characterization of equilibrium dynamics definable with at most one second order (fixed-point) variable.

Expressiveness needs to be sharpened from the perspective of models as well. We would like to characterize the class of improvement graphs for different subclasses of games, resource allocation systems and voting rules, considering the wide variety of details in the literature. This would in general necessitate enriching the logical language and we wish to consider minimal extensions.

Another important issue is the identification of subclasses that avoid the navigation of huge improvement graphs. Potential games provide an interesting subclass and they correspond to some appropriate allocation rules and forms of voting (under specific election rules). But these are only specific exemplifying instances, studying the structure of formulas and their models will (hopefully) lead us to many such correspondences.

An important direction is the study of infinite strategy spaces. Clearly the model checking algorithm needs a finite presentation of the input but this is possible and it is then interesting to explore convergence of fixed-point iterations.

We would be successful if researchers from the various disciplines can make use of this framework and learn about the different properties from the other disciplines. Even better would be if those other properties aid in algorithm research in their respective communities. We would also want to investigate the connections between complexity and logic. Say, the whole idea of reachability as a notion being captured in directed graphs. Then we would also like to find out whether the nesting of mlfp quantifiers is necessary? Can the nesting depth be reduced. We would also like to understand whether mlfp like lfp can admit a canonical form of mlfp_0 just like lfp_0.

Further on the lines of inexpressibility we did try to capture certain idea in game theory like IENBR for our purposes in our current logic and it seemed out of our reach to express. A very interesting project would be to try to show inexpressibility of such property of games, which can help in more foundational understanding of the least fixed point logics. This point can be elaborated

On the model theoretic front, we need to see what kind of subsets of the set of all directed labelled graphs are the improvement graphs for each of the three settings explored.

Chapter 5

Modal logic for Social network games

This chapter's aims to motivate the use of a modal logic to describe the a class of games which are succinctly presented. In the previous chapter we saw the use of the monadic variant of First Order Logic with a Least Fixed Point to describe the properties of the Improvement Graph for general normal form games and saw it being utilised as an uniform framework to also reason about different computational models from the social choice theory setting, namely voting and fair division.

We ask ourselves, whether we can characterise a class of games which are not isomorphic invariants of each other?

For this seemingly innocuous question it was finally revealed that First Order and its variants are too powerful to go about this exercise, what we needed is a property like Bisimulation to be possible for us. Thus, our research direction inevitably turned towards Modal Logics!

5.1 Introduction

A fundamental assumption of non-cooperative game theory is that players strategize individually and independent of each other. This was referred to as the *Great Simplification* by von Neumann in 1928 and indeed, this is what leads to the abstraction of normal form games. However, such a flat structure on the set of players is not always realistic. In the case of social networks like Facebook and Twitter, individuals are influenced by their friends, and often seek to influence their friends, in the choices they make. The 'payoff' in such interactive behaviour is often social, in the sense that matching one's friends' choices may indeed be the desired outcome. Such *majority games* (and their dual, so-called *minority games*) are also extensively discussed in the literature. 'Facebook logic' ([**?**]) and its counterparts discuss such relationships and their impact on decision making.

5.1.1 Motivation

A specific class of games on social networks was studied by Apt and Simon ([**?**]), which is the starting point of departure here. While they study the complexity of computing equilibria in such games, we take their *improvement dynamics* and seek a logical description. The central question we take up is this: how do we abstract away from the details of utilities and preferences, and get to the core of strategization by players in such games?

A natural way for such abstraction is to consider game equivalences and seek logical descriptions of equivalence classes. When outcomes are determined locally, by neighbourhoods in the social network, this induces further structure in the improvement graphs, which leads to interesting bisimulation classes. This naturally leads us to a modal logic as a tool for strategization structure.

5.1.2 Related Work

5.1.3 Preliminaries

5.2 Social Network Games

Definition 5.2.1: Social Network Game

It is a tuple $G = (n, \Sigma, \Omega, E, \pi)$, where $E \subseteq ([n] \times [n])$ is the edge relation of the social network graph, $\pi = (\pi_1, \dots, \pi_n)$ is the payoff function, one for each player, where $\pi_i : \Sigma^{|N_i|} \to \Omega$, $N_i = \{i\} \cup \{j \mid (j, i) \in E\}$, is the *neighbourhood* of player i.

π induces a function from $\Sigma^n \to \Omega^n$ which, by abuse of notation, we again denote by π.

5.2.1 Modelling examples

Threshold based reasoning is common in games on social networks ([**?**]), and we present two examples of such modelling.

The first is that of a **Majority Game**, used in modelling social phenomena such as voting: ([**?**], [**?**]). $\Sigma = \{0, 1\}$. Let $\sigma = (a_1, \dots, a_n)$ be any profile. The payoff for player i at σ is 1 if $\frac{|\{j \in N(i) | a_j = a_i, j \neq i\}|}{|N(i)|} > \frac{1}{2}$, and is 0 otherwise. The trivial equilibria for this game are all players choosing 0, or all choosing 1.

A non-trivial vote will be the following:

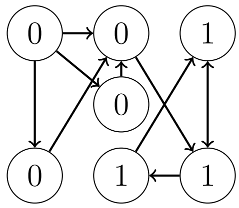

Figure 5.1: Non trivial Nash Equilibrium in the Majority Game

For another example, consider the **"Best Shot" Public Goods game** ([**?**]). Again, $\Sigma = \{0, 1\}$. In this game, there is an option of taking an altruistic action for the public good, or refraining from it. Doing good carries a fixed uniform cost $c \in (0, 1)$. Of course, if some neighbour takes action, it is much better and one can enjoy the result doing nothing. Alas, if everyone thinks so, nobody benefits. This is specified by the payoff function as follows. Again let $\sigma = (a_1, \ldots, a_n)$:

$$\pi_i(\sigma) = \begin{cases} 0 & \text{if } a_i = 0, a_j = 0 \text{ for all } j \in N(i) \\ 1 & \text{if } a_i = 0, a_j = 1 \text{ for some } j \in N(i) \\ 1 - c & \text{if } a_i = 1 \end{cases}$$

Here is a Nash equilibrium for this game:

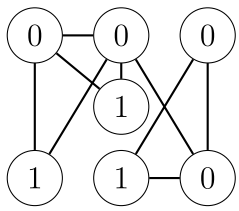

Figure 5.2: Nash Equilibrium in the Public Goods Game

5.2.2 Bisimulation Relation

The Improvement Graph dynamics induces a natural **game equivalence**: we can consider two games to be equivalent if they have a similar (but not necessarily isomorphic) improvement structure.

Definition 5.2.2: improvement bisimulation

Let $\mathbb{G}, \mathbb{G}'$ be two n player games with strategy sets Σ and Σ', a relation $R \subseteq (\Sigma^n, \Sigma'^n)$ is an improvement bisimulation if whenever $(\sigma, \sigma') \in R$, for all $i \in [n]$,

- whenever $\sigma \rightarrow_i \sigma_1$ in $I_{\mathbb{G}}$, there exists profile σ_1' in game $\mathbb{G}'$ such that $\sigma' \rightarrow_i \sigma_1'$ in $I_{\mathbb{G}'}$ and $(\sigma_1, \sigma_1') \in R$.
- whenever $\sigma' \rightarrow_i \sigma_1'$ in $I_{\mathbb{G}'}$, there exists profile σ_1 in game G such that $\sigma \rightarrow_i \sigma_1$ in $I_{\mathbb{G}}$ and $(\sigma_1, \sigma_1') \in R$.

The relation is on games in general rather than on social network games. Clearly, the improvement bisimulation relation is an equivalence relation. We say that $\mathbb{G}$ and $\mathbb{G}'$ are bisimilar if there exists a nonempty bisimulation on their improvement graphs. When we reduce strategic form games by this equivalence, we abstract from specific outcomes as well as player strategies but preserve the *player strategisation structure*. Note that outcomes and orderings on them have entirely disappeared in the bisimulation classes, only the information that some improvement in outcome is possible (or not) is retained. This is a semantic characterization, and we need structural constraints to capture the semantic conditions. What we would like to do is to study player rationale to provide logical structure to the strategization. In the context of social network games, we use threshold reasoning over player neighbourhoods as a way of specifying this rationale.

5.3 Logic

Player strategisation involves reasoning about strategies played by other players in their neighbourhood. Thus mutual strategization by players becomes relevant, and the logic we define below includes precisely such *local* reasoning by players as well as *global* improvement dynamics.

5.3.1 Syntax

The formulas of the logic are presented in two layers: **local player formulas** and **global outcome formulas**. The logic is parameterised by n, the number of players, and the strategy set Σ.

The syntax of local formulas is given by:

$$\alpha \in L_i ::= a \in \Sigma \mid e_j \mid \neg\alpha \mid \alpha \vee \alpha' \mid N_{rel\ r}\ \alpha$$

where $rel \in \{\geq, \leq, <, >\}$, $i \in [n]$ and r is a rational number, $0 \leq r \leq 1$.

The atomic formula a asserts that player i chooses a, and the atomic formula e_j asserts that there is a directed edge from j to i, that player i is dependent on j. $N_{rel\ r}\alpha$ considers the size of the neighbourhood choosing α: for instance, $N_{\leq r}\alpha$ asserts that at most an r-fraction of players in the neighbourhood of i support α.

Fix P a finite set of atomic propositions denoting *conditions on outcomes.* These are qualitative outcomes, used to denote levels of satisfaction. We will characterize outcomes by sets of propositions, which can be equivalently thought of as boolean formulas on P.

The syntax of global formulas is given by:

$$\phi \in \Phi ::= p@i,\ p \in P \mid \alpha@i, \alpha \in L_i \mid \neg\phi \mid \phi \vee \phi' \mid \langle i\rangle\phi \mid \Diamond^*\phi$$

The global formulas constitute a standard propositional modal logic of transitive closure built over local formulas. Note that the atomic formulas $p@i$ and $\alpha@i$ are of different sort: the former refers to outcomes and the latter to strategies. The other boolean connectives $\wedge$, $\supset$ and $\equiv$, for conjunction, implication and equivalence are defined in the standard manner, for both local and global formulas. The dual formulas are: $[i]\phi = \neg\langle i\rangle\neg\phi$ and $\Box^*\alpha = \neg\Diamond^*\neg\phi$ We use the abbreviation $\bigcirc\phi = \bigvee_{i\in[n]} \langle i\rangle\phi$ and $\odot\phi = \neg\bigcirc\neg\phi$. (We use $\top$ and $\bot$ to refer to the propositional constants 'True' and 'False' which are coded by $p@i \vee \neg p@i$ and $p@i \wedge \neg p@i$, for a fixed propositional symbol p.)

5.3.2 Semantics

The formulas are interpreted over strategy profiles of social network games.

Definition 5.3.1: Model of a social network game

$M = (n, \Sigma, 2^P, E, \pi)$ where $\Omega = 2^P$ is the set of outcomes. The ordering can be seen as an ordering on boolean formulas on P.

The semantics is given by assertions of the form $M, \sigma \models \phi$, read as ϕ is true of the strategy profile σ in model M. This in turn depends on the satisfaction relation for local formulas. For $i \in [n]$ and $\alpha \in L_i$, we define i-local satisfaction relations:

- $M, \sigma \models_i a$ if $\sigma[i] = a$.
- $M, \sigma \models_i e_j$ if $(j, i) \in E$.
- $M, \sigma \models_i \neg\alpha$ if $M, \sigma \not\models_i \alpha$.
- $M, \sigma \models_i \alpha \vee \beta$ if $M, \sigma \models_i \alpha$ or $M, \sigma \models_i \beta$.
- $M, \sigma \models_i N_{rel\ r}\ \alpha$ if $\frac{|\{j|M,\sigma\models_j\alpha\}|}{|N_i|}\ rel\ r$.

The semantics of global formulas can then be defined as follows. Below, let $\rightarrow^* = (\cup_i \rightarrow_i)^*$, the reflexive transitive closure of the union of the improvement edge relations.

- $M, \sigma \models p@i$ if $p \in \pi_i(\sigma)$.

- $M, \sigma \models \alpha@i$ if $M, \sigma \models_i \alpha$.

- $M, \sigma \models \neg\phi$ if $M, \sigma \not\models \phi$.

- $M, \sigma \models \phi \vee \psi$ if $M, \sigma \models \phi$ or $M, \sigma \models \psi$.

- $M, \sigma \models \langle i \rangle \phi$ if there exists σ' such that $\sigma \rightarrow_i \sigma'$ and $M, \sigma' \models \phi$.

- $M, \sigma \models \Diamond^* \phi$ if there exists σ' such that $\sigma \rightarrow^* \sigma'$ and $M, \sigma' \models \phi$.

5.4 Expressivity

It is easy to see that Nash Equilibrium is given by the simple formula: $NE = \bigwedge_{i \in [n]} [i]\bot$. To assert that there is a path from the current profile to a Nash Equilibrium, we write: $\Diamond^*(NE)$. To assert the "Weak Finite Improvement Property", that a Nash Equilibrium profile is reachable from every profile, we write: $\Box^*\Diamond^*(NE)$.

The strategy specification for the majority game is simple. Let the payoff set be given by: $P = \{p_0, p_1\}$ with $\{p_0\} \preceq \{p_1\}$. The formula $(N_{>\frac{1}{2}}(e_j \wedge 1))@i \equiv p_1@i$ defines the payoff map.

For the Public Goods Game, let $P = \{p_0, p_c, p_1\}$ with $\{p_0\} \preceq \{p_c\} \preceq \{p_1\}$. The payoff map is specified by:

$$(\bigwedge_i 0@i) \supset (\bigwedge_i p_0@i) \wedge \bigwedge_i (1@i \supset p_c@i) \wedge \bigwedge_i ((0 \wedge N_{>0}1)@i \supset p_1@i)$$

5.5 Axiomatisation

We now present an axiomatization of the valid formulas. We have one axiom system Ax_i for each player i in the system, and in addition a global axiom system AX to reason about improvement dynamics. In some sense, this helps to isolate how much global reasoning is required.

Below, we say $rel\ r$ *entails* $rel'\ r'$ when $r \leq r'$ and either $rel = rel' = \leq$ or $rel = <$ and $rel' = \leq$, or $r \geq r'$ and either $rel = rel' = \geq$ or $rel = >$ and $rel' = \geq$. Further we say rel' is the *complement* of rel if one of them is $\geq$ and the other is $<$, or one is $\leq$ and the other $>$.

We use the notation $\vdash_i \alpha$ to mean that the formula $\alpha \in L_i$ is a theorem of system Ax_i. Similarly, $\vdash \phi$ means that ϕ is a theorem of the global system.

Ax_i, The axiom schemes for agent i

$(A0_i)$ All the substitutional instances of propositional tautologies

$(A1_i)$ $N_{rel\ r}(\alpha \supset \beta) \supset (N_{rel\ r}\alpha \supset N_{rel\ r}\beta)$

$(A2_i)$ $\alpha \supset N_{>0}\alpha$

$(A3_i)$ $N_{rel\ r}\alpha \equiv \neg N_{rel'\ r}\alpha$, $\quad rel'$ complement rel

$(A4_i)$ $N_{rel\ r}\alpha \supset N_{rel' r'}\alpha$, $\quad rel\ r$ entails $rel'\ r'$

Inference rules

$$(MP_i)\ \frac{\alpha, \quad \alpha \supset \beta}{\beta} \qquad (NG_i)\ \frac{\alpha}{N_{\geq 1}\alpha}$$

The axioms of the local system are quite standard. The Kripke axiom applies to every instance of the $N_{rel\ r}$ modality, and the remaining axioms express properties of inequalities. The rule (NG_i) reflects the fact that properties which are invariant in the system hold throughout the neighbourhood.

In the global axiom system. we have Kripke axioms for $[i]$ modalities and for transitive closure, and an induction rule. We need a "transfer" rule to infer $\alpha@i$ globally when we infer α locally. The remaining axioms relate to social network games: specifying the fact that formulas are asserted at strategy profiles, corresponding to one choice for each player, that payoffs for a player are determined by the player's neighbourhood, and so on.

Global axiom schemes AX

$(B0)$ $(\neg\alpha)@i \equiv \neg\alpha@i$

$(B1)$ $(\alpha \vee \beta)@i \equiv (\alpha@i \vee \beta@i)$

$(B2)$ $[i](\phi_1 \supset \phi_2) \supset ([i]\phi_1 \supset [i]\phi_2)$

$(B3)$ $\Diamond^*\phi \equiv (\phi \vee \bigcirc\Diamond^*\phi)$

$(B4)$ $\alpha@j \equiv [i]\alpha@j, \quad j \neq i$

$(B5)$ $(p@j \equiv [i]p@j) \wedge (\neg p@j \equiv [i]\neg p@j) \quad i \notin N_j$

$(B6)$ $(\bigwedge_{j \in N_i} a_j@j) \supset$
$((p@i \supset \Box^*(\bigwedge_{j \in N_i} a_j@j \supset p@i)) \wedge (\neg p@i \supset \Box^*(\bigwedge_{j \in N_i} a_j@j \supset \neg p@i))$

$(B7)$ $e_j@i \equiv \bigodot e_j@i$

$(B8)$ $\Box^* \bigwedge_{i \in [n]} (\bigvee_{a \in \Sigma} (a@i \wedge \bigwedge_{b \neq a} \neg b@i))$

$(B9)$ $(N_{rel\ r}\alpha)@i \supset \bigvee_{J,K \subseteq N_i} (\bigwedge_{j \in J} \alpha@j \wedge \bigwedge_{k \in K} \neg\alpha@j)\ K = N_i - J, \dfrac{|J|}{|J \cup K|}\ rel\ r$

Inference rules

(MP) $\dfrac{\phi, \quad \phi \supset \psi}{\psi}$ $\quad (GG)$ $\dfrac{\vdash_i \alpha}{\alpha@i}$ $\quad (G_i)$ $\dfrac{\phi}{[i]\phi}$

$(Conc)$ $\dfrac{\gamma_1 \vee \ldots \vee \gamma_\ell \quad (N_i \cap N_j = \emptyset)}{[\langle i \rangle\phi \wedge \langle j \rangle\psi] \supset \bigvee_{1 \leq k \leq \ell} [\langle i \rangle(\phi \wedge \langle j \rangle\gamma_k) \wedge \langle j \rangle(\psi \wedge \langle i \rangle\gamma_k)]}$

(Ind) $\dfrac{\psi \supset (\phi \wedge \bigodot\psi)}{\psi \supset \Box^*\phi}$

5.6 Completeness/Decidability

Proposition 5.6.1: Every theorem of AX is valid

AX is a sound axiomatic system.

The global axioms (B4) and (B5) assert that an i-improvement does not affect the strategies of other players, and hence the payoffs to players that do not have i in their neighbourhoods are unaffected. (B6) asserts that the payoff for a player i is determined only by the strategies of players in the neighbourhood of i. (B7) is a sanity check, that the social network graph is unaltered by improvement dynamics. (B8) asserts that the formulas are asserted over strategy profiles, with every player making a definite choice. (B9) asserts the correctness of neighbourhood threshold formulas.

The rule (Conc) asserts that players can concurrently improve if their neighbourhoods are disjoint, asserting the existence of a square in the improvement graph. This rule typifies the pattern of reasoning in a "true concurrency" based logic.

The soundness of the axioms mostly follow by the semantic definitions. (B4) follows from the fact that when $\sigma \rightarrow_i \sigma'$, $\sigma[j] = \sigma'[j]$. (B5) and (B6) follow from the definition of π_i. (B7) is valid since the social network does not vary with profiles. (B8) follows from the definition of strategy profiles. (B9) follows from the semantics of $N_{rel\ r}$ modality.

Among the rules, only the soundness of rule (Conc) is interesting. Assume the validity of the disjunction in the premise, and let $M, \sigma \models \langle i \rangle \phi \wedge \langle j \rangle \psi$. Let $\sigma \rightarrow_i \sigma_1$ and $\sigma \rightarrow_j \sigma_2$. By Proposition 2.4, there exists a profile σ_3 such that $\sigma_1 \rightarrow_j \sigma_3$ and $\sigma_2 \rightarrow_i \sigma_3$. Clearly, for some k, $M, \sigma_3 \models \gamma_k$. Hence $M, \sigma_1 \models \phi \wedge \langle j \rangle \gamma_k$, and $M, \sigma_2 \models \psi \wedge \langle i \rangle \gamma_k$. Thus, $M, \sigma \models \langle i \rangle(\phi \wedge \langle j \rangle \gamma_k) \wedge \langle j \rangle(\psi \wedge \langle i \rangle \gamma_k)$, as required.

Soundness of the Axioms

Soundness of Local Axioms

Lemma 5.6.1: Soundness of Axiom $A1_i$

Axiom $A1_i$ is Sound.

Proof. □

Lemma 5.6.2: Soundness of Axiom $A2_i$

Axiom $A2_i$ is Sound.

Proof. □

Lemma 5.6.3: Soundness of Axiom $A3_i$

Axiom $A3_i$ is Sound.

Proof. □

Lemma 5.6.4: Soundness of Axiom $A4_i$

Axiom $A4_i$ is Sound.

Proof. □

Soundness of Global Axioms

Lemma 5.6.5: Soundness of Axiom B0

Axiom B0 is Sound.

Proof. □

Lemma 5.6.6: Soundness of Axiom B1

Axiom B1 is Sound.

Proof. □

Lemma 5.6.7: Soundness of Axiom B2

Axiom B2 is Sound.

Proof. □

Lemma 5.6.8: Soundness of Axiom B3

Axiom B3 is Sound.

Proof. □

Lemma 5.6.9: Soundness of Axiom B4

Axiom B4 is Sound.

Proof. □

Lemma 5.6.10: Soundness of Axiom B5

Axiom B5 is Sound.

Proof. □

Lemma 5.6.11: Soundness of Axiom B6

Axiom B6 is Sound.

Proof. □

Lemma 5.6.12: Soundness of Axiom B7

Axiom B7 is Sound.

Proof. □

Lemma 5.6.13: Soundness of Axiom B8

Axiom B8 is Sound.

Proof. □

Lemma 5.6.14: Soundness of Axiom B9

Axiom B9 is Sound.

Proof. □

Soundness of the Inference Rules

Soundness of Local Inference Rules

Lemma 5.6.15: Soundness of NG_i

Inference Rule NG_i is Sound.

Proof. □

Soundness of Global Inference Rules

Lemma 5.6.16: Soundness of GG

Inference Rule GG is Sound.

Proof. □

Lemma 5.6.17: Soundness of GG

Inference Rule GG is Sound.

Proof. □

Lemma 5.6.18: Soundness of G_i

Inference Rule G_i is Sound.

Proof. □

Lemma 5.6.19: Soundness of $Conc$

Inference Rule $Conc$ is Sound.

Proof. ☐

Lemma 5.6.20: Soundness of Ind

Inference Rule Ind is Sound.

Proof. ☐

Theorem 5.6.1: Completeness of AX

AX provides a complete axiomatization of the valid formulas. Satisfiability of a formula ϕ can be decided in nondeterministic exponential time ($2^{O(m \cdot n)}$, where m is the length of ϕ and n is the number of players).

Proof. Call a formula ϕ *consistent* if $\nvdash \neg\phi$. Call $\alpha \in L_i$ *i-consistent* if $\nvdash_i \neg\alpha$. A finite set of formulas A is consistent if the conjunction of all formulas in A, denoted $\hat{A}$, is consistent. When we have a finite family S of sets of formulas, we write $\tilde{S}$ to denote the disjunction of all formulas $\hat{A}$, where $A \in S$.

For completeness, it suffices to prove that every consistent formula is satisfiable. In fact we show that every consistent formula ϕ is satisfiable in a model of size $2^{O(m \cdot n)}$ where m is the length of ϕ and n is the number of players. From this and soundness, we see a bounded model property: that every satisfiable formula is satisfiable in a model of size exponentially bounded in the size of the formula. This property at once gives a nondeterministic exponential time decision procedure for the logic as well.

Fix a given consistent formula ϕ_0. We confine our attention only to the subformulas of ϕ_0, and maximal consistent sets of subformulas. Towards this, for any

i-local formula $\alpha \in L_i$, let $SF_i(\alpha)$ denote the set of subformulas of α. We assume it to be negation closed and to contain α. $|SF_i(\alpha)| = O(|\alpha|)$. Similarly, for any global formula ϕ, define $SF(\phi)$ to be the set of subformulas of ϕ, which is again negation closed and contains ϕ; further, if $\Diamond^*\psi \in SF(\phi)$ then so also $\bigcirc\Diamond^*\psi \in SF(\phi)$. Again, $|SF(\phi)| = O(|\phi|)$.

Let $R \subseteq SF(\phi_0)$. We call R an **atom** if it is a maximal consistent subset (MCS) of $SF(\phi_0)$. Note that, by rule (GG), for any atom R, $A_i = \{\alpha \mid \alpha@i \in R\}$ is i-consistent.

Let $(A_1, \dots, A_n)$ be the tuple of 'local atoms' in R.

Let AT denote the set of all atoms.

Definition 5.6.1: Word Model

Define $\rightarrow \subseteq (AT \times [n] \times AT)$ by: $R_1 \rightarrow_i R_2$ iff $\{\phi \mid [i]\phi \in R_1\} \subseteq R_2$. Note that when $R_1 \rightarrow_i R_2$ and $\Box^*\phi \in R_1$, $\{\phi, \Box^*\phi\} \subseteq R_2$. Let $G_0 = (AT, \rightarrow)$.

Since ϕ_0 is consistent, there exists an MCS $R_0 \in AT$ such that $\phi_0 \in R_0$. Let G_1 be the induced subgraph of G_0 by restricting to atoms reachable from A_0, denoted $(AT_1, \rightarrow)$.

We have the following observations on G_1.

Truth Lemma

- Every R in AT_1 induces a profile σ_R over $[n]$.
- For every R, R' in AT_1, and $i \in [n]$, if $\{a_j@j \in R \mid j \in N_i\} = \{a_j@j \in R' \mid j \in N_i\}$ then $p@i \in R$ iff $p@i \in R'$.
- For every R in AT_1, if A_i is the i-local atom of R, and $N_{rel\ r}\alpha \in A_i$, then $|\{j \mid e_j \in A_i, \alpha \in A_j\}|\ rel\ r \cdot |N_i|$.
- For every R in AT_1, if $\langle i\rangle\phi \in R$ then there exists $R' \in AT_1$ such that $R \rightarrow_i R'$,

$\phi \in R'$ and the boolean outcome formula in R' is higher in the preference ordering than the one in R.

- For every R in AT_1, if $\Diamond^* \phi \in R$, then there exists an atom R' in AT_1 reachable from R such that $\phi \in R'$.

Lemma 5.6.21: Truth Lemma 1

Every R in AT_1 induces a profile σ_R over $[n]$.

Proof. □

Lemma 5.6.22: Truth Lemma 2

For every R, R' in AT_1, and $i \in [n]$, if $\{a_j@j \in R \mid j \in N_i\} = \{a_j@j \in R' \mid j \in N_i\}$ then $p@i \in R$ iff $p@i \in R'$.

For every R in AT_1, if A_i is the i-local atom of R, and $N_{rel\ r}\alpha \in A_i$, then $|\{j \mid e_j \in A_i, \alpha \in A_j\}|\ rel\ r \cdot |N_i|$.

Proof. □

Lemma 5.6.23: Truth Lemma 3

For every R in AT_1, if $\langle i \rangle \phi \in R$ then there exists $R' \in AT_1$ such that $R \rightarrow_i R'$, $\phi \in R'$ and the boolean outcome formula in R' is higher in the preference ordering than the one in R.

Proof. □

Lemma 5.6.24: Truth Lemma 4

For every R in AT_1, if $\Diamond^* \phi \in R$, then there exists an atom R' in AT_1 reachable from R such that $\phi \in R'$.

Proof. □

Axioms (B8) and (B6) ensure the first two conditions, the third uses the local axiom systems. The (Conc) rule ensures the fourth condition that when $\langle i\rangle\psi \in R$, we can indeed "compute" the maximal consistent set R' such that $R \to_i R'$. The last condition requires an argument such as the one used for propositional dynamic logic.

Define the game $G_{\phi_0} = (n, \Sigma, \Omega, E, \pi)$ by: $\Omega = 2^{P_0}$ where $P_0 = \{p \mid p@i \in SF(\phi_0), i \in [n]\}$; $E = \{(j,i) \mid \mathrm{e}_j \in A_i \text{ of } R_0\}$; $\pi_i(a_{j_1}, \ldots, a_{j_k}) = \{p@i \in R \in AT_1 \mid a_{j_1}@j_1, \ldots, a_{j_k}@j_k \in R\}$, where $k = |N_i|$ and $\{\top@i\}$ if no such atom R exists, where $\top$ stands for "True" Note that for every profile that occurs in G_1, the payoff map is non-trivial. It is well-defined, by the second condition above. We have a model $M_{\phi_0} = (n, \Sigma, 2^{P_0}, E, \pi)$.

Then we show that for every subformula ϕ and every global maximal consistent set $R \in AT_1$, $\phi \in R$ iff $M_{\phi_0}, \sigma_R \models \phi$. This is proved by induction on the structure of ϕ. The axiom system ensures that neighbourhood specifications are consistent across players and the (Conc) rule ensures that when $\langle i\rangle\psi \in R$, we can indeed "compute" the maximal consistent set R' such that $R \to_i R'$.

Since there exists a maximal consistent set $R_0 \in AT_1$ such that the given formula $\psi \in A_0$, we now have: $M_0, \sigma_{R_0} \models \psi$ and we are done.

This is proved by **induction** on structure of ψ.

Base Case .

i $\psi :=$

ii $\psi := a@i$

iii $\psi := p@i$

Induction Case .

i $\psi := \neg\phi$

ii $\psi := \phi_1 \vee \phi_2$

iii $\psi := \langle\exists\rangle\phi$

iv $\psi := \Diamond^*\phi$

□

Thus we have completeness of axiomatization as well as decidability of satisfiability. While we have presented a non-deterministic exponential time decision procedure, we believe that it can be improved to deterministic exponential time: the main idea is to construct the entire atom graph but avoid guessing a good subgraph, but instead delete nodes and edges until what remains is a good subgraph.

5.7 Complexity Bounds

Theorem 5.7.1: Satisfiability problem is DEXPTIME-hard

Proof. We will use a version of the propositional dynamic logic whose satisfiability problem is already DEXPTIME-hard due to (citing paper). All that is remaining for us to prove is show satisfiability of the PDL formula is equivalent to the satisfiability of the translated global outcome formulas, Φ . To go about this, we translate the syntax of formulas from the PDL to the global outcome formulas of the strategy logic and vice versa.

PDL Syntax and Semantics Let P be a set of propositions and A be a finite set of actions. Then the following is the syntax for a version of PDL,

$$\alpha \in \mathbf{\Phi_{PDL}} ::= p, \quad p \in P \mid \neg\alpha \mid \alpha \vee \alpha' \mid \langle a \rangle \alpha, \ a \in A \mid \Diamond^* \alpha$$

A model for PDL formulas will be $M_{PDL} = (W, \rightarrow, \mathbf{val})$ where W is a finite set of worlds. $\rightarrow = \cup_{\{a \in A\}} \rightarrow_a$ where for any $a \in A$, $\rightarrow_a \subseteq W \times W$ and $\mathbf{val} : W \mapsto 2^P$.

For the direction of showing satisfiability from PDL to **global outcome formulas**.

$\Psi : \mathbf{\Phi_{PDL}} \mapsto \Phi$ is defined inductively, such that only for $\Psi(p) = \vee_{\{a \in A\}} p@a$, $\Psi(\neg\alpha) = \neg\Psi(\alpha)$, $\Psi(\alpha \vee \alpha') = \Psi(\alpha) \vee \Psi(\alpha')$, $\Psi(\langle a \rangle \alpha) = \langle a \rangle \Psi(\alpha)$. Since for converting a $\mathbf{\Phi_{PDL}}$ formula into a Φ formula under the Ψ translation involves changing the leaves of the corresponding formula tree we can safely conclude the translation is polynomial in size of the original $\mathbf{\Phi_{PDL}}$ formula with a size overhead in the order of $O(A)$. And the procedure is also polynomial in time with respect to the size of the original $\mathbf{\Phi_{PDL}}$ formula.

Suppose $\alpha \in \mathbf{\Phi_{PDL}}$ is satisfiable in a model $M_{PDL} = (W, \rightarrow, \mathbf{val})$. For the translated global outcome formula $\Psi(\alpha)$, the social network game model where it will be satisfiable will be $M = (|A|, Z, 2^P, \emptyset, \pi = \mathbf{val})$ where Z is any finite set such that $|Z| \leq \frac{\ln(|W|)}{|A|}$. Now, we don't exactly need the entirety of he social network game model. α is satisfiable means, there exists $w \in W$ such that, $M_{PDL}, w \models \alpha$. We would need to work with only the reachability set of w, $\mathbf{Reach}(w) = \{v \in W \mid w \rightarrow^* v\}$. Next details yet to work out

For the direction of showing satisfiability from **global outcome formulas** to PDL.

We observe that the Improvement Graph of any game can be thought of as a labelled transition system which are the models of the propositional dynamic logic. Given a social network model , $M = (n, \Sigma, 2^P, E, \pi)$ and a strategy profile σ, we compute the reachability set of σ, $\mathbf{Reach}(\sigma) = \{\sigma' \in \Sigma^n \mid \sigma \rightarrow^* \sigma'\}$. $A = [n]$ We translate the formulas similarly as above. $\chi : \Phi \mapsto \Phi_{\mathbf{PDL}}$ Next details yet to work out

$\square$

5.8 Bisimulation

5.9 Model Checking

The model checking problem asks, given a social network game $M = (n, \Sigma, 2^P, E, \pi)$, and a formula ϕ whether there is some profile σ such that $M, \sigma \models \phi$. This can be solved in time $2^{O(n)} \cdot |E| \cdot |\phi|$, by explicitly constructing the strategy space and then running a standard labelling algorithm. This is exponential in the number of players, which is unavoidable since computing Nash equilibrium in social network games is known to be NP-hard ([**?**]).

5.10 Discussion

Chapter 6

Modal Logic for Large Games

This chapter will explore trying to tie in ideas we set in motion in our last chapter on social network threshold games, and applying it to a setting where the number of players are arbitrary. How can we attain a possible modelling of such a scenario without venturing into the incompleteness results of using any First Order logic for our reasoning. The answer comes to us in the form of **Implicit Quantification** modal logics. We will show how that logic is rich enough to grant us our expressibility requirements and we are still able to produce a decidable axiomatization procedure.

6.1 Background

To motivate large games let us take a concrete simple example. Let us try to abstractly model the stock market on which the pillars of our current capitalist society rest.

Let's take this simple abstraction of the stock market. A stock is a possible appreciating asset class. It is considered a part of company that is given out by the company management to the people called shareholders in exchange for capital that can keep the company functioning. Shareholders are players of the market who are involved in two simple actions of buying and selling stocks. The idea for every shareholder also known as investor is to buy a stock of a company when the value of the stock is higher than the price at which it is being sold and finally selling off the stock when he/she sees the value of the stock decreasing in contrast to its value. Thus making a profit in the market.

So the fundamental economical problem of our society is how do we determine the **value** of an object in the capitalist world. How much should a painter get for their art? How much should a doctor get for their service? And in the same vein, how much should someone delving in abstract theoretical computer science be compensated for proving a theorem that need not necessarily have immediate applications! The proxy to value is **price**, which is easier to measure.

When set in a word where demand-supply is a maxim, then price can be thought of as being proportional to the number of other players who value the object. Price discovery can be done through the mechanisms of auctions. It is inherent in the auction format that the players who value the item will bid for ever increasing prices to the object of desire. So, the price of a stock at any point in time depends on how much other market players value it as well. And that would mean in particular it's less to do with the person being able to deductively deduce a choice and more to do with being able to believe in a certain choice that would also tend to be intriguing to others.

Thus we can slowly see that this is a Large Game. A game where a player does not in general know how many other players he/she is competing with, or even how many players are there doing the same reasoning as he/she is doing! But, the payoff for the particular player is crucially subjected to the number of players thinking similarly as him/her.

Being able to model such a phenomena requires mathematics to be able to capture such nuances in reasoning in a human network.

Of course the real life scenario is more complicated , and we have a presented a very simple abstraction here. In reality there are elements of different level of players who have differing knowledge levels. And as any human activity would have, there are high rates of collusion amongst the upper echelons of society making the lower tier play their game.

Hopefully the nature of Large Games are clear and other more relatable examples like the policy design during the covid outbreak can be seen in such a light as well.

6.1.1 Contributions of this work

This work is situated in this arena of formal reasoning about games. We are interested in reasoning about how players reason individually when the outcome is collective. Players have common knowledge of how choice distributions determine payoffs and reason about the effect of player deviations. They reason not about the actual number of players that choose an option, but about what fraction of players make that choice. We wish to study paths to equilibrium in such a context.

We study *improvement dynamics* ([**?**]) in large games. This is a graph structure on strategy spaces, where nodes are profiles, and player labeled edges represent a deviation by a player to get an improved outcome. Nash equilibrium is simply a sink node in this graph. In the case of large games, it is appropriate to consider improvement not by single players but by several players at the same time, shifting choice distributions significantly.

The notion of *player types* is central to game theory: it specifies the beliefs that a player has on other players' behaviour, and how the player would act contingently. Logically speaking, they specify player responses to postulated strategy distributions. A typical player type may describe a profile where at least half the players choose a and at most a quarter choose b, from which an improvement path may lead to a profile where less than a quarter choose a and more than two-thirds choose b. So called *herd behaviour* and *runaway phenomena* ([**?**]) can be thus reasoned about in a logic with such player types and reasoning about transitive closure.

Since the number of players is finite but unbounded, it is natural to expect a first order logic with quantification over player variables. As it turns out, such a logic is undecidable and has no recursive axiomatisation, hence we study a propositional fragment of such a logic, with constant symbols denoting players, and *implicit quantification* over players. Distribution constraints can still force a lower bound on the number of players in the games. We show that the logic is decidable and present a complete axiomatisation of the valid formulas. We present a bisimulation characterisation of the logic and model checking results as well.

The principal technical challenge in such inference is that while a formula is interpreted in a choice profile (an n-tuple where n is the number of players), the formula cannot specify the number n. Thus for a decision procedure or for a completeness proof, we need to *compute* the number of players from the given formula and work out the dynamics. Moreover, when we have implicit quantification, we may know that some players *some* players deviate without knowing which ones, and then ensuring consistency between profiles is challenging.

Such analysis has an interesting consequence. Logical player types induce an equivalence on players in games, and thus analysis of a large game can be reduced to one "between" player types. We also get a dual use for the logic on large games: the dynamic aspect of the logic can be used to study the equilibrium dynamics of a given large game; dually, the logic can be used as a compact *specification mechanism* for large games which could in general be quite unwieldy to specify explicitly.

6.1.2 Related work

The logical foundations of game theory have been studied for some year now, and modal characterisation of equilibria presented by several researchers ([?], [?], [?]). The seminal work of Parikh ([?]) presented propositional game logic, a way of reasoning about game composition. Strategic reasoning in extensive form games has been studied in depth in the context of logics of strategic ability such as Alternating Temporal Logics and other process based logics ([?], [?], [?]. [?], [?]).

However, in general, these studies typically fix n and study n-player games, whereas our principal point of departure in this paper is in the study of games with unboundedly many players. Moreover these studies associate payoffs with strategy profiles rather than with distributions as we do here. In our earlier work, dynamics in large games modelling imitation behaviour([?]) and dynamic game forms ([?]) have been studied.

How choices diffuse across networks ([?], [?], [?]) is a study essential in spirit to ours but distributions again make the principal difference. Majority dynamics is implicit in our reasoning, and it has been extensively studied as a class of games ([?]).

Large games arise naturally in *evolutionary game theory* ([?]), in which long range behaviour of a population is determined by distributions of player types among the population. Such games bear a close relationship to the class of *population protocols* ([?]) in distributed computing, where again interactions do not depend on the identity of the interacting processes. In the logic we present, we do use player identities to describe strategisation by players, but the strategies themselves depend only on distributions.

6.2 Large games

Let N denote a finite set of players. We will often consider N to be the set $\{1, 2, \ldots, |N|\}$, and refer to players i, j etc in this set. In general, we will want to consider subsets of players, and use u, v etc to denote these subsets.

To talk about outcomes of games, rather than work with quantitative payoffs, we use a propositional language to describe abstract conditions on outcomes. Fix $\mathcal{P}$, a countable set of atomic propositions. $\mathcal{P}$ consists of qualitative descriptions of outcomes. Every game involves only a finite set $P \subseteq \mathcal{P}$ of these propositions. In general an outcome for a player is a boolean formula over $\mathcal{P}$ (which is a set of propositions).

Fix a finite set of actions (or choices, or strategies) which we denote by $\mathbf{\Gamma}$. An action distribution for n players is a $|\mathbf{\Gamma}|$ tuple of integers $\mathbf{y} = (y_1, \ldots, y_{|\mathbf{\Gamma}|})$ such that $y_j \geq 0$ and $\Sigma_{j=1}^{k} y_j = n, \;\; 1 \leq j \leq |\mathbf{\Gamma}|$. That is, the jth component of $\mathbf{y}$ gives the number of players who play action a_j. Let $\mathbf{Y}[n]$ denote the set of all action distributions for n players.

We have an outcome function $\omega : \mathbf{Y}[n] \to (2^P)^{|\mathbf{\Gamma}|}$ which gives the truth of the outcome propositions P according to the action distribution of the players: the idea is that every player choosing an action a gets the same outcome.

Thus formally, a game $\mathcal{G}$ over action set $\mathbf{\Gamma}$ is a tuple $\mathcal{G} = (N, P, \omega, (\preceq_i \mid i \in N))$, where P is a finite subset of $\mathcal{P}$, ω is an outcome function, and $\preceq_i$ is a partial ordering on 2^P, for each player i specifying the preferences of each players. Since every element of 2^P can be viewed as a boolean formula over P, we will consider it as an ordering on boolean formulas. $\prec_i$ denotes the corresponding strict preference for each player i.

A configuration of the game is given by $\mathbf{a} \in \mathbf{\Gamma}^N$ is an *action profile*, which specifies a choice of action for each player, which induces a distribution and hence, implicitly, an outcome for each player. Let Σ denote the set of strategy profiles Γ^N. Note that ω induces a map $\hat{\omega} : \Sigma \to (2^P)^N$.

In general we use σ, σ' etc to denote profiles, and $\sigma[i]$ denoting the i^{th} element of σ, which is interpreted as the choice of player i. We write $\sigma \prec_i \sigma'$ to denote that $\hat{\omega}(\sigma)[i] \prec_i \hat{\omega}(\sigma')[i]$.

When players act individually, rationality assumptions dictate that, at any profile, if a player can get a better outcome by unilaterally altering the choice, the player would do so. A simple way to study such reasoning is given by the improvement graph dynamics defined as follows.

Definition 6.2.1: Improvement Graph originally

A **player improvement graph** $I_{\mathcal{G}}$, associated with the game $\mathcal{G}$, is the graph $G_{\mathcal{G}} = (\Sigma, E_{\mathcal{G}})$, where $\Sigma = \Gamma^N$, and $E_{\mathcal{G}} \subseteq (\Sigma \times N \times \Sigma)$ is the player labelled edge relation given by: $(\sigma, i, \sigma') \in E_{\mathcal{G}}$ iff $\sigma \prec_i \sigma'$, and $\sigma[j] = \sigma'[j]$, for all $j \neq i$.

We have an i-labelled edge from a strategy profile to another, if player i can unilaterally deviate from the former to the latter to get an improved payoff. Note that at a profile, there can be different i-improvement edges leading to different profiles (with perhaps incomparable outcomes). A path in $I_{\mathcal{G}}$ is an improvement path.

However, in large games, unless the outcome map has major discontinuities, deviation by a single player may cause no change in outcome at all. We may end up having a player improvement graph with very few edges. Moreover consider a swap game where player i deviates from choice a to choice b iff another player j deviates from b to a. This would result in the same distribution and hence may not result in improvement for either player, and hence 'should' not be considered meaningful deviations. We therefore consider deviation by *teams* of players. Note that they need not be acting together intentionally; it is sufficient that they all deviate in different directions, but improving the outcome for all of them.

Definition 6.2.2: Improvement Graph For large games

An **improvement graph** $I_\mathcal{G}$, associated with the game $\mathcal{G}$, is the graph $I_\mathcal{G} = (\Sigma, E_\mathcal{G})$, where $\Sigma = \Gamma^N$, and $E_\mathcal{G} \subseteq (\Sigma \times 2^{|N|} \times \Sigma)$ is the labelled edge relation given by: $(\sigma, u, \sigma') \in E_\mathcal{G}$ iff for all $i \in u$, $\sigma \prec_i \sigma'$, and for all $j \notin u$, $\sigma[j] = \sigma'[j]$.

We have a u-labelled edge from a strategy profile to another, if every player i in u can unilaterally deviate from the former to the latter to get an improved payoff. Note that at a profile, there can be different u-improvement edges leading to different profiles (with perhaps incomparable outcomes). A path in $I_\mathcal{G}$ is an improvement path. (When the set u is a singleton $\{i\}$, we will denote the edge as $\rightarrow_i$.)

Note that the improvement graph can have cycles. For instance, consider the two-player game of matching pennies: both players call heads or tails, the first player wins when the results match, and the second wins when there is a mismatch. We then have the cycle $(H, H) \rightarrow_2 (H, T) \rightarrow_1 (T, T) \rightarrow_2 (T, H) \rightarrow_1 (H, H)$.

In any particular game, N is fixed as well as the size of Γ and hence $I_\mathcal{G}$ is a finite directed graph, though its size is exponential in the number of players. It contains a good deal of interesting information about the game $\mathcal{G}$. For instance consider a *sink* node of $I_\mathcal{G}$, which has no out-going edge: it is easy to see that a strategy profile is a sink node if and only if it constitutes a *Nash equilibrium*, from which no player has any incentive to deviate.

In the analysis of games, we are typically interested in questions like whether the game has a Nash equilibrium, whether every improvement path is finite, whether an equilibrium profile is reachable from every profile, etc. Algorithmically all these questions are efficiently solvable in terms of the size of the improvement graph, which is unfortunately large.

When we consider algorithmic study of large games, an important source of difficulty is the *game specification* itself. For an n-player game, the outcome function is a map from all possible choice distributions over n to the outcome set. This is clearly difficult since we need to consider the number of $|\mathbf{\Gamma}|$-way partitions of n, and this can be very large. In practice, we would like a partial specification mechanism for the outcome function given as a table of distribution vectors with outcomes. This is sufficient to generate the improvement graph. Moreover, player preferences need to be given only over the distribution vectors present in the table. Such specifications may be considered partial game specifications.

Note that the improvement graph dynamics induces a natural **game equivalence**: we can consider two games to be equivalent if they have a similar (not necessarily isomorphic) improvement structure which can be characterized by an appropriate notion of bisimulation.

6.3 Logics for large games

How do we express strategisation by players in large games in logic? Towards this, we first observe that the best response of a player is not to what other players choose, but how many players make a particular choice. Therefore deviation by a player is also based on the expectation of how many other players would deviate as well. Such reasoning can be formalised by assertions of the form "If half or more of the players play a then I should respond with b". Since a player need not know the actual number of players in the game, it makes sense to work with ratios rather than actual numbers. Moreover, by recursion over such formulas, we can build player types. If the above assertion defines a player type t, we can have another type t' which asserts: "If at least one third of the players are of type t, then play a", and so on.

A more tricky question is how we are to refer to players at all in the syntax of the logic. Note that every game comes with a set of players, but player identities are immaterial in a large game. We therefore want only a way of abstractly specifying player types rather than work with actual player identities. The logical device of using *variables* and *quantification* over them is the obvious solution: we would like to assert: $\forall x.(\alpha(x) \implies p@x)$ where ϕ is some type specification, expressing that every player of type α gets an outcome p.

6.3.1 Player types as formulas

The main characteristic of strategisation in large games is player response to potential choice distributions. Specifically, $\sharp\ a\ rel\ r$ is a *distribution constraint*: it specifies a constraint on the set of players of choosing a. For instance $\sharp\ a\ \leq\ r$ asserts that at most an r-fraction of players choose a.

Distribution Constraint Formula

The syntax of distribution constraint formulas is given by:

$$\delta ::= \sharp\ a\ rel\ r,\ a \in \mathbf{\Gamma} \mid \neg\delta \mid \delta \vee \delta'$$

where $a \in \mathbf{\Gamma}$, $rel \in \{\geq, \leq, <, >\}$, r is a rational number, $0 \leq r \leq 1$.

It is easy to see that a distribution constraint formula specifies a finite set of distribution constraints of the form $\sharp\ a\ rel\ r$.

Fix a game. Distribution constraint formulas are interpreted on its strategy profiles.

- $\sigma \models_d \sharp\ a\ rel\ r$ if $\frac{|\{j|\sigma[j]=a\}|}{|N|}\ rel\ r$.

- $\sigma \models_d \neg\delta$ if $\sigma \not\models_d \delta$.

- $\sigma \models_d \delta \vee \delta'$ if $\sigma \models_d \delta$ or $\sigma \models_d \delta'$.

We say $(rel\ r)$ *entails* $(rel'\ r')$ when $r \leq r'$ and either $rel = rel' = \ \leq$ or $rel = \ <$ and $rel' = \leq$, or $r \geq r'$ and either $rel = rel' = \ \geq$ or $rel = \ >$ and $rel' = \ \geq$.

We use the abbreviation $\sharp\ a = r$ to denote $\sharp\ a \leq r \wedge \sharp\ a \geq r$. Let $k = |\mathbf{\Gamma}|$ and $distr(r_1, \ldots, r_k)$ stand for $\bigwedge_{\ell=1}^{k} \sharp a_\ell = r_\ell$, where $\Sigma_\ell\ r_\ell = 1$. This is a complete distribution, independent of the number of players.

For $\mathbf{\Gamma} = \{a, b\}$, some distribution constraint formulas that are clearly unsatisfiable, like: $\sharp\ a < 0$, $\sharp\ a > 1$, $\sharp a > \frac{1}{2} \wedge \sharp b > \frac{1}{2}$, and $\sharp a < \frac{1}{2} \wedge \sharp b < \frac{1}{2}$.

A satisfiable instance is : $\sharp a \leq \frac{1}{2} \wedge \sharp b \leq \frac{1}{2}$.

In general, $\bigwedge_{a \in \mathbf{\Gamma}} \sharp a \leq r_a$ is unsatisfiable when $\Sigma_{a \in \mathbf{\Gamma}}\ r_a < 1$ and $\bigwedge_{a \in \mathbf{\Gamma}} \sharp a \geq r_a$ is unsatisfiable when $\Sigma_{a \in \mathbf{\Gamma}}\ r_a > 1$.

On the other hand when we have a satisfiable formula like $\sharp a = \frac{3}{7} \wedge \sharp b = \frac{2}{5} \wedge \sharp c > 0$, the number of players, which needs to be an integer, has to be at least 35. One possible distribution that satisfies the constraint is $(15, 14, 6)$, but clearly so also is $(60, 56, 24)$. It is in fact an easy combinatorial exercise to decide when a set of distribution constraints is satisfiable.

Proposition 6.3.1: Satisfiability of Distribution Constraint Formula

Let J be a finite set of distribution constraint formulas. If J is satisfiable, then there is a least number n and a finite set of $\mathbf{\Gamma}$-distributions over n each of which satisfies J.

The proof of this proposition is by considering a variable x_a for each a in $\mathbf{\Gamma}$, and setting up a system of inequalities given by distribution constraints. If this system has a solution in non-negative integers, there is one that minimises $\Sigma_a x_a$. For this we convert the constraints into a normal form, collecting all the constraints into a single one of the form $x_a \; rel \; r_a$, and then take the least common multiple of denominators of all r_a's.

Indeed, the algorithm can output not only n but also a set of $\mathbf{\Gamma}$-distributions over n that satisfy J.

This exercise has a pleasant consequence. Assume that all the rationals in formulas are given in binary, if m is the maximum length of any integer in the distribution constraints, and there are ℓ-many integers occurring, then the least number n referred to above needs only ℓm bits to specify. (Note that the set $\mathbf{\Gamma}$ is fixed for the language, and hence $|\mathbf{\Gamma}|$ is treated as a constant.) Note that though n is described using ℓm bits, the number itself is of the order of $2^{\ell m}$. Thus a distribution constraint formula δ can 'force' at most $2^{|\delta|^2}$-many players to exist.

For player types, we will use formulas of the form $\delta \rightarrow a$ which asserts a type of player who responds with a when the distribution constraint δ holds.

As an example, $\sharp \; a \; > \frac{2}{3} \implies b$, asserts a type of player who responds with b when more than two-thirds of the players choose a. Moreover we can have type assertions of the form $\sharp \; a \; > \frac{2}{3} \implies b$ or even better, $(\sharp \; a \; > \frac{2}{3} \wedge \sharp \; c \; > \frac{1}{10}) \implies b$. Note that the types can construct complex patterns of hypothetical responses.

Player Type Formula

The syntax of player types is given by:

$$\alpha ::= \delta \rightarrow a, \; a \in \mathbf{\Gamma} \mid \neg\alpha \mid \alpha \vee \alpha'$$

where δ is a distribution constraint formula.

Type formulas are interpreted at profiles, for individual players. As above, fix a game. Let σ be a profile and i a player.

- $\sigma, i \models_t \delta \rightarrow a$ if $(\sigma \models_d \delta)$ implies $\sigma[i] = a$.
- $\sigma, i \models_t \neg\alpha$ if $\sigma, i \not\models_t \alpha$.
- $\sigma, i \models_t \alpha \vee \alpha'$ if $\sigma, i \models_t \alpha$ or $\sigma, i \models_t \alpha'$.

While the basic strategy specification is a strategic response, unconditional choice is easily coded up. $(\sharp a \geq 0) \rightarrow a$ specifies a vacuous constraint which is equivalent to simply asserting a. Therefore we will simply consider every $a \in \mathbf{\Gamma}$ as a type formula as well.

Again some type formulas can be unsatisfiable: for instance, $(\sharp a = 1) \rightarrow b$, where a and b are distinct. It is easy to see that the previous proposition can be employed to prove a similar proposition on type formulas as well.

Proposition 6.3.2: Satisfiability of Type Formulas

Let J be a finite set of type formulas. If J is satisfiable, then there is a least number n and a finite set of $\mathbf{\Gamma}$-distributions over n each of which satisfies J.

This suggests that the type analysis in a given game with unboundedly many players can be reduced to one in which the number of players is bounded (exponentially) by the size of the type specification.

We formalise this intuition as follows. Let $\mathcal{G} = (N, P, \omega, (\preceq_i \mid i \in N))$ be a game. We say that a type formula α is **realisable** in $\mathcal{G}$ if there exist $\sigma \in \Sigma$ and $i \in N$ such that $\mathcal{G}, \sigma, i \models_t \alpha$. Note that there are at most 2^{2^k}-many realisable types (upto logical equivalence), where $k = |\mathbf{\Gamma}|$, in any game.

We also get an interesting notion of *type equivalence*. In a game $\mathcal{G}$, we say that two players i and j are type equivalent if for all $\sigma \in \Sigma$, and all type formulas α, $\sigma, i \models_t \alpha$ iff $\sigma, j \models_t \alpha$. Thus, we can quotient any given game by type equivalence and work with boundedly many players, the bound depending only on the type specification.

Another consequence of this analysis is that we we can limit outcome specification.

Proposition 6.3.3: Satisfiability of Player Type Formulas

Given a finite set of player type formulas J, if J is satisfiable, then we can provide a table of distributions B such that the formulas in J are realised in a game whose partial specification is provided by outcomes over B.

Elucidate on the Proposition

A Natural Example for Player Types

Have to fill in

The model checking problem for a logic is typically phrased as follows: given a finite model M and a formula α, check that α is true of M. However, we use logic in a different direction. Type formulas provide succinct specifications for distributions that are necessary for analysis. Hence the game to be studied can be limited to a partial specification where outcomes are provided only for these distributions. Thus, only a 'small' part of the game model may need to be presented.

6.3.2 A logic of quantification

What we have discussed so far are assertions that are evaluated at strategy profiles. But we also want to reason about player improvement and paths to equilibrium. However, improvement involves deviation by sets of players, which motivates us to consider *set variables* and quantification over them as well.

Thus we are led to a two-level syntax of formulas. We have already considered type formulas at one level, Fix a set of player variables $V = \{x_0, x_1, \ldots\}$ with x, y etc ranging over V and a set of team variables $TV = \{X_0, X_1, \ldots\}$ with X, Y etc ranging over TV. We can this logic the Monadic Second Order Logic of Strategization in Large Games, $MSO - Strat$, for short. (Here the term 'monadic' refers to set quantification.)

The syntax of strategisation formulas is given by:

$$\phi \in \Phi ::= \alpha@x \mid p@x \mid x \in X \mid x = y \mid \neg\phi \mid \phi \vee \phi' \mid \langle X \rangle \phi \mid \Diamond^* \phi \mid \exists x.\phi \mid \exists X.\phi$$

where α is a type formula, $p \in \mathcal{P}, x \in V$ and $X \in TV$.

We do not have constants in the syntax but adding player names and interpreting them in a given model appropriately can be added on standard lines.

Note that we can easily specify that a team is a singleton: $\exists x.x \in X \wedge \forall y.(y \in X \implies y = x)$ would do it. Hence we could abbreviate $\exists X.\exists x.Sing(X) \wedge x \in X \wedge \langle X \rangle \phi$ by $\exists x.\langle x \rangle \phi$.

For semantics, the game model is given by $M = (N, P, \omega, \preceq, \delta)$ where the map $\delta = (\delta_1, \delta_2)$, $\delta_1 : V \rightarrow N$ assigns every variable to a player, and $\delta_2 : TV \rightarrow 2^N$ assigns every set variable to a set of players.

- $M, \sigma, \delta \models \alpha@x$ if $\sigma, \delta_1(x) \models_t \alpha$.
- $M, \sigma, \delta \models p@x$ if $p \in \hat{\omega}_{\delta_1(x)}(\sigma)$.
- $M, \sigma, \delta \models x \in X$ if $\delta_1(x) \in \delta_2(X)$.
- $M, \sigma, \delta \models x = y$ if $\delta_1(x) = \delta_1(y)$.
- $M, \sigma, \delta \models \neg\phi$ if $M, \sigma, \delta \not\models \phi$.

- $M, \sigma, \delta \models \phi \vee \psi$ if $M, \sigma, \delta \models \phi$ or $M, \sigma, \delta \models \psi$.

- $M, \sigma, \delta \models \langle X \rangle \phi$ if there exists σ' such that $\sigma \rightarrow_{\delta_2(X)} \sigma'$ and $M, \sigma', \delta \models \phi$.

- $M, \sigma, \delta \models \Diamond^* \phi$ if there exists σ' such that $\sigma \rightarrow^* \sigma'$ and $M, \sigma', \delta \models \phi$.

- $M, \sigma, \delta \models \exists x\ \phi$ if there is some $i \in N$ such that $M, \sigma, \delta_{[x \rightarrow i]} \models \phi$.

- $M, \sigma, \delta \models \exists X\ \phi$ if there is some $u \subseteq N$ such that $M, \sigma, \delta_{[X \rightarrow u]} \models \phi$.

Above $\delta_{[x \rightarrow i]}$ is the map $\delta' = (\delta'_1, \delta_2)$, where $\delta'_1(y) = i$ when $y = x$ and $\delta'_1(y) = \delta_1(y)$ otherwise. Similarly, $\delta_{[X \rightarrow u]}$ is the map $\delta' = (\delta_1, \delta'_2)$, where $\delta'_2(Y) = u$ when $Y = X$ and $\delta'_2(Y) = \delta_2(Y)$ otherwise.

The logic is expressive enough to force infinitely many players. For instance, the formula $\Box^* \exists x.(\langle x \rangle \top \wedge \Box^* [x] \bot)$ is satisfiable only in game models with infinitely many players. But since we consider only large games with unbounded but finitely many players, such formulas are *unsatisfiable* in our setting. But we can prove that even over models with only *finitely many players* the satisfiability problem for this logic is undecidable. We will do this in a later section. Here it suffices to point out that the mix of quantifiers and modalities makes the logic way too expressive, even if we had only first order variables.

We therefore consider an alternative propositional formalization.

6.3.3 Implicit quantification

We note that type formulas implicitly define players for us. So we only need to add player names to the logic so that formulas can be interpreted on strategy profiles. However, these names correspond to player types and hence we cannot form sets of types to refer to actual sets of players, which would be needed for specifying improvement dynamics. For this, we take a cue from **Implicitly Quantified Term-Modal logic** ([?]) and leave the modality to implicitly specify a set of players. This leads us to an interesting logic.

Recall that the logic is parameterised by the strategy set $\mathbf{\Gamma}$. Let C denote a set of constant symbols, intended to be interpreted as player names. We let i, j etc to range over C below.

The syntax of formulas is given by:

$$\phi \in \Phi ::= \delta \mid a@i \mid p@i \mid \neg\phi \mid \phi \vee \phi' \mid \langle\exists\rangle\phi \mid \Diamond^*\phi$$

where $p \in \mathcal{P}, a \in \mathbf{\Gamma}, i \in C$, δ is a distribution constraint and α is a type formula. We call this the Propositional Modal Logic of Implicit Quantification for Strategization in Large Games, $MIQ - Strat$, for short.

The atomic formula $a@i$ asserts that player i chooses a. $p@i$ specifies that player i receives outcome p. it is easily seen that all type formulas can be expressed in the logic. The modality $\langle\exists\rangle\phi$ asserts that there is a team u that has an improvement edge to a profile in which ϕ holds. Note that by each player in u switching to some different choice, the outcome is affected positively for everyone in u. $\Diamond^*\phi$ says that the outcome ϕ is reachable by a finite path from the current profile.

The dual modalities are also interesting: $[\forall]\phi$ asserts that every deviation by every team results in a profile that satisfies ϕ: in some sense, this means that ϕ is an invariant. $\Box^*\phi$ asserts that ϕ is an invariant property, true at all reachable states.

Semantics The formulas are interpreted over strategy profiles of large games. A model is a large game $M = (N, P, \omega, (\preceq_i, i \in N), \iota)$ where $\omega : \mathbf{Y}[|N|] \to (2^P)^{\mathbf{\Gamma}}$ is the outcome function, and $\iota : C \to N$ is an *injective* interpretation of constant symbols. Implicitly for us different "player names" refer to different players, We refer also to the induced outcome function for each $a \in \mathbf{\Gamma}$ by $\hat{\omega}_a : \Sigma \to 2^P$ where Σ is the set of all strategy profiles.

The semantics is given by assertions of the form $M, \sigma \models \phi$, read as ϕ is true of the strategy profile σ in model M. The semantics of improvement formulas can then be defined as follows. Below, let $\to^* = (\cup_u \to_u)^*$, the reflexive transitive closure of the union of the improvement edge relations.

- $M, \sigma \models \delta$ if $\sigma \models_d \delta$.
- $M, \sigma \models a@i$ if $\sigma[\iota(i)] = a$.
- $M, \sigma \models p@i$ if $p \in \hat{\omega}_{\iota(i)}(\sigma)$.
- $M, \sigma \models \neg\phi$ if $M, \sigma \not\models \phi$.
- $M, \sigma \models \phi \vee \psi$ if $M, \sigma \models \phi$ or $M, \sigma \models \psi$.
- $M, \sigma \models \langle\exists\rangle\phi$ if there exists $u \subseteq N$ and σ' such that $\sigma \to_u \sigma'$ and $M, \sigma' \models \phi$.
- $M, \sigma \models \Diamond^*\phi$ if there exists σ' such that $\sigma \to^* \sigma'$ and $M, \sigma' \models \phi$.

The other boolean connectives $\wedge$, $\implies$ and $\equiv$ are defined in the standard way. The dual modalities are: $[\forall]\phi = \neg\langle\exists\rangle\neg\phi$ and $\Box^*\phi = \neg\Diamond^*\neg\phi$. Their semantics are as follows :

- $M, \sigma \models [\forall]\phi$ iff for all $u \subseteq N$ and for all σ', if $(\sigma \to_u \sigma')$ then $M, \sigma' \models \phi$
- $M, \sigma \models \Box^*\phi$ iff for all σ' if $\sigma \to^* \sigma$ then $M, \sigma' \models \phi$

Fix a constant symbol i and a propositional symbol p. We use $\top$ to denote the tautology $p@i \vee \neg p@i$ and $\bot = \neg\top$ for a contradiction. (Similarly we also use $\top@i$ and $\bot@i$.)

Note that for the truth of a formula ϕ in a model M at a profile σ, it suffices to restrict ourselves to only profiles reachable from σ. We therefore confine our attention to models which are *pointed*: these are "rooted" improvement graphs with root σ such that the entire graph is the reachability set of σ.

The $\langle\forall\rangle$ modality

In the logic of implicit modal quantification, there is another modality: $\langle\forall\rangle$ and it's dual $[\exists]$ with the semantics:

- $M, \sigma \models \langle\forall\rangle\phi$ if for every $u \subseteq N,$ there exists σ' such that $\sigma \rightarrow_u \sigma'$ and $M, \sigma' \models \phi$
- $M, \sigma \models [\exists]\phi$ if there exists $u \subseteq N$ and for all σ', if $\sigma \rightarrow_u \sigma'$ then $M, \sigma' \models \phi$

The modality $\langle\forall\rangle\phi$ asserts that *every* team of players has an improvement at that profile, making this a 'maximally non-Pareto' situation. Its dual modality $[\exists]\phi$ asserts that there is a team of players for whom every deviation 'together' leads to a profile that satisfies ϕ.

This is logically very interesting and adds a good deal of expressiveness to the logic. However, this kind of reasoning would be of interest when we consider coalitions and co-operative or co-ordinated behaviour of players. Since we are only considering unilateral deviation by players (albeit concurrently by many), we stick with the simpler logic with only one modality of implicit quantification.

Examples of such Formulas for Coordination Games

Examples of Large Games describable in MIQ-STRAT

The formula $NE \stackrel{\text{def}}{=} [\forall]\bot$ is interesting: it specifies that the profile it is asserted in is a Nash equilibrium, since no player can improve on their outcome. $\Diamond^* NE$ says that an equilibrium is reachable, and $\Box^*\Diamond^* NE$ asserts that from every reachable profile, there is a path to an equilibrium profile. Note that we cannot express the notion that a player's response is the best response to the remaining players' choices.

Let $\mathbf{\Gamma} = \{a, b, c\}$, where, following Blonski ([**?**]), they represent stocks. The formula $\sharp a > \frac{1}{2} \implies a@i$ specifies a trader who owns stock a only if it is favoured by a majority of the population. Now consider players who make the same choices only when they are guaranteed to receive the same outcome: the formula $\Box^* \bigwedge_{a \in \mathbf{\Gamma}} (a@i \equiv a@j \implies p@i \equiv p@j)$ specifies this.

Consider a special class of games where the set of propositional symbols is the set of rational numbers in $[0, 1]$. Consider a game $\mathcal{G} = (N, P, \omega)$ where we have a finite set of rational payoffs. Now consider the formula:

$$\Box^* \bigwedge_{a \in \mathbf{\Gamma}} (a@i \implies (\sharp\ a = r \implies r@i))$$

It says that player i is of *matching type*: the payoff to player i is exactly the proportion of the population whose choice matches i's.

$$\Box^* \bigwedge_{a \in \mathbf{\Gamma}} (a@i \implies (\sharp\ a = r \implies (1 - r)@i))$$

specifies a player whose type is *mismatching*. Kalai [**?**] discusses such generalizations of matching pennies games.

Voting rules ([**?**]) often use payoffs that are matching: that is, the satisfaction of a voter V who voted for candidate C is given by the fraction of the votes that C received, specified by the same formula as above.

Bisimulation

Formulas in propositional modal logic preserve bisimulation. Further, over finitely branching structures, bisimulation and modal elementary equivalence coincide. Thus bisimulation can be seen as elementary equivalence in modal logic. We will take this up in a subsequent section.

Model checking

The Model checking problem asks, given a finite model, a state and a formula, whether the formula holds at that state in the model. For logics on large games, we are presented with a game and a formula, and we may ask whether the formula holds at a specific strategy profile of the game.

For $MIQ - STRAT$, we can easily modify the standard labelling procedure of modal logic with transitive closure to provide a model checking algorithm. But since the input is the game and the model checking is to be done on the improvement graph, we obtain the running time of $O((|\phi|^2 \times |\Gamma|^N))$, where ϕ is the given formula. However, since the specification induces an equivalence relation on the players, there is scope to consider a quotiented model and thus obtain a "better" running time of $O(2^{2^{|\phi|}} \cdot |N|^c)$ for some constant c. Such a procedure would be better when the size of N is large.

For $MSO - Strat$, since the number of players is finite, we can first eliminate the quantifiers, replacing them by conjunctions and disjunctions and use the model checking algorithm for the propositional fragment, resulting in a formula size blow up of $O(|\phi| \times 2^{|N| \times qr(\phi)})$. Therefore the algorithm takes $O(|\phi|^2 \times 2^{2 \times |N| \times qr(\phi)} \times |\mathbf{\Gamma}|^{|N|})$ time.

For these logics, it may be more interesting to model check invariant properties that hold for *every* strategy profile, or realisability, of properties that hold at *some* strategy profile.

6.4 Decidability

We now consider the satisfiability problem for the logics we have discussed. While the propositional modal logic of implicit quantification for large games is decidable, the monadic second order logic of strategization for large games is undecidable, as one may expect.

6.4.1 MIQ-Strat

We first show that the logic $MIQ - Strat$ has the *bounded players* property.

Lemma 6.4.1: $MIQ\ STRAT$ **satisfiability**

If a formula ϕ of $MIQ - Strat$ is satisfiable, then it is satisfiable in a game model in which the number of players is bounded by $2^{O(|\phi||\mathbf{\Gamma}|)^2}$.

The proof principally follows the proposition 2 which asserts that a type formula can force only $2^{O(m^2)}$ many players where m is the maximum number specified in the distribution constraints. Thus we need to consider strategy profiles of bounded length. On such profiles, a standard filtration argument, along the lines of the one for Propositional Dynamic Logic ([**?**]) shows that every satisfiable formula is satisfiable in a model of (exponentially) bounded size.

Theorem 6.4.1: $MIQ - SAT$

The satisfiability problem for the logic $MIQ - Strat$ can be decided in NExptime.

Rather than proving this theorem in detail, we will later present a proof that at once proves completeness of the presented axiom system as well as the decidability of satisfiability.

6.4.2 MSO-Strat

We can prove that even over *finite models* the satisfiability problem for $MSO-Strat$ is undecidable. In fact, we will need neither set variables nor the set quantification, the first order fragment is already strong enough to force undecidability.

Consider $FO-Strat$, the first order fragment of $MSO-Strat$ defined as follows:

$$\phi \in \Phi ::= \alpha@x \mid p@x \mid \neg\phi \mid \phi \vee \phi' \mid \langle x\rangle\phi \mid \Diamond^*\phi \mid \exists x.\phi$$

where α is a type formula, $p \in \mathcal{P}, x \in V$.

To prove this, we use the following version of tiling problem called **finite tiling problem**: Given a tiling instance $\mathcal{T} = (T, V, H, t_0, t_f)$ where T is a finite set of tiles and $H \subseteq (T \times T)$ and $V \subseteq (T \times T)$ are the horizontal and vertical constraints respectively and $t_0, t_f \in T$, does there exist n and a tiling function $f : [0, \dots n]^2 \to T$ such that $f(0,0) = t_0$ and $f(n,n) = t_f$ and for all i, j we have $[f(i,j), f(i+1,j)] \in H$ and $[f(i,j), f(i,j+1)] \in V$.

Deciding whether given tiling system has a finite tiling is *recursively enumerable* but *not recursive*, as we can reduce the halting problem of Turing machines to the finite tiling problem. An example is shown for $n = 6$ in Figure 6.1.

Overview

For the reduction to an instance of Finite Satisfiability of FO-STRAT for Large Games, fix the action set $\mathbf{\Gamma} = \{a, b, c\}$. First we need to encode the grid and the idea is to associate the grid point (i, j) with the strategy profiles that induce the distribution $(i, j, 2n - (i + j))$. So $(0, 0)$ and (n, n) grid points are associated with profiles that induce the distribution $(0, 0, 2n)$ and $(n, n, 0)$ respectively.

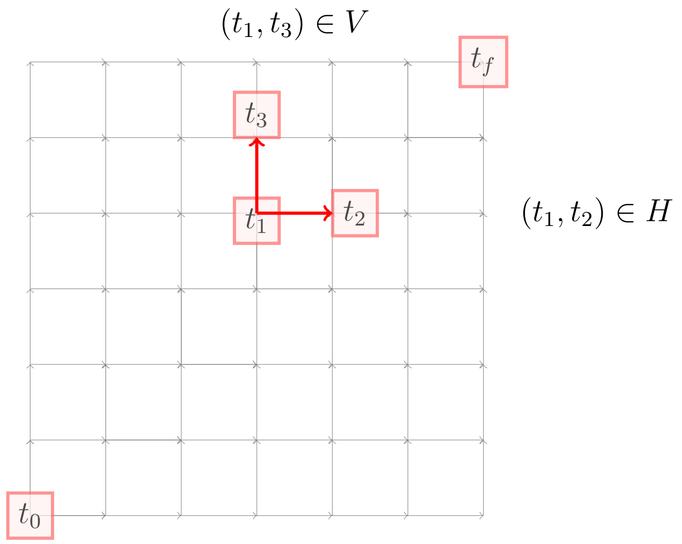

Figure 6.1: An illustration of the Finite Tiling Game

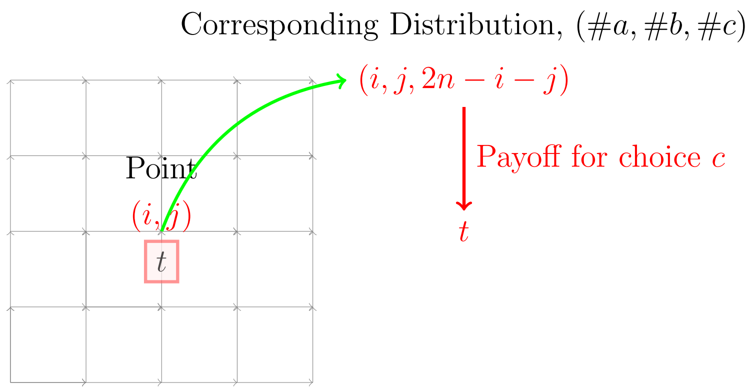

Figure 6.2: Reduction to Large Games

For the successor edges of the grid points, we need to ensure that for every profile that induces the distribution $(i, j, 2n - (i + j))$ there is an edge to two different strategies which induce the distributions $(i+1, j, 2n-(i+j+1))$ and $(i, j+1, 2n-(i+j+1))$ respectively. The tiling information is encoded as the pay-off obtained by players who choose action c. If (i, j) has tile t then the distribution $(i, j, 2n-(i+j))$ obtains payoff t for the players who choose c, as can be understood from Figure 6.2.

The horizontal edge out of (i, j) corresponds to a horizontal constraint and the vertical edge out of (i, j) corresponds to a vertical constraint. The edges will correspond to the improvement edges of the Large Game we are going to construct. It reflects that a player playing choice c in the corresponding strategy profile for (i, j)

will shift to choose either strategies a and b. If the player chooses a, a horizontal constraint in the Tiling instance get satisfied. If the player chooses b, a vertical constraint in the Tiling instance gets satisfied.

But for the distribution $(n, n, 0)$ since there are no players who pick c, we associate the payoff t_f to all the players. We also make sure t_f is maximal pay-off, so that there are no outgoing edges and hence is the end point of the grid.

Fix a tiling instance $\mathcal{T} = (T, V, H, t_0, t_f)$ where T is a finite set of tiles and $H \subseteq (T \times T)$ and $V \subseteq (T \times T)$. Fix $\mathbf{\Gamma} = \{a, b, c\}$. For every tile $t \in T$ let t be a corresponding proposition and let p, q be new (singleton) payoffs such that the ordering relation is defined where for all $t \neq t_f$ we have $q \prec t \prec p$ and $p \prec t_f$. The formulas that encode the grid are given by the conjunction of the following formulas:

$$\phi_1 := \Box^* \forall x. \Big(\big(a@x \vee b@x \big) \implies p@x \wedge \Box^* p@x \Big)$$

$$\phi_2 := \Box^* \Big(\big(\sharp a = \tfrac{1}{2} \vee \sharp b = \tfrac{1}{2} \big) \implies \forall x([x]\bot) \Big)$$

$$\phi_3 := \Box^* \Big(\big(\sharp a < \tfrac{1}{2} \wedge \sharp b < \tfrac{1}{2} \big) \implies \\ \forall x \; c@x \implies \big(\neg p@x \;\wedge\; \neg t_f@x \wedge \; \langle x \rangle (a@x) \;\wedge \langle x \rangle (b@x) \; \big) \Big)$$

The meaning of the formulas encoding player behaviour are as follows :

ϕ_1 ensures that, if σ induces the distribution $(i, j, 2n - (i + j))$ then the only permitted edges are from σ to σ' where σ' induces the distribution of the form $(i + 1, j, 2n - (i + j + 1))$ or $(i, j + 1, 2n - (i + j + 1))$.

ϕ_2 ensures we do not have any transitions from the right or top border grid points.

ϕ_3 ensures that for every profile σ if σ induces the distribution $(i, j, 2n - (i + j))$ where $i < n$ and $j < n$ then all players playing c will not get p as payoff and hence they can necessarily make a transition to σ' where they play either a

or b and get a better pay off. Therefore, all points in the interior of the grid necessarily have a transitions to the right and top successor grid points.

The tiling information at the grid point (i,j) is encoded as the pay off obtained by players who play c. In particular if (i,j) is tiled t then in the distribution $(i,j,2n-(i+j))$ will have $t \in \omega_c((i,j,2n-(i+j)))$.

For every $t \in T$ and $x \in V$ define the formula $only(t,x) := \; t@x \wedge \bigwedge_{t' \neq t} \neg t'@x$ which states that x gets payoff corresponding to tile t (and no other tile).

The tiling constraints are encoded as follows and Table 6.1 contains the meaning of the formulas :

$$\psi_0 \;:=\; \sharp a \leq 0 \wedge \sharp b \leq 0 \wedge \forall x\Big(c@x \implies only(t_0,x)\Big)$$

$$\psi_1 \;:=\; \bigvee_{t \in T}\Big(\Box^*\big(\forall z(c@z \implies only(t,z)\big)\Big)$$

$$\psi_2 \;:=\; \Box^*\Big(\big(\sharp a = \tfrac{1}{2} \wedge \sharp b = \tfrac{1}{2})\big) \vee (\alpha_H \wedge \alpha_V)\Big)$$

$$\begin{aligned}\psi_3 \;:=\; & \Diamond^*(\sharp c = 0 \wedge \sharp a = \tfrac{1}{2} \wedge \sharp b = \tfrac{1}{2}) \;\wedge \\ & \Box^*\Big(\;\big(\exists x\langle x\rangle(a@x \wedge \sharp c = 0) \implies \bigvee_{(t,t_f)\in H}(\forall z(only(t,z))\big) \wedge \\ & \quad \big(\exists x\langle x\rangle(b@x \wedge \sharp c = 0) \implies \bigvee\nolimits_{(t,t_f)\in V}(\forall z(only(t,z))\big)\Big)\end{aligned}$$

$$\psi_4 \;:=\; \Box^*\big(\sharp c = 0 \implies \forall x.t_f@x\big)$$

where,

$$\alpha_H \;:=\; \bigvee_{(t,t')\in H}\Big(\forall x\big(c@x \implies t@x\big) \wedge \forall y\big(\langle y\rangle a@y \implies \forall z(c@z \implies only(t',z))\big)\Big)$$

$$\alpha_V \;:=\; \bigvee_{(t,t')\in V}\Big(\forall x\big(c@x \implies t@x\big) \wedge \forall y\big(\langle y\rangle b@y \implies \forall z(c@z \implies only(t',z))\big)\Big)$$

We can now prove the following theorem.

Formula	Intended Meaning
ψ_0	ensures that we start at σ_0 with distribution $(0, 0, 2n)$ and all players (playing c) get t_0 as payoff
ψ_1	ensures that only one tile is obtained as payoff at any profile for players choosing c.
ψ_2	ensures that either the vertical and horizontal tiling constraints are respected or we are at a top or right border grid point.
ψ_3	ensures that the grid points $(n, n-1)$ and $(n-1, n)$ satisfy the tiling constraints to their respective top and right successors.
ψ_4	ensures that the grid point (n, n) gets the tile t_f
α_H	states that if players playing c get t as payoff in current profile and get t' in the next profile where the transition player is now choosing action a then $(t, t') \in H$.
α_V	states the vertical constraints when the transition player chooses the action b.

Table 6.1: Encoding the Grid Constraints

Theorem 6.4.2: Reduction to Finite tiling

Given a tiling instance $\mathcal{T} = (T, H, V, t_0, t_f)$, we can construct a formula ϕ_T such that $\mathcal{T}$ has a finite tiling iff ϕ_T is satisfiable.

Proof. Suppose $\mathcal{T}$ has a finite tiling. Then there is some n and a tiling function $f : [0, \dots n]^2 \to T$. Define the game over $2n$ players $M = ([1, \dots 2n], P_T, \omega)$ where $P_T = \{p, q\} \cup \{t \mid t \in T\}$ and ω is defined as follows in Table 6.2 :

It can be verified that for σ_0 which is the profile that induces the distribution $(0, 0, 2n)$ we have $M, \sigma_0 \models \phi_T$ where $\phi_T = \phi_1 \wedge \phi_2 \wedge \phi_3 \wedge \psi_0 \wedge \psi_1 \wedge \psi_2 \wedge \psi_3 \wedge \psi_4$.

For the other direction, assume that the formula is satisfiable over some game. Let $M, \sigma_0 \models \phi_T$. Then ψ_1 ensures that the number of players is even. Let $2n$ be the number of players. Define the function $f : [0 \dots n]^2 \to T$ such that for all i, j if $i \neq n$

Distribution	Condition	ω_a	ω_b	ω_c
$(i,j,2n-(i+j))$	$i,j<n$	$\{p\}$	$\{p\}$	$\{f(i,j)\}$
$(i,j,2n-(i+j))$	$i=n$ and $j<n$	$\{p\}$	$\{p\}$	$\{f(i,j)\}$
$(i,j,2n-(i+j))$	$i<n$ and $j=n$	$\{p\}$	$\{p\}$	$\{f(i,j)\}$
$(n,n,0)$		$\{t_f\}$	$\{t_f\}$	$\{\}$
$(i,j,2n-(i+j))$	$i>n$ or $j>n$	$\{q\}$	$\{q\}$	$\{q\}$

Table 6.2: Payoff Function for the $2n$ players

or $j \neq n$ then $f(i,j) = t$ such that there is a profile σ reachable from σ_0 such that σ induces the distribution $(i,j,2n-(i+j))$ and $M,\sigma \models \forall x(c@x \implies only(t,x))$. Finally define $f(n,n) = t_f$.

To see that f is well defined, it is enough to prove that for all (i,j) there is some profile σ reachable from σ_0 that induces the distribution $(i,j,2n-(j+j))$. This is proved by induction on $i+j$. Base case is when $i+j=0$ and σ_0 satisfies the condition. For the induction step, since $i<n$ and $j<n$ there is some player who plays c, and ϕ_3 ensures that there is a next transition to the profile that induces the required distribution.

Finally to see that f satisfies the tiling constraints, note that we have $f(0,0) = t_0$ and $f(n,n) = t_f$. Also, ψ_2 ensures that horizontal and vertical constraints for all the grid points are satisfied except (n,n) and ψ_3 ensures that these constraints are satisfied for the grid point $(n,n-1)$ and $(n-1,n)$ to their respective top and right successor (n,n).

$\square$

Remark

Note that the coding uses only *three* player types. The entire encoding can be done in only one free variable. Moreover, even the monodic fragment, where every modal formula has only one variable free in its scope, is undecidable.

6.5 Axiomatization

We now consider an inference system for validity in $MIQ - Strat$. The axiomatization follows that of propositional dynamic logic ([?]) with some characteristics of modal logic of implicit quantification, but the key element here is the characteristic formulas that describe choice distributions.

Firstly, some preliminaries before we get to axiom schemes.

Recall that the abbreviation $\sharp\ a = r$ denotes $\sharp\ a \leq r \wedge \sharp\ a \geq r$. Let $k = |\mathbf{\Gamma}|$ and $distr(r_1, \ldots, r_k)$ stand for $\bigwedge_{\ell=1}^{k} \sharp a_\ell \ = \ r_\ell$, where $\Sigma_\ell\ r_\ell = 1$. This is a complete distribution, independent of the number of players.

Let $X \subseteq \mathbf{\Gamma}$, and $D_X = \{\sharp a_j\ rel_j\ s_j \mid a_j \in X\}$. We say that $distr(r_1, \ldots, r_k)$ is a *completion* of D_X if for all $a_j \in X$, $(= \ r_j)$ entails $(rel_j\ s_j)$, and we say D_X is coherent if there exists a completion of D_X.

For $A \subseteq P$ and $i \in C$(the set of constants), let $\nu_A@i$ denote the formula $\bigwedge_{p \in A} p@i \wedge \bigwedge_{q \notin A} \neg q@i$.

Axiom schemes AX

- $(A0)$ Substitutional instances of tautologies of propositional logic
- $(A1)$ $[\forall](\phi \Rightarrow \psi) \Rightarrow ([\forall]\phi \Rightarrow [\forall]\psi)$
- $(A2)$ $\Diamond^*\phi \equiv (\phi \vee \langle\exists\rangle\Diamond^*\phi)$
- $(A3)$ $\bigvee_{a \in \mathbf{\Gamma}} \sharp a \ > \ 0$
- $(A4)$ $distr(r_1, \ldots, r_k) \implies (p@i \implies \Box^*(distr(r_1, \ldots, r_k) \implies p@i))$
- $(A5)$ $(a@i \wedge \nu_A@i) \implies [\forall](\neg a@i \implies \neg\nu_B@i) \quad B \preceq A$
- $(A6)$ $\neg\sharp a\ rel\ r \equiv \sharp a\ rel'\ r \quad rel' = \neg rel, \quad a \in \mathbf{\Gamma}$
- $(A7)$ $\sharp a\ rel\ r \implies \sharp a\ rel'\ r' \quad (rel\ r)$ entails $(rel'\ r') \quad a \in \mathbf{\Gamma}$

Inference rules

$$(MP)\ \frac{\phi, \quad \phi \implies \psi}{\psi} \qquad (Ind)\ \frac{\phi \implies [\forall]\phi}{\phi \implies \Box^*\phi}$$

$$(Gen)\ \frac{\phi}{[\forall]\phi}$$

$$(Num)\ \frac{\bigwedge_{i\in D} \top@i \implies \phi, \quad D \subseteq C, |D| = O(2^{|\phi|^2})}{\phi}$$

$$(Impr)\ \frac{(\phi \wedge \bigwedge_{i\in u} a_i@i) \implies [\forall](\bigvee_{j\in u} a_j@j \vee \bigvee_{j\notin u}(\neg a_j)@j) \quad u \subseteq_{fin} \mathbf{\Gamma}}{\phi \implies [\forall]\bot}$$

Soundness of the Axiom schemes in AX

Axioms (A0), (A1), (A2) and the inference rules Modus Ponens Gen and Induction are standard in any propositional modal logic of transitive closure.

Axiom (A3) is a non-triviality condition, which asserts that some choice is made by some player at any profile.

(A4) asserts that the outcomes are maps that depend only on choice distributions, and is the characteristic formula for large games.

(A5) describes improvement dynamics: when player i deviates from strategy a, it is because of some improved outcome.

(A6) and (A7) facilitate reasoning with inequalities.

Lemma 6.5.1: Soundness of $A1$

Axiom A1 is sound.

Proof. Assume $M, \sigma \models [\forall](\phi \implies \psi)$ and $M, \sigma \models [\forall](\phi)$ holds. Fix a subset of players $u \subseteq N$,

For σ' such that $\sigma \to_u \sigma'$ holds, we have, $M, \sigma' \models (\phi \implies \psi)$ and $M, \sigma' \models \phi$

therefore, $M, \sigma' \models \psi$. Therefore by definition, $M, \sigma \models [\forall]\psi$ $\square$

Lemma 6.5.2: Soundness of $A2$

Axiom A2 is sound.

Proof. From the definition of $\Diamond^*$ we have that, $\rightarrow^* = (\cup_u \rightarrow_u)^*$, the reflexive transitive closure of the union of the improvement edge relations, which means, that if $M, \sigma \models \Diamond^*\phi$, then either ϕ holds at σ, ie, $M, \sigma \models \phi$ or there are a non zero length path, $\sigma \rightarrow \sigma_1 \rightarrow \sigma_2 \ldots \rightarrow \sigma'$ (labelled appropriately by some subsets of players), to some σ' such that $M, \sigma' \models \phi$.

So we can see at σ_1 using the definition of $\Diamond^*$, $M, \sigma_1 \models \Diamond^*\phi$ holds as well. Which means, $M, \sigma \models \langle\exists\rangle\phi$ and therefore $M, \sigma \models \phi \vee \langle\exists\rangle\phi$. $\square$

Lemma 6.5.3: Soundness of $A3$

Axiom A3 is sound.

Proof. Suppose not.

Then $\bigwedge_{a \in \mathbf{\Gamma}} \sharp a \leq 0$ will hold at a strategy profile. It means that no player has chosen any of the strategies. But this violates the definition of strategy profiles, which are functions from the player set N to strategy set, $\mathbf{\Gamma}$. It means at least one strategy should be chosen by any of the players. $\square$

Lemma 6.5.4: Soundness of $A4$

Axiom A4 is sound.

Proof. Suppose for M, σ, $M, \sigma \models distr(r_1, \ldots, r_k)$ and $M, \sigma \models p@i$ for some constant i and for any σ' reachable from σ we have $M, \sigma' \models distr(r_1, \ldots, r_k)$ but we have, $M, \sigma' \models \neg p@i$.

σ and σ' have the same distributions, If for a particular player interpreted from the constant i, we have different payoffs, it would mean the payoff function ω is not a function! This is a contradiction. $\square$

Lemma 6.5.5: Soundness of $A5$

Axiom A5 is sound.

Proof. Suppose not. Therefore let, $M, \sigma \models a@i \wedge \nu_A@i$ and for a subset $u \subseteq N$ of players, we have $\sigma \rightarrow_u \sigma'$ and $M, \sigma' \models \neg a@i$ but, we have $M, \sigma' \models \nu_B@i$.

Since $M, \sigma' \models \neg a@i$, this means $i \in u$, therefore i would only deviate when there is an improvement in payoff ie, $A \prec B$. But, we are given in the axiom scheme that $B \preceq A$ which is an impossibility! $\square$

Lemma 6.5.6: Soundness of $A6$

Axiom A6 is sound.

Proof.

$$
\begin{aligned}
&\sigma \models \neg \sharp a \ rel \ r \\
&\quad \iff \not\models \sharp a \ rel \ r \\
&\quad \iff \frac{|\{j \in N \mid \sigma[j] = a\}|}{|N|} \ rel \ r \text{ does not hold} \\
&\quad \iff \frac{|\{j \in N \mid \sigma[j] = a\}|}{|N|} \ rel' \ r \\
&\sigma \models \sharp a \ rel' \ r
\end{aligned}
$$

* where $rel' = \neg rel$

$\square$

Lemma 6.5.7: Soundness of $A7$

Axiom A7 is sound.

Proof. Showing this for $rel = \leq$. Similarly for others it can be worked out.

$$\begin{aligned}
\sigma \models \sharp a \ &\leq \ r \\
\iff & \frac{|\{j \in N \mid \sigma[j] = a\}|}{|N|} \leq \ r \\
\iff & \frac{|\{j \in N \mid \sigma[j] = a\}|}{|N|} \leq \ r' \text{ where } r \leq r' \\
\sigma \models \sharp a \ &\leq \ r'
\end{aligned}$$

□

Soundness of the Inference Rules in AX

Lemma 6.5.8: Soundness of Ind

Inference Rule Ind is sound.

Proof. For all (M, σ), we have $M, \sigma \models \phi \implies [\forall]\phi$

Let us suppose the conclusion is not valid, that is there exists σ' such that $M, \sigma' \models \phi \wedge \Diamond^* \neg\phi$. By the assumption of the premise we have $M, \sigma' \models [\forall]\phi$ but we also have by our supposition that $M, \sigma' \models \Diamond^* \neg\phi$ which get's us our contradiction. □

Lemma 6.5.9: Soundness of Gen

Inference Rule Gen is sound.

Proof. For any (M, σ), we have, $M, \sigma \models \phi$.

Therefore, for any $u \subseteq N$, and σ' such that, $\sigma \rightarrow_u \sigma'$, we would have, $M, \sigma' \models \phi$. Therefore, $M, \sigma \models [\forall]\phi$ □

Lemma 6.5.10: Soundness of Num

Inference Rule Num is sound.

Proof. The rule (Num) asserts the essence of the proposition we encountered earlier.

Suppose the premise of (Num) is valid but the conclusion is not.

Then $\neg\phi$ is satisfiable. But then it is satisfiable in an n-player game, where $n \leq 2^{O(|\phi|^2)}$. In that game we interpret D as names of players, and we see that $\bigwedge_{i \in D} \top@i \wedge \neg\phi$ is satisfiable, which contradicts our assumption that the premise is valid.

□

Lemma 6.5.11: Soundness of Impr

Inference Rule Impr is sound.

Proof. The rule (Impr) asserts that whenever a set of players u have an improvement, they deviate from the current profile.

Suppose that the premise of the rule is valid but the conclusion is not.

Then the formula $\phi \wedge \langle\exists\rangle\top$ is satisfied in model M at profile σ.

Let u be a set of players such that for some σ', we have: $\sigma \rightarrow_u \sigma'$. Let $a_i = \sigma[i]$. Clearly, the formula $\phi \wedge \bigwedge_{i \in u} a_i@i$ holds at σ. Since this is an improvement edge, we know that for all $i \in u$, $\sigma'[i] \neq a_i$, and for all $j \notin u$, $\sigma'[j] = a_j$. Therefore $\bigwedge_{i \in u} \neg a_i@i \wedge \bigwedge_{j \notin u} a_j@j$ holds at σ'. Therefore, $(\phi \wedge \bigwedge_{i \in u} a_i@i) \wedge \langle\exists\rangle(\bigwedge_{i \in u} \neg a_i@i \wedge \bigwedge_{j \notin u} a_j@j)$ holds at σ, contradicting the validity of the premise. □

The (Impr) rule, when used along with axiom (A5) ensures that the new outcome for each of the players in u is preferred over the current one.

Theorem 6.5.1: Satisfiability problem of *MIQ STRAT*

Every consistent formula is satisfiable in a model with $2^{O(|\phi|^2)}$ players. Therefore the axiom system is complete, and the satisfiabililty problem for the logic can be solved in nondeterministic double exponential time.

Overview

We say $\vdash \phi$ if ϕ is a theorem of the system. We say that α is consistent if $\nvdash \alpha$. A finite set of formulas A is consistent if the conjunction of all formulas in A, denoted $\hat{A}$, is consistent. When we have a finite family R of sets of formulas, we write $\tilde{R}$ to denote the disjunction of all formulas $\hat{A}$, where $A \in R$.

For completeness, we show that every consistent formula is satisfiable. Fix a consistent formula ϕ_0. We invoke the rule (Num) to argue that for some set D of constant symbols $\phi_1 = \bigwedge_{i \in D} \top@i \wedge \phi_0$ is consistent. Let $D_1 = D \cup \{j \in C \mid j$ occurs in $\phi_0\}$. We fix ϕ_1 and D_1 for the rest of the proof.

Let δ be any distribution constraint formula. Let $SF_d(\delta)$ be the set of subformulas of δ closed under the condition: if $\neg\sharp\ a\ rel\ r \in SF_d(\delta)$ then $\sharp\ a\ rel'\ r \in SF_d(\delta)$ where $rel' = \neg rel$.

Let ϕ be any formula. We define $SF(\phi)$ to be the least set of formulas containing ϕ and closed under the following conditions:

- For all $a \in \mathbf{\Gamma}$ and $i \in D_1$, $\top@i, a@i \in SF(\phi)$.
- If $p@j \in SF(\phi)$ and $i \in D_1$, $p@i \in SF(\phi)$.
- If $\delta \in SF(\phi)$, then $SF_d(\delta) \subseteq SF(\phi)$.
- $\neg\psi \in SF(\phi)$ iff $\psi \in SF(\phi)$ (where we treat $\neg\neg\phi$ to be the same as ϕ).
- If $\psi_1 \vee \psi_2 \in SF(\phi)$ then $\{\psi_1, \psi_2\} \subseteq SF(\phi)$.

- If $\langle\exists\rangle\psi \in SF(\phi)$ then $\psi \in SF(\phi)$.
- If $\Diamond^*\psi \in SF(\phi)$ then $\{\psi, \langle\exists\rangle\Diamond^*\psi\} \subseteq SF(\phi)$.

We call $A \subseteq SF(\phi_1)$ an atom if it is a maximal consistent subset (MCS) of $SF(\phi_1)$: that is A is consistent, and for any $A \subseteq B \subseteq SF(\phi_1)$ such that B is consistent, we have $A = B$. Note that every consistent subset A of $SF(\phi_1)$ can be extended to a maximal consistent subset of $SF(\phi_1)$ by **Lindenbaum's Lemma**. Let AT denote the set of MCS's.

For $A \in AT$ and $i \in D_1$, let $P_i(A) = \{p \mid p@i \in A\}$.

For $A, B \in AT$ and for $u \subseteq D_1$, we write

$$\begin{aligned} A \prec_u B \textbf{ iff } \quad & \text{for all } i \in u,\ P_i(A) \prec P_i(B) \\ & \{a@i \in A \mid i \in u\} \cap B = \emptyset \\ & \{a@j \in A \mid j \notin u\} \subseteq B. \end{aligned}$$

Definition 6.5.1: Word Model

Define $\Rightarrow \subseteq (AT \times 2^{D_1} \times AT)$, an edge relation on AT labelled by constant symbols from D_1 by:

$$\begin{aligned} A \Rightarrow_u B \textbf{ iff } \quad & \hat{A} \wedge \langle\exists\rangle\hat{B} \text{ is consistent} \\ & A \prec_u B. \end{aligned}$$

Let G denote the graph $(AT, \Rightarrow)$.

Since ϕ_1 is consistent, there exists an MCS $A_1 \in AT$ such that $\phi_1 \in A_1$. Let G_1 be the induced subgraph of G by restricting to atoms reachable from A_1, denoted AT_1.

We have the following observations on G_1.

Truth Lemma

1. Every A in AT_1 induces a profile σ_A over D_1.

2. For every A in AT_1, there exists $distr(r_1, \ldots, r_k)$ such that $\hat{A} \wedge distr(r_1, \ldots, r_k)$ is consistent (where $k = |\mathbf{\Gamma}|$). (Note that $distr(r_1, \ldots, r_k)$ need not be in $SF(\phi_1)$.)

3. For every A in AT_1, if $\sharp a \ rel \ r \in A$, then $|\{i \in D_1 \mid a@i \in A\}| \ rel \ r \cdot |D_1|$.

4. For every A, B in AT_1, if the distributions induced by both profiles are the same, then for all $i \in D_1$ and propositions $p \in SF(\phi_1) \cap P$, $p@i \in A$ iff $p@i \in B$.

5. For every A in AT_1, $\alpha \in A$ **iff** $\neg\alpha \notin A$.

6. For every A in AT_1, $\alpha \vee \beta \in A$ **iff** $\alpha \in A$ or $\beta \in A$.

7. Whenever $A \Rightarrow_u B$, $\{\phi \mid [\forall]\phi \in A\} \subseteq B$.

8. For every A in AT_1, if $\langle\exists\rangle\phi \in A$ then there exists $B \in AT_1$ and $u \subseteq D_1$ such that $A \Rightarrow_u B$ and $\phi \in B$.

9. For every A in AT_1, if $\Diamond^*\phi \in A$, then there exists an atom B in AT_1 reachable from A such that $\phi \in B$.

Lemma 6.5.12: Truth Lemma 1

Every A in AT_1 induces a profile σ_A over D_1.

Proof. From the axiom schemes, $\hat{A} \wedge \bigvee_{a \in \mathbf{\Gamma}} \sharp a > 0$ must be valid. So it must correspond to some strategy profile.

We will try to construct σ_A from any given A.

Consider the Set of player constraint formulas in A, $Con_A = \{\sharp\ a\ rel\ r \in A \mid a \in \mathbf{\Gamma}, r \in \mathbb{Q}\}$. And, $\gamma_A = \{(a,i) \mid a@i \in A, a \in \mathbf{\Gamma}, i \in C\}$

If both Con_A and γ_A are empty, choose $a \in \mathbf{\Gamma}$ and make the default strategy profile for such a consistent set to be, $a^{|D_1|}$.

Else, From Con_A , we fix $a \in \mathbf{\Gamma}$ that belongs to Con_A.

Get the constraint for the particular strategy a such that $\sharp\ a\ rel\ r$ is entailed by all other, $\sharp\ a\ rel'\ r'$. Use the r of such a minimal entailed constraint for a. and pick $C' \subseteq D_1$, such that, $|C'|\ rel\ r \cdot |D_1|$. And iterate this process for all other strategies present in Con_A.

In this process, any incongruity would lead to an inconsistent set of player distribution formulas which would contradict our initial assumption that A was a maximal consistent set. And this constructive process makes the mapping from a consistent set to a strategy profile a well defined function.

□

Lemma 6.5.13: Truth Lemma 2

For every A in AT_1, there exists $distr(r_1, \ldots, r_k)$ such that $\hat{A} \wedge distr(r_1, \ldots, r_k)$ is consistent (where $k = |\mathbf{\Gamma}|$).

Proof. From Lemma 6.5.12, we can induce a strategy profile σ_A, and hence we can from the strategy profile extract the distribution required. Each $r_a = \{j \in D_1 \mid \sigma_A[j] = a\}$. And then we throw in the formula $\sharp a = r_a$ and take a conjunction over all these distribution constraint formulas, for each strategy choice in $\mathbf{\Gamma}$, to get the required $distr$ formula. □

Lemma 6.5.14: Truth Lemma 3

For every A in AT_1, if $\sharp a\ rel\ r \in A$, then $|\{i \in D_1 \mid a@i \in A\}|\ rel\ r \cdot |D_1|$.

Proof. Observe the construction of strategy profiles from Lemma 6.5.12, this was an invariant it maintained in its construction. □

Lemma 6.5.15: Truth Lemma 4

For every A, B in AT_1, if the distributions induced by both profiles are the same, then for all $i \in D_1$ and propositions $p \in SF(\phi_1) \cap P$, $p@i \in A$ **iff** $p@i \in B$.

Proof. If A or B is reachable from either of it's paths, then because of Axiom $A4$ we would have $distr(r_1, \ldots, r_k) \implies (p@i \implies \Box^*(distr(r_1, \ldots, r_k) \implies p@i))$ to hold true we must have $p@i \in A$ **iff** $p@i \in B$. Else, we would be able to derive the inconsistent formula $distr(r_1, \ldots, r_k) \implies (p@i \implies \Box^*(distr(r_1, \ldots, r_k) \implies \neg p@i))$ □

Lemma 6.5.16: Truth Lemma 5

For every A in AT_1, $\alpha \in A$ **iff** $\neg\alpha \notin A$.

Proof. Follows from the truth lemma for maximal consistent sets of propositional logic. □

Lemma 6.5.17: Truth Lemma 6

For every A in AT_1, $\alpha \vee \beta \in A$ **iff** $\alpha \in A$ or $\beta \in A$.

Proof. Follows from the truth lemma for maximal consistent sets of propositional logic. □

Lemma 6.5.18: Truth Lemma 7

Whenever $A \Rightarrow_u B$, $\{\phi \mid [\forall]\phi \in A\} \subseteq B$.

Proof. Suppose not.

By Lemma 6.5.16, we have $\neg\phi \in B$. By def of $\Rightarrow_u$ we have, $\hat{A} \wedge \langle\exists\rangle\hat{B}$ is consistent. But we would be having this inconsistent subformula, $[\forall]\phi \wedge \langle\exists\rangle\neg\phi$ which would violate the fact that A and B are MCS. $\square$

Lemma 6.5.19: Truth Lemma 8

For every A in AT_1, if $\langle\exists\rangle\phi \in A$ then there exists $B \in AT_1$ and $u \subseteq D_1$ such that $A \Rightarrow_u B$ and $\phi \in B$.

Proof. Let $\nu_A = \bigvee_{P_i(A) \prec Q} \nu_Q$. Moreover, if $\langle\exists\rangle\phi \in A$, then the formula $\hat{A} \wedge \langle\exists\rangle\phi$ is consistent, and by rule (Impr) and axiom (A5), for some $u \subseteq D_1$, the formula

$$(\hat{A} \wedge \bigwedge_{i \in u}(a_i@i \wedge \nu_{P_i(A)}@i)) \wedge \langle\exists\rangle(\bigwedge_{i \in u}((\neg a_i)@i \wedge \nu_A@i))$$

is consistent.

Since $a_i@i \in A$ and $\nu_{P_i(A)}@i \in A$ we can rewrite the above formula as $\hat{A} \wedge \langle\exists\rangle\hat{B}'$ where B' are the corresponding formulas stated above. By **Lindembaum's Lemma** we can convert B' to a maximal consistent set B. This B will have the following properties :

$$\begin{aligned}
&\forall i \in u : \\
&\qquad P_i(A) \prec P_i(B) \\
&\qquad \{a@i \in A \mid i \in u\} \cap B = \emptyset \\
&\qquad \{a@j \in A \mid j \notin u\} \subseteq B
\end{aligned}$$

Clearly, $A \prec_u B$. and because we also have $\hat{A} \wedge \langle\exists\rangle\hat{B}$ we have, $A \Rightarrow_u B$.

Now we need to prove, $\phi \in B$. Suppose not. Then by Lemma 6.5.16 we have $\neg\phi \in B$, which means we can have $\langle\exists\rangle\neg\phi \in A$ which would make A have an inconsistent formula violating the fact that A is a maximal consistent set.

Therefore, $A \Rightarrow_u B$ and $\phi \in B$.

$\square$

Lemma 6.5.20: Truth Lemma 9

For every A in AT_1, if $\Diamond^*\phi \in A$, then there exists an atom B in AT_1 reachable from A such that $\phi \in B$.

Proof. Thanks to axiom (A2), if $\Box^*\phi \in A$, then for every B reachable from A, $\Box^*\phi \in B$ and hence $\phi \in B$ as well. $\square$

The inference rules play a crucial role in the proofs of these statements, especially the last two. The proof of the 'fulfilment' statement for $\Diamond^*\phi \in A$ is similar to that of propositional dynamic logic ([**?**]).

We can now define the game.

Definition 6.5.2: Large Game Model from the formulas

$\mathcal{G}_{\phi_1} = (N_1, P_1, \omega_1)$ by:

$N_1 = D_1$

$P_1 = P \cap SF(\phi_1)$

$\omega_1(A)(a) = \{p \mid \{p@i, a@i\} \subseteq A\}$.

The map ω_1 is well defined by the observation above.

Let σ_1 be the profile induced by A_1, and consider the improvement graph for $\mathcal{G}_1$ given by the reachability set of σ_1. which we denote by Σ_1.

Thus we have the model M_1

Theorem 6.5.2: Reducing satisfiability to set inclusion

We now show that for all A reachable from A_1 in AT_1, and σ_A induced by A, and for all $\psi \in SF(\phi)$, $\psi \in A$ iff $M_1, \sigma_A \models \psi$.

Proof. This is proved by **induction** on structure of ψ.

Base Case .

i $\psi := \delta$, where δ is a distribution constrained formula. By the construction in Lemma 6.5.12 , it should be clear that when $\delta \in A$ then $M_1, \sigma_A \models \delta$. For the other direction, suppose $\delta \notin A$.Then we have $\neg\delta \in A$. By, Axiom $A6$, let's call this δ'. By, definition $M_1, \sigma_A \models \delta' \iff M_1, \sigma_A \not\models \delta$. Therefore if $\delta \notin A$, we have $M_1, \sigma_A \not\models \delta$.

ii $\psi := a@i$ By construction of σ_A, we would have, if $a@i \in A$ then $M_1, \sigma_A \models a@i$. If $a@i \notin A$, then by construction, $M_1, \sigma_A \not\models a@i$.

iii $\psi := p@i$ From the definition of ω_1 . If the σ_A corresponding to A has the strategy choice 'a' played at player i, then by definition, since $p@i \in A$, we would have $p \in \omega_1(A)(a)$ which would mean, $M_1, \sigma_A \models p@i$. For the other direction, $M_1, \sigma_A \models p@i$ by definition means that $p \in \omega_1(A)$ and this means, that at least $p@i \in A$ by how ω_1 has been defined.

Induction Case .

i $\psi := \neg\phi$ From Lemma 6.5.16, we know, $\phi \notin A$ which means $M_1, \sigma_A \not\models \phi$, by the Induction hypothesis. Which by definition means, $M_1, \sigma_A \models \neg\phi$. For the opposite direction, we just work through the argument in reverse.

ii $\psi := \phi_1 \vee \phi_2$ From Lemma 6.5.17, we know, $\phi_1 \in A$ or $\phi_2 \in A$ and by the induction hypothesis, it means we have $M_1, \sigma_A \models \phi_1$ or $M_1, \sigma_A \models \phi_2$, which by definition means that, $M_1, \sigma_A \models \phi_1 \vee \phi_2$. We can similarly reverse the argument to get the other direction.

iii $\psi := \langle \exists \rangle \phi$ From Lemma 6.5.19, we know that there exists $B \in AT_1$ and $u \in D_1$ such that, $A \Rightarrow_u B$ and $\phi \in B$ which by the induction hypothesis means $M_1, \sigma_B \models \phi$. which taken together, means $M_1, \sigma_A \models \langle \exists \rangle \phi$.

For the other direction, suppose $M_1, \sigma_B \models \phi$ but $\langle \exists \rangle \phi \notin A$. Since, $M_1, \sigma_B \models \phi$ holds it means by definition, we have corresponding $A \Rightarrow_u B$ for some $u \subseteq D_1$ holds as well. And, by Lemma 6.5.16, $\langle \exists \rangle \phi \notin A$ means $[\forall] \neg \phi \in A$. From Lemma 6.5.18 and, $A \Rightarrow_u B$, it means $\{\phi, \neg \phi\} \subseteq B$ which cannot be true because B is a MCS.

iv $\psi := \diamond^* \phi$ From Lemma 6.5.20, we know that there exists $B \in AT_1$ reachable from A such that, $\phi \in B$ which by the induction hypothesis means $M_1, \sigma_B \models \phi$. which taken together, means $M_1, \sigma_A \models \diamond^* \phi$.

For the opposite direction ... (also prove the truth lemma for it!)

The modal cases are all shown from the induction hypotheses using the observations above. The only tricky case is that of distribution constraint formulas, which is proved using the observation on consistency of *distr* formulas.

$\square$

We conjecture that the upper bound can be improved to deterministic double exponential time. We have a lower bound of deterministic exponential time, following the same lines as that of propositional dynamic logic.

6.5.1 The Road Map to Upper Bound

The co-induction procedure by Von-pratt.

6.5.2 The Road Map to the Lower Bound

Reduction to the PDL formulation.

6.6 Bisimulation

Bisimulation is the standard tool for analyzing models of modal logics, and propositional modal logic is the bisimulation-invariant fragment of first order logic. While the notion of bisimulation makes sense for First Order Modal Logic as well, defining it for $MSO-STRAT$ comes with complications involving the use of quantifiers over player variables. Since the bisimulation for $MIQ-STRAT$ already requires us to consider relations over player names (in addition to relating system states), we discuss only bisimulation for $MIQ-STRAT$ here. (Bismulation for $MSO-STRAT$ can be generalized from that for $MIQ-STRAT$ along the lines of the notion for Term Modal Logic ([**?**], [**?**])).

6.6.1 MIQ STRAT Bisimulation

Definition 6.6.1: Bisimulation definition

Given two large games $\mathcal{G}$ and $\mathcal{G}'$ with their corresponding models $M = (N, P, \omega, \preceq, \iota)$ and $M' = (N', P, \omega', \preceq', \iota')$ the *improvement bisimulation* is given by $R \subseteq (\Sigma \times \Sigma')$ which is a non-empty set, such that for all $(\sigma, \sigma') \in R$ and for every $i \in C$ the following holds:

1. $\sigma[\iota(i)] = \sigma'[\iota'(i)]$.

2. $\hat{\omega}_{\iota(i)}(\sigma) = \hat{\omega}'_{\iota'(i)}(\sigma')$.

3. for every $a \in \mathbf{\Gamma}$ we have $\frac{|\{j|\sigma[j]=a\}|}{|N|} = \frac{|\{k|\sigma'[k]=a\}|}{|N'|}$

4. (a) for all $k \subseteq N$ and for all $\sigma \rightarrow_k \sigma_1$ in $I_{\mathcal{G}}$, there exists some $k' \subseteq N'$ and a profile σ'_1 in $I_{\mathcal{G}'}$ such that $\sigma' \rightarrow_{k'} \sigma'_1$ in $I_{\mathcal{G}'}$ and $(\sigma_1, \sigma'_1) \in R$.

   (b) for all $k' \subseteq N'$ and for all $\sigma' \rightarrow_{k'} \sigma'_1$ in $I_{\mathcal{G}'}$, there exists some $k \subseteq N$ and a profile σ_1 in $I_{\mathcal{G}}$ such that $\sigma \rightarrow_k \sigma_1$ in $I_{\mathcal{G}}$ and $(\sigma_1, \sigma'_1) \in R$.

5. (a) whenever σ_1 is reachable from σ in $I_{\mathcal{G}}$, there exists a profile σ'_1 in $I_{\mathcal{G}'}$ that is reachable from σ' such that $(\sigma_1, \sigma'_1) \in R$

   (b) whenever σ'_1 is reachable from σ' in $I_{\mathcal{G}'}$, there exists a profile σ_1 in $I_{\mathcal{G}}$ that is reachable from σ such that $(\sigma_1, \sigma'_1) \in R$

Clearly, the improvement bisimulation relation is an equivalence relation. Note that to compare two games, the underlying payoff propositions are same for the two games. Whenever $(\sigma, \sigma') \in R$ the first two conditions ensure player interpretations corresponding to designated players preserve actions as well as payoffs. The third condition ensures that σ and σ' induce the same distribution for every action. The other conditions are standard. Note that the fourth condition ensures also that the preference relations are preserved by name mappings as required.

For any two games $\mathcal{G}$ and $\mathcal{G}'$ with their corresponding models $M = (N, P, \omega, \preceq, \iota)$ and $M' = (N', P, \omega', \preceq', \iota')$ for every $\sigma \in I_{\mathcal{G}}$ and $\sigma' \in I_{\mathcal{G}'}$ we say that σ and σ' are strategy bisimilar if there is some improvement bisimulation R over M and M' such that $(\sigma, \sigma') \in R$. Also, σ and σ' are elementarily equivalent if for every formula ϕ in the logic, we have $M, \sigma \models \phi$ iff $M', \sigma' \models \phi$.

Theorem 6.6.1: Elementary Equivalence and Bisimulation coincides

Bisimulation equivalence and elementary equivalence (of $MIQ - STRAT$) coincide for large games.

Proof. ($\Rightarrow$) Assume that σ and σ' are strategy bisimilar. Let R be the improvement bisimilar relation such that $(\sigma, \sigma') \in R$. We prove that for all $(\pi, \pi') \in R$ and for all formula ϕ we have $M, \pi \models \phi$ iff $M', \pi' \models \phi$. The proof is by induction on the structure of ϕ.

1. Case δ of type $\sharp\ a\ rel\ r$: $M, \pi \models \sharp\ a\ rel\ r$ iff $\frac{|\{j|\pi[j]=a\}|}{|N|}\ rel\ r$ iff (by condition 3) $\frac{|\{j|\pi'[j]=a\}|}{|N'|}\ rel\ r$ iff $M', \pi' \models \sharp\ a\ rel\ r$.

2. Case $a@i$: $M, \pi \models a@i$ iff $\pi[\iota(i)] = a$ iff (by condition 1) $\pi'[\iota'(i)] = a$ iff $M', \pi' \models a@i$.

3. Case $p@i$: $M, \pi \models p@i$ iff $p \in \hat{\omega}_{\iota(i)}(\pi)$ iff (by condition 2) $p \in \hat{\omega}'_{\iota'(i)}(\pi')$ iff $M, \pi' \models p@i$.

4. $\neg\phi$ and $\phi \vee \phi'$ are standard.

5. Case $\langle\exists\rangle\phi$: If $M, \pi \models \langle\exists\rangle\phi$ then there is some $k \subseteq N$ and some $\pi \rightarrow_k \pi_1$ such that $M, \pi_1 \models \phi$ and by condition $4.a$ there is some $k' \subseteq N'$ and some $\pi' \rightarrow_{k'} \pi'_1$ such that $(\pi_1, \pi'_1) \in R$. By induction $M', \pi'_1 \models \phi$ and hence $M', \pi' \models \langle\exists\rangle\phi$.

   If $M', \pi' \models \langle\exists\rangle\phi$ then it can be similarly argued suing condition $4.b$ that $M, \pi \models \langle\exists\rangle\phi$.

6. Case $\Diamond^*\phi$: if $M', \pi' \models \Diamond^*\phi$ then there is some π'_1 reachable from π such that $M', \pi'_1 \models \phi$ and by condition $5.b$ there is some π_1 reachable from π such that $(\pi_1, \pi'_1) \in R$ and hence $M, \pi_1 \models \phi$ which implies that $M, \pi \models \Diamond^*\phi$.

   Similarly if $M, \pi \models \Diamond^*\phi$ then it can be argued that $M', \pi' \models \Diamond^*\phi$.

($\Leftarrow$) Assume that σ and σ' are elementarily equivalent.

Define $R = \{(\pi, \pi') \mid \pi$ and π' are elementarily equivalent$\}$. Clearly $(\sigma, \sigma') \in R$. Now we verify that R satisfies all conditions of improvement bisimulation for every $(\pi, \pi') \in R$.

1. If first condition is violated then there is some $i \in C$ such that $\pi[\iota(i)] = a \neq \pi'[\iota'(i)]$. Then we have $M, \pi \models a@i$ and $M', \pi' \not\models a@i$ which is a contradiction.

2. If second condition is violated then there is some $i \in C$ and $p \in P$ such that $p \in \hat{\omega}_{\iota(i)}(\sigma)$ and $p \notin \hat{\omega}'_{\iota'(i)}(\sigma')$ (the other case is symmetric). Then we have $M, \pi \models p@i$ and $M', \pi' \not\models p@i$ which is a contradiction.

3. If third condition is violated then there is some $a \in \mathbf{\Gamma}$ such that $\frac{|\{j|\sigma[j]=a\}|}{|N|} = r \neq \frac{|\{k|\sigma'[k]=a\}|}{|N'|}$
   Then we have $M, \pi \models (\sharp\ a = r)$ and $M', \pi' \not\models (\sharp\ a = r)$ which is a contradiction.

4. If 4.a is violated then there is some $k \subseteq N$ and some $\pi \rightarrow_k \pi_1$ such that for all $k' \subseteq N'$ and $\pi' \rightarrow_{k'} \pi'_1$ we have $(\pi_1, \pi'_1) \notin R$. Let $S = \{\pi'_1 \mid$ for some $k' \subseteq N'$ we have $\pi' \rightarrow_{k'} \pi_1\}$. Since $I_{\mathcal{G}'}$ is finite, S is finite.

   Similarly it can be argued that violation of 4.b leads to a contradiction.

5. If 5.b is violated then there is some π'_1 reachable from π' such that for all π_1 reachable from π we have $(\pi_1, \pi'_1) \notin R$.
   Let $S = \{\pi_1 \mid \pi_1$ is reachable from $\pi\}$. Since $I_{\mathcal{G}}$ is finite, S is finite.

   Now if $S = \emptyset$ then $M, \pi \models \Box^*\bot$ and $M', \pi' \models \Diamond^*\top$ which is a contradiction. So, let $S = \{\pi_1, \ldots \pi_t\}$ and for every $l \leq t$ we have β_l such that $M, \pi_l \models \beta_l$ and $M', \pi'_1 \models \neg\beta_l$. Thus we have $M, \pi \models \Box^*(\bigvee_{l=1}^{t} \beta_l)$ and $M', \pi' \models \Diamond^*(\bigwedge_{l=1}^{t} \neg\beta_l)$ which is a contradiction.

Similarly it can be argued that violation of $5.a$ leads to a contradiction.

$\square$

Note that since improvement graphs are finite, elementary equivalence implies bisimilarity as well.

6.6.2 MSO STRAT Bisimulation

We present the bisimulation for $MSO - STRAT$ here.

Definition 6.6.2: Definition for Bisimulation for MSO-STRAT

For any two games $\mathcal{G}$ and $\mathcal{G}'$ with their corresponding models $M = (N, P, \omega, \delta)$ and $M' = (N', P, \omega', \delta')$ for every $\sigma \in I_{\mathcal{G}}$ and $\sigma' \in I_{\mathcal{G}'}$ and for all $k \geq 0$ and for all $\vec{a} = (a_1, a_2, \ldots, a_k) \subseteq N, \vec{a'} = (a'_1, a'_2, \ldots, a'_k) \subseteq N'$, and, for all $l \geq 0$, and for all $\vec{u} = (u_1, u_2, \ldots, u_l) \subseteq \mathcal{P}(N)$, and for all $\vec{u'} = (u'_1, u'_2, \ldots, u'_l) \subseteq \mathcal{P}(N')$ we say that $(M, \sigma, \vec{a}, \vec{u})$ and $(M'\sigma', \vec{a'}, \vec{u'})$ are **bisimilar** if the conditions listed out below holds.

Notation We write $(M, \sigma, \vec{a}, \vec{u}) \rightleftharpoons^{k,l} (M', \sigma', \vec{a'}, \vec{u'})$ to denote bisimilar structures

And $\leftrightharpoons \; := \bigcup_{k,l \geq 0} \leftrightharpoons^{k,l}$.

Definition 6.6.3: Type Equivalence

In the definition we need to talk about the strategy profiles as well! Given a game $\mathcal{G}$ and it's corresponding model $M = (N, P, \omega)$ and two different variable maps δ^1 and δ^2, if for all player type formula $\alpha@x$ and all strategy profiles σ the following holds

$$M, \delta^1 \models \alpha@x \iff M, \delta^2 \models \alpha@x$$

then it is said (M, δ^1) is *type equivalent* to (M, δ^2). Write it as $(M, \delta^1) \sim (M, \delta^2)$.

Definition 6.6.4: Elementary Equivalence

We say, (M, σ) and (M', σ') are elementarily equivalent if for every $MSO-Strat$ formula ϕ, having k free first order variables and l free second order variables, we have

$$M, \sigma, a_1, \ldots, a_k, u_1, \ldots, u_l \models \phi(\vec{x}, \vec{X})$$

iff

$$M', \sigma', a'_1, \ldots, a'_k, u'_1, \ldots, u'_l \models \phi(\vec{x}, \vec{X})$$

where $|\vec{x}| = k$ and $|\vec{X}| = l$.

We denote this by $(M, \sigma, \vec{a}, \vec{u}) \equiv^{k,l} (M', \sigma', \vec{a'}, \vec{u'})$

If $k, l = 0$, we will have, $(M, \sigma) \equiv (M', \sigma')$ which is the same as saying for all $MSO - Strat$ sentences ϕ we have, $M, \sigma \models \phi$ iff $M', \sigma' \models \phi$

Definition 6.6.5: Bisimulation Conditions

Given two large games $\mathcal{G}$ and $\mathcal{G}'$ with their corresponding models $M = (N, P, \omega, \delta)$ and $M' = (N', P, \omega', \delta')$, suppose we already have $(M, \sigma, \vec{a}, \vec{u}) \rightleftharpoons^{k,l} (M', \sigma', \vec{a'}, \vec{u'})$, then for every $x \in V$ and every $X \in TV$ the following holds:

1. for every $a \in \mathbf{\Gamma}$ we have $\frac{|\{j \in N | \sigma[j] = a\}|}{|N|} = \frac{|\{k \in N' | \sigma'[k] = a\}|}{|N'|}$
2. (a) If $i = \delta_1(x) \in N$ then, there exists $i' \in N'$ s.t.
   - i $i' = \delta'_1(x)$ $st.$ $(M, \sigma, i, \vec{a}, \vec{u}) \rightleftharpoons^{k+1,l} (M', \sigma', i', \vec{a'}, \vec{u'})$
   - ii $\sigma[i] = \sigma'[i']$.
   - iii $\hat{\omega}_i(\sigma) = \hat{\omega}'_{i'}(\sigma')$

   (b) If $i' = \delta'_1(x) \in N$ then, there exists $i \in N$ s.t.

i $i = \delta_1(x)$ $st.$ $(M, \sigma, i, \vec{a}, \vec{u}) \rightleftharpoons^{k+1,l} (M', \sigma', i', \vec{a'}, \vec{u'})$

ii $\sigma[i] = \sigma'[i']$.

iii $\hat{\omega}_i(\sigma) = \hat{\omega}'_{i'}(\sigma')$

3. (a) If $u = \delta_2(X) \subseteq N$ then, there exists $u' \subseteq N'$ s.t. $u' = \delta'_2(X)$ and $(M, \sigma, \vec{a}, u, \vec{u}) \rightleftharpoons^{k,l+1} (M', \sigma', \vec{a'}, u', \vec{u'})$

   (b) If $u' = \delta'_2(X) \subseteq N$ then, there exists $u \subseteq N$ s.t. $u = \delta_2(X)$ and $(M, \sigma, \vec{a}, u, \vec{u}) \rightleftharpoons^{k,l+1} (M', \sigma', \vec{a'}, u', \vec{u'})$

4. (a) If $i = \delta_1(x) \in N$ and $u = \delta_2(X) \subseteq N$ and $i \in u$, then there exists $i' = \delta'_1(x) \in N'$ and $u' = \delta'_2(X) \subseteq N'$ and $i' \in u'$ such that, $(M, \sigma, i, \vec{a}, u, \vec{u}) \rightleftharpoons^{k+1,l+1} (M', \sigma', i', \vec{a'}, u', \vec{u'})$

   (b) If $i' = \delta'_1(x) \in N'$ and $u' = \delta'_2(X) \subseteq N'$ and $i' \in u'$, then there exists $i = \delta_1(x) \in N$ and $u = \delta_2(X) \subseteq N$ and $i \in u$ such that, $(M, \sigma, i, \vec{a}, u, \vec{u}) \rightleftharpoons^{k+1,l+1} (M', \sigma', i', \vec{a'}, u', \vec{u'})$

5. (a) for all $n \subseteq N$ and for all $\sigma \rightarrow_n \sigma_1$ in $I_{\mathcal{G}}$, there exists some $n' \subseteq N'$ and a profile σ'_1 in $I_{\mathcal{G}'}$ such that $\sigma' \rightarrow_{n'} \sigma'_1$ in $I_{\mathcal{G}'}$ and $(M, \sigma_1, \vec{a}, \vec{u}) \rightleftharpoons^{k,l} (M', \sigma'_1, \vec{a'}, \vec{u'})$.

   (b) for all $n' \subseteq N'$ and for all $\sigma' \rightarrow_{n'} \sigma'_1$ in $I_{\mathcal{G}'}$, there exists some $n \subseteq N$ and a profile σ_1 in $I_{\mathcal{G}}$ such that $\sigma \rightarrow_n \sigma_1$ in $I_{\mathcal{G}}$ and $(M, \sigma_1, \vec{a}, \vec{u}) \rightleftharpoons^{k,l}$ $(M', \sigma'_1, \vec{a'}, \vec{u'})$.

6. (a) whenever σ_1 is reachable from σ in $I_{\mathcal{G}}$, there exists a profile σ'_1 in $I_{\mathcal{G}'}$ that is reachable from σ' such that $(M, \sigma_1, \vec{a}, \vec{u}) \rightleftharpoons^{k,l} (M', \sigma'_1, \vec{a'}, \vec{u'})$.

   (b) whenever σ'_1 is reachable from σ' in $I_{\mathcal{G}'}$, there exists a profile σ_1 in $I_{\mathcal{G}}$ that is reachable from σ such that $(M, \sigma_1, \vec{a}, \vec{u}) \rightleftharpoons^{k,l} (M', \sigma'_1, \vec{a'}, \vec{u'})$.

Clearly, the improvement bisimulation relation is an equivalence relation. Note that to compare two games, the underlying payoff propositions are same for the two games.

Lemma 6.6.1: Relation with Type equivalences

Type equivalent implies (coincides?) Bisimulation equivalence.

Proof. yet to fill in

1. Case $\sharp\ a\ rel\ r$: $M, \pi \models \sharp\ a\ rel\ r$ iff $\frac{|\{j|\pi[j]=a\}|}{|N|}\ rel\ r$ iff (by condition 1) $\frac{|\{j|\pi'[j]=a\}|}{|N'|}\ rel\ r$ iff $M', \pi' \models \sharp\ a\ rel\ r$.

$\square$

Theorem 6.6.2: MSO STRAT Bisimulation result

Bisimulation equivalence and elementary equivalence coincide for large games.

Proof. **I.H.** *for all integer k, for all integer l, for all k-elements of M, $(i_1, i_2, \ldots, i_k)$, for all l-subsets of M, $(u_1, u_2, \ldots, u_l)$, for all k-elements of M', $(i'_1, i'_2, \ldots, i'_k)$, for all l-subsets of M', $(u'_1, u'_2, \ldots, u'_l)$, for all strategy profiles π, for all strategy profiles π', for all MSO-STRAT formula $\phi(\vec{x}, \vec{X})$ (where $|\vec{x}| = k$ and $|\vec{X}| = l$)*

$$(M, \pi, \vec{i}, \vec{u}) \rightleftharpoons^{k,l} (M', \pi', \vec{i'}, \vec{u'})$$

$$\Longleftrightarrow$$

$$(M, \pi, \vec{i}, \vec{u}) \models \phi(\vec{x}, \vec{X}) \iff (M', \pi', \vec{i'}, \vec{u'}) \models \phi(\vec{x}, \vec{X})$$

($\Rightarrow$) We induct on the structure of the MSO-STRAT formula ϕ and use a weakening of the **I.H.** -

$$(M, \pi, \vec{i}, \vec{u}) \rightleftharpoons (M', \pi', \vec{i'}, \vec{u'})$$

$$\Longrightarrow$$

$$(M, \pi, \vec{i}, \vec{u}) \models \phi(\vec{x}, \vec{X}) \iff (M', \pi', \vec{i'}, \vec{u'}) \models \phi(\vec{x}, \vec{X})$$

1. Case $\sharp\ a\ rel\ r$:

$$\begin{aligned} M, \pi \models \sharp\ a\ rel\ r &\iff \frac{|\{j \mid \pi[j] = a\}|}{|N|}\ rel\ r \\ &\iff \frac{|\{j \mid \pi'[j] = a\}|}{|N'|}\ rel\ r \qquad \text{Condition 1} \\ &\iff M', \pi' \models \sharp\ a\ rel\ r \end{aligned}$$

2. Case $a@x$:

$$\begin{aligned} M, \pi, \delta \models a@x &\iff \pi[\delta_1(x)] = a \\ &\quad \text{Pick } i' \in N'\ s.t.\ i' = \delta'_1(x)\ s.t. \qquad \text{Condition } 2a.i. \\ &\quad (M, \pi, i) \leftrightharpoons (M', \pi', i') \\ &\quad \Longrightarrow \pi'[i'] = a \qquad \text{Condition } 2a.ii \\ &\quad \iff M', \pi', \delta' \models a@x. \end{aligned}$$

Similarly, if $M', \pi', \delta' \models a@x$ then using condition $2b.ii$ it can be argued that $M, \pi, \delta \models a@x$.

3. Case $p@x$:

$$
\begin{aligned}
M,\pi,\delta \models p@x \iff\ & p \in \hat{\omega}_{\delta_1(x)}(\pi) \\
& \text{Pick } i' \in N' \text{ s.t. } i' = \delta'_1(x) \qquad \text{Condition } 2a.i. \\
& (M,\pi,i) \leftrightharpoons (M',\pi',i') \\
& \implies p \in \hat{\omega}'_{i'}(\pi') \qquad \text{Condition } 2a.iii \\
& \iff M',\pi',\delta' \models p@x.
\end{aligned}
$$

Similarly for $M',\pi',\delta' \models p@x$ using condition $2b.iii$ we can argue for $M,\pi,\delta \models p@x$.

4. Case $\neg\phi$:

$$
\begin{aligned}
M,\pi,\delta \models \neg\phi &\iff M,\pi,\delta \not\models \phi \\
&\iff M',\pi',\delta' \not\models \phi \qquad \text{Equivalent restatement of the } \textbf{I.H.} \text{ in the negative} \\
&\iff M',\pi',\delta' \models \neg\phi
\end{aligned}
$$

5. Case $\phi \vee \phi'$:

$$
\begin{aligned}
M,\pi,\delta \models \phi \vee \phi' &\iff M,\pi,\delta \models \phi \text{ or } M,\pi,\delta \models \phi \\
&\iff M',\pi',\delta' \models \phi \text{ or } M',\pi',\delta' \models \phi' \qquad \textbf{I.H} \\
&\iff M',\pi',\delta' \models \phi \vee \phi'
\end{aligned}
$$

6. Case $x \in X$:

$$
\begin{aligned}
M,\pi,\delta \models x \in X &\iff i = \delta_1(x) \in u = \delta_2(X) \\
&\iff M,\pi,i,u \models x \in X \\
&\text{Pick } i' \in N' \text{ and } u' \subseteq N' \text{ by Condition 4a we have} \\
&M,\pi,i,u \rightleftharpoons M',\pi',i',u' \\
&\implies M',\pi',\delta' \models x \in X
\end{aligned}
$$

Similarly, for $M,\pi',\delta' \models x \in X$ using condition $4b$ we can deduce, $M,\pi,\delta \models x \in X$.

7. Case $\langle X\rangle\phi$:

$$
\begin{aligned}
M,\pi \models \langle X\rangle\phi &\iff \exists k \subseteq N, \pi \to_k \pi_1 \; st. \; M,\pi_1 \models \phi \\
&\implies \exists k' \subseteq N', \pi' \to_{k'} \pi_1' \; st. \\
&(M,\pi_1) \rightleftharpoons (M',\pi_1') && \text{Condition } 5.a \\
&\implies M',\pi_1' \models \phi && \textbf{I.H.} \\
&\iff M',\pi' \models \langle X\rangle\phi.
\end{aligned}
$$

If $M',\pi' \models \langle X\rangle\phi$ then it can be similarly argued using condition $5.b$ that $M,\pi \models \langle X\rangle\phi$.

8. Case $\Diamond^*\phi$:

$$M', \pi' \models \Diamond^*\phi \iff \exists \pi'_1, \pi' \to^* \pi'_1,\ st.\ M', \pi'_1 \models \phi$$

$$\implies \exists \pi_1, \pi \to^* \pi_1\ st.$$

$$(M, \pi_1) \rightleftharpoons (M', \pi'_1) \qquad \text{condition } 6.b$$

$$\implies M, \pi_1 \models \phi \qquad \textbf{I.H.}$$

$$\iff M, \pi \models \Diamond^*\phi$$

Similarly if $M, \pi \models \Diamond^*\phi$ then it can be argued using condition $6.a$ that $M', \pi' \models \Diamond^*\phi$.

9. Case $\exists x \phi$:

$$M, \pi, \vec{a}, \vec{u} \models \exists x.\phi(x, \vec{x}, \vec{X}) \iff M, \pi, i, \vec{a}, \vec{u} \models \phi(x, \vec{x}, \vec{X})$$

Pick, $\vec{a'}, \vec{u'}$ and i' exists by Condition $2.a.i$ so that

$$M, \pi, i, \vec{a}, \vec{u} \leftrightharpoons M', \pi', i', \vec{a'}, \vec{u'}$$

$$\implies M', \pi', i', \vec{a'}, \vec{u'} \models \phi(x, \vec{x}, \vec{X}) \qquad \textbf{I.H}$$

$$\iff M', \pi', \vec{a'}, \vec{u'} \models \exists x.\phi(x, \vec{x}, \vec{X})$$

A similar argument for the reverse direction will be able to show, if $M', \pi', \vec{a'}, \vec{u'} \models \exists x.\phi(x, \vec{x}, \vec{X})$ then, $M, \pi, \vec{a}, \vec{u} \models \exists x.\phi(x, \vec{x}, \vec{X})$

10. Case $\exists X\phi$:

$$M, \pi, \vec{a}, \vec{u} \models \exists X.\phi(\vec{x}, X, \vec{X}) \iff M, \pi, \vec{a}, u, \vec{u} \models \phi(\vec{x}, X, \vec{X})$$

Pick $\vec{a'}, \vec{u'}, u' \subseteq N'$ which exists by condition $3a$ s.t.

$$M, \pi, \vec{a}, u, \vec{u} \leftrightharpoons M', \pi', \vec{a'}, u', \vec{u'}$$

$$\implies M', \pi', \vec{a'}, u', \vec{u'} \models \phi(\vec{x}, X, \vec{X}) \qquad \textbf{I.H}$$

$$\iff M', \pi', \vec{a'}, \vec{u'} \models \exists X.\phi(\vec{x}, X, \vec{X})$$

($\Leftarrow$) This is yet to be filled in properly. Assume that $(M, \sigma, \vec{a}, \vec{u}) \equiv^{k,l} (M', \sigma', \vec{a'}, \vec{u'})$

Define $\leftrightharpoons\ := \{(M, \pi, \vec{a}, \vec{u}), (M', \pi', \vec{a'}, \vec{u'})) \mid s.t.\ \exists k, l\ (M, \pi, \vec{a}, \vec{u}) \equiv^{k,l} (M', \pi', \vec{a'}, \vec{u'})\}$.

Clearly $((M, \sigma), (M', \sigma')) \in \leftrightharpoons$.

Now we verify that $\leftrightharpoons$ satisfies all conditions of bisimulation for every element in $\leftrightharpoons$.

1. If condition 1. is violated then there is some $a \in \mathbf{\Gamma}$ such that $\frac{|\{j|\pi[j]=a\}|}{|N|} = r \neq \frac{|\{k|\pi'[k]=a\}|}{|N'|}$
   Then we have $M, \pi \models (\sharp\ a = r)$ and $M', \pi' \not\models (\sharp\ a = r)$ which is a contradiction.

2. If condition 2a. is violated then there is some $i \in N$ and either :

Case i. there is no $i' \in N'$ s.t. $i' = \delta_1'(x)$, then there is no, $(M, \pi, i, \vec{a}, \vec{u}) \leftrightharpoons^{k+1,l} (M', \pi', i, \vec{a'}, \vec{u'})$ therefore no, $(M, \pi, i, \vec{a}, \vec{u}) \equiv^{k+1,l} (M', \pi', i, \vec{a'}, \vec{u'})$ because we proved the ($\Rightarrow$) direction. Then there exists a MSO-STRAT $k+1, l$ - formula, $\phi(x, \vec{x}, \vec{X})$ s.t. we can convert it to a k, l- formula $\exists x.\phi(x, \vec{x}, \vec{X})$. s.t. $M, \pi \models \exists x.\phi(x, \vec{x}, \vec{X})$ and $M', \pi' \not\models \exists x.\phi(x, \vec{x}, \vec{X})$, which is a contradiction to our original assumption, hence condition 2a.i. cannot be violated.

Case ii. such that no $i' \in N'$ exists, such that $\pi[i] = a = \pi'[i']$. Then we have $M, \pi \models a@x$ and $M', \pi' \not\models a@x$ which is a contradiction.

Case iii. such that no $i' \in N'$ exists, such that, $p \in \hat{\omega}_i(\pi)$ and $p \in \hat{\omega}'_{i'}(\pi')$ Then we have $M, \pi \models p@x$ and $M', \pi' \not\models p@x$ which is a contradiction.

The arguments for condition 2b. are symmetric.

3. If condition 3a. is violated then there is some $u \subseteq N$, and, $u = \delta_2(X)$ but there exists no, $u' \subseteq N'$ such that $u' = \delta'_2(X)$. If this happens, it means there exists $(M, \pi, i, \vec{a}, \vec{u}) \not\equiv^{k,l+1} (M', \pi', i, \vec{a'}, \vec{u'})$ which would imply by ($\Rightarrow$) that there exists a MSO-STRAT $k, l+1$ formula $\phi(\vec{x}, X, \vec{X})$ which made $\equiv^{k,l+1}$ not hold true in the two models. Now we can convert the formula into a k, l MSO-STRAT formula, $\exists X.\phi(\vec{x}, X, \vec{X})$ such that $M, \pi \models \exists X.\phi(\vec{x}, X, \vec{X})$ but $M', \pi' \not\models \exists X.\phi(\vec{x}, X, \vec{X})$ which gives us a contradiction.

   The argument for 3b. is symmetric.

4. If condition 4a. is violated then, there exists $i \in N$ and $u \subseteq N$ such that $i \in u$ but,

Case i. Either there exists no $i' \in N'$ s.t. for all $u' \subseteq N'$, $i' \in u'$,

Case ii. there exists $i' \in N'$ but no, $u' \subseteq N'$ such that , $i' \in u'$

Case iii. there doesn't exist both i' and u' satisfying the above properties

   then $(M, \pi, i, \vec{a}, u, \vec{u}) \not\equiv^{k+1,l+1} (M', \pi', i, \vec{a'}, u', \vec{u'})$ then by ($\Rightarrow$) we have a $k+1, l+1$ MSO-STRAT formula that violates $\equiv^{k+1,l+1}$, the formula would have the form, $x \in X \wedge \phi(x, \vec{x}, X, \vec{X})$. We can convert this to, $\exists x, X.\ (x \in X) \wedge \phi(x, \vec{x}, X, \vec{X})$ such that, $M, \pi \models \exists x, X.\ (x \in X) \wedge \phi(x, \vec{x}, X, \vec{X})$ but $(M', \pi') \not\models \exists x, X.\ (x \in X) \wedge \phi(x, \vec{x}, X, \vec{X})$ which leads us to a contradiction from our starting assumption.

   The argument for condition 4b. is symmetric.

5. If $5.a$ is violated then there is some $n \subseteq N$ and some $\pi \rightarrow_n \pi_1$ such that for all $n' \subseteq N'$ and $\pi' \rightarrow_{n'} \pi'_1$ we have $(M, \pi_1, \vec{a}, \vec{u}) \not\equiv^{k,l} (M', \pi'_1, \vec{a'}, \vec{u'})$.

   Let $S = \{\pi'_1 \mid \text{for some } n' \subseteq N',\ \pi' \rightarrow_{n'} \pi_1\}$. Since $I_{\mathcal{G}'}$ is finite, S is finite.

   Now, if $S = \emptyset$, then $M, \pi \models \langle\exists\rangle\top$ but, $M', \pi' \models \neg\langle\exists\rangle\top$ which is a contradiction.

   Else, let $S = \{\pi'_1, \ldots \pi'_s\}$ and for every $l \leq s$ we have k, l MSO-STRAT formulas α_l such that $M, \pi_1 \models \alpha_l$ and $M', \pi'_l \models \neg\alpha_l$. Thus we have $M, \pi \models \langle\exists\rangle\big(\bigwedge_{l=1}^{s} \alpha_l\big)$ and $M', \pi' \models \neg\langle\exists\rangle\big(\bigwedge_{l=1}^{s} \alpha_l\big)$ which is a contradiction.

   Similarly it can be argued that violation of $5.b$ leads to a contradiction.

6. If $6.b$ is violated then there is some π'_1 reachable from π' such that for all π_1 reachable from π we have $(M, \pi_1, \vec{a}, \vec{u}) \not\equiv^{k,l} (M', \pi'_1, \vec{a'}, \vec{u'})$.

   Let $S = \{\pi_1 \mid \pi_1 \text{ is reachable from } \pi\}$. Since $I_{\mathcal{G}}$ is finite, S is finite.

   Now if $S = \emptyset$ then $M, \pi \models \Box^*\bot$ and $M', \pi' \models \Diamond^*\top$ which is a contradiction. So, let $S = \{\pi_1, \ldots \pi_t\}$ and for every $l \leq t$ we have k, l MSO-STRAT formulas β_l such that $M, \pi_l \models \beta_l$ and $M', \pi'_1 \models \neg\beta_l$. Thus we have $M, \pi \models \Box^*\big(\bigvee_{l=1}^{t} \beta_l\big)$ and $M', \pi' \models \Diamond^*\big(\bigwedge_{l=1}^{t} \neg\beta_l\big)$ which is a contradiction.

   Similarly it can be argued that violation of $6.a$ leads to a contradiction.

$\square$

6.7 Discussion

We have considered reasoning about strategization in large games, where a player responds not to what other specific players choose, but to what fraction of the population choose a specific strategy. Since this involves reasoning about games with unboundedly many players, we presented logics with no player identities but player variables or names for player types. A monadic second order logic with player variables and team variables is natural for such reasoning, but such a logic becomes undecidable and non-axiomatisable, so we consider a propositional modal logic of implicit quantification which we show to be decidable and for which we present a complete axiom system. The use of implicit quantification in complex distribution constraints characterise the modal logic. We also present a bisimulation characterisation for elementary equivalence.

Note that the (Num) rule and the (Impr) rule in the presented system are infinitary (but recursive). We conjecture that there is indeed no finite axiomatisation of even this propositional modal logic.

The *presentation* of large games poses many challenges, since specifying total functions on distributions can be unwieldy. The use of logic to specify player types and thereby computing the distributions for which we need outcomes to be specified opens up new possibilities in the algorithmic analysis of large games. Further, player equivalence on games (based on types) offers an interesting possibility of algebraic analysis.

While we have studied improvement dynamics for teams of players acting individually, it is of great interest to study *coalitions* and coordination. The use of implicit quantification is natural in such contexts.

The logic we have presented here is preliminary and needs more sophistication to be useful. For one thing, players do not respond to actual distributions but to *expectations* on distributions. Other important questions relate to modal characterization of subclasses of large games such as majority games and minority games, or type matching and type mismatching games.

Want to go back to the Table of contents?

Chapter 7

Conclusion

7.1 Thoughts on Improvement Graph

7.2 Thoughts on investigation using MLFP

7.3 Thoughts on exploration using Modal Logics