

ÉCOLE NORMALE SUPÉRIEURE DE LYON
Laboratoire de l'Informatique du Parallélisme

THÈSE

présentée et soutenue publiquement le 16 janvier 2008 par

Stéphane LE ROUX

pour l'obtention du grade de

Docteur de l'École Normale Supérieure de Lyon
spécialité : informatique

au titre de l'École doctorale de mathématiques et d'informatique fondamentale de Lyon

**Generalisation and Formalisation
in Game Theory**

Directeurs de thèse : Pierre LESCANNE

Après avis de : Franck DELAPLACE
Jean-François MONIN

Devant la commission d'examen formée de :

Pierre CASTÉLAN	Membre
Jean-Paul DELAHAYE	Membre
Franck DELAPLACE	Membre/Rapporteur
Pierre LESCANNE	Membre
Jean-François MONIN	Membre/Rapporteur
Sylvain SORIN	Membre

Contents

1	Introduction	11
1.1	Extended Abstract	11
1.2	Proof Theory	14
1.3	Game Theory	16
1.3.1	General Game Theory	17
1.3.2	Strategic Game and Nash Equilibrium	17
1.3.3	Nash's Theorem	18
1.3.4	Sequential Game and Nash Equilibrium	18
1.3.5	Kuhn's Theorem	19
1.3.6	Ordering Payoffs	19
1.3.7	Graphs and Games, Sequential and Simultaneous	20
1.4	Contributions	21
1.4.1	Will Nash Equilibria Be Nash Equilibria?	21
1.4.2	Abstracting over Payoff Functions	22
1.4.3	Abstracting over Game Structure	22
1.4.4	Acyclic Preferences, Nash and Perfect Equilibria: a Formal and Constructive Equivalence	23
1.4.5	Acyclic Preferences and Nash Equilibrium Existence: Another Proof of the Equivalence	24
1.4.6	Sequential Graph Games	24
1.4.7	Abstract Compromising Equilibria	26
1.4.8	Discrete Non-Determinism and Nash Equilibria for Strategy-Based Games	27
1.5	Convention	29
1.6	Reading Dependencies of Chapters	29
1.7	Graphical Summary	31
2	Abstract Nash Equilibria	33
2.1	Introduction	33
2.1.1	Contribution	33
2.1.2	Contents	34
2.2	Strategic Game and Nash Equilibrium	34
2.2.1	Strategic Game	34
2.2.2	Nash Equilibrium	35
2.2.3	Strict Nash Equilibrium	37
2.3	Convertibility Preference Game	37
2.4	Best Response Game	39

3	Topological Sorting	43
3.1	Introduction	43
3.1.1	Decidability and Computability	43
3.1.2	Transitive Closure, Linear Extension, and Topological Sorting	44
3.1.3	Contribution	44
3.1.4	Contents	45
3.1.5	Convention	45
3.2	Preliminaries	46
3.2.1	Types and Relations	46
3.2.2	Excluded Middle and Decidability	46
3.3	On Lists	47
3.3.1	Lists in the Coq Standard Library	47
3.3.2	Decomposition of a List	48
3.3.3	Repeat-Free Lists	49
3.4	On Relations	49
3.4.1	Transitive Closure in the Coq Standard Library	49
3.4.2	Irreflexivity	50
3.4.3	Restrictions	50
3.5	On Paths and Transitive Closure	51
3.5.1	Paths	51
3.5.2	Bounded Paths	52
3.5.3	Restriction, Decidability, and Transitive Closure	52
3.5.4	Intermediate Results	53
3.6	Linear Extension and Topological Sorting	53
3.6.1	Total	54
3.6.2	Try Add Arc	54
3.6.3	Try Add Arc (One to Many)	55
3.6.4	Try Add Arc (Many to Many)	55
3.6.5	Linear Extension/Topological Sort Function	56
3.6.6	Linear Extension	57
3.6.7	Topological Sorting	58
3.7	Conclusion	60
4	Abstract Sequential Games	63
4.1	Introduction	64
4.1.1	Contribution	64
4.1.2	Contents	65
4.2	Traditional Sequential Game Theory	65
4.3	Why Total Order?	69
4.3.1	Selfishness Refinements	69
4.3.2	Lack of Information	70
4.4	Preliminaries	71
4.4.1	Extension of Predicates to Lists	71
4.4.2	Extension of Functions to Lists	72
4.4.3	Extension of Binary Relations to Lists	72
4.4.4	No Successor	74
4.5	Sequential Games	75
4.5.1	Definition of Sequential Games	75
4.5.2	Induction Principle for Sequential Games	77

4.5.3	Outcomes Used in a Sequential Game	79
4.6	Strategy Profiles	81
4.6.1	Definition of Strategy Profiles	81
4.6.2	Induction Principle for Strategy Profiles	83
4.6.3	Underlying Game of a Strategy Profile	84
4.6.4	Induced Outcome of a Strategy Profile	86
4.7	Convertibility	87
4.7.1	Definition of Convertibility	87
4.7.2	Induction Principle for Convertibility	90
4.8	Concepts of Equilibrium	91
4.9	Existence of Equilibria	92
4.9.1	“Backward Induction”	92
4.9.2	The Total Order Case	94
4.9.3	Limitation	95
4.9.4	General Case	96
4.9.5	Examples	98
4.10	Conclusion	99
5	Abstract Sequential Games Again	101
5.1	Introduction	101
5.1.1	Contribution	101
5.1.2	Contents	102
5.2	Preliminaries	102
5.3	The Proof	104
5.4	Conclusion	109
6	Graphs and Path Equilibria	111
6.1	Introduction	111
6.1.1	Contribution	111
6.1.2	Contents	113
6.1.3	Conventions	114
6.2	Dalographs and Equilibria	114
6.2.1	Dalographs	114
6.2.2	Walks and Paths	115
6.2.3	Equilibria	118
6.3	Binary Relations	120
6.3.1	General Binary Relations	120
6.3.2	Binary Relations over Sequences	123
6.4	Equilibrium Existence	125
6.4.1	The Proof	125
6.4.2	Examples	130
6.4.3	Application to Network Routing	135
6.5	Simple closures	135
6.6	Preservation of Equilibrium Existence	139
6.7	Sufficient Condition and Necessary Condition	143
6.7.1	Synthesis	144
6.7.2	Example	146
6.7.3	Application to Network Routing	147
6.8	Conclusion	147

7	Sequential Graph Games	149
7.1	Introduction	149
7.1.1	Graphs and Games	150
7.1.2	Contribution	151
7.1.3	Contents	152
7.2	Sequential Graph Games and Equilibria	152
7.2.1	Sequential Graph Games	152
7.2.2	Plays	154
7.2.3	Equilibria	157
7.3	Sequential Game Equilibrium	162
7.3.1	Sequential Tree Games are Sequential Graph Games . . .	162
7.3.2	Subgame Perfect Equilibrium and Nash Equilibrium Coincide	164
7.4	Global Equilibrium Existence	168
7.4.1	Preliminary Results	168
7.4.2	Sufficient and Necessary Conditions	172
7.4.3	Further Equilibrium (Non-) Existence	175
7.5	Conclusion	176
8	Abstract Compromised Equilibria	179
8.1	Introduction	179
8.1.1	Nash's Theorem	179
8.1.2	Contribution	179
8.1.3	Contents	180
8.2	Probabilistic Nash Equilibrium	180
8.3	Continuous Abstract Nash Equilibria	182
8.3.1	Continuous BR Nash Equilibrium	183
8.3.2	Continuous CP Nash Equilibrium	183
8.4	Discrete and Dynamic Equilibria	184
8.4.1	Strongly Connected Component	184
8.4.2	Discrete and Dynamic Equilibrium for CP Game	187
8.4.3	Discrete and Dynamic Strict Equilibrium for BR Game . .	188
8.4.4	Examples and Comparisons	189
8.5	Conclusion	191
9	Discrete Non Determinism	193
9.1	Introduction	193
9.1.1	Contribution	194
9.1.2	Contents	195
9.2	Abstract Strategic Games	195
9.3	From Continuous to Discrete Non Determinism	197
9.4	A Simple Pre-Fixed Point Result	198
9.5	Ndbr Multi Strategic Games	200
9.6	Equilibrium for Abstract Strategic Games	202
9.6.1	Non Deterministic Equilibrium Existence	203
9.6.2	Example	205
9.6.3	Comparisons	207
9.7	Equilibrium for Multi Strategic Games	209
9.8	Conclusion	214

<i>CONTENTS</i>	7
10 Conclusion	215
10.1 Summary and Additional Remarks	215
10.2 Further Research	219

Résumé

Les jeux stratégiques sont une classe de jeux fondamentale en théorie des jeux : la notion d'équilibre de Nash fut d'abord définie pour cette classe. On peut se demander si tout jeu stratégique a un équilibre de Nash, mais ce n'est pas le cas. En outre, il semble qu'il n'existe pas de caractérisation simple des jeux stratégiques qui ont un équilibre de Nash. Dans la littérature, on peut trouver au moins deux méthodes qui pallient ce problème : selon la première méthode, on définit les jeux séquentiels qui, à plongement près, sont une sous-classe des jeux stratégiques ; ensuite on définit les équilibres parfaits en sous-jeux qui sont, pour les jeux séquentiels, un raffinement de la notion d'équilibre de Nash ; enfin on prouve que tout jeu séquentiel a un équilibre parfait en sous-jeux. C'est ainsi que Kuhn montra que tout jeu séquentiel a un équilibre de Nash. Selon la deuxième méthode, on affaiblit la notion d'équilibre de Nash, par exemple en utilisant les probabilités. C'est ainsi que Nash montra que tout jeu stratégique fini a un équilibre de Nash probabiliste.

Les travaux susnommés furent effectués pour des jeux impliquant seulement des gains qui sont des nombres réels; ceci après justification de cette restriction par von Neumann et Morgenstern qui mentionnèrent également, sans en poursuivre l'étude, une notion de gain abstrait. En raison de son succès, la restriction aux nombres réels devint rapidement un principe pour la plupart des théoriciens de jeux. Malheureusement, ce glissement d'une restriction consciente vers un dogme peut interdire l'émergence d'approches alternatives. Cependant, certaines de ces approches pourraient être non seulement intéressantes pour elles-mêmes, mais aussi aider à mieux appréhender l'approche traditionnelle.

Cette thèse propose la suppression du dogme "tout est nombre réel" et l'étude d'approches alternatives. Elle introduit des formalismes abstraits qui généralisent les notions de jeu stratégique et d'équilibre de Nash. Bien entendu, certains jeux abstraits n'ont pas d'équilibre de Nash abstrait. Pour pallier ce problème d'existence, cette thèse exploite successivement les techniques de Kuhn et de Nash. Selon Kuhn et ses précurseurs (*e.g.* Zermelo), cette thèse introduit la notion de jeu séquentiel abstrait et généralise le résultat de Kuhn de manière substantielle, tout ceci étant intégralement formalisé dans l'assistant de preuve Coq. Ensuite on généralise les jeux séquentiels abstraits au moyen des graphes et on obtient des résultats encore plus généraux. Selon Nash et ses précurseurs (*e.g.* Borel), cette thèse considère des manières d'affaiblir la notion d'équilibre de Nash afin de garantir l'existence d'équilibre pour tout jeu. Cependant, l'approche probabiliste n'est plus pertinente dans les jeux abstraits. Alors, en fonction de la classe de jeux abstraits considérée, on résout le problème soit grâce à une notion de non-déterminisme discret, soit grâce à une notion de puits pour composante fortement connexe dans un graphe orienté.

Abstract

Strategic games are a fundamental class of games in game theory: the notion of Nash equilibrium was first defined for this class. One may wonder whether or not every strategic game has a Nash equilibrium, but this is not the case. Furthermore, there seems to be no simple characterisation of the strategic games that actually have a Nash equilibrium. At least two ways to cope with this issue can be found in the literature: First, one defines a subclass (up to embedding) of strategic games, namely sequential games; then one defines the stronger notion of subgame perfect equilibrium as a refinement of Nash equilibrium for this subclass; finally, one proves that every sequential game has a subgame perfect equilibrium. That is how Kuhn proved that every sequential game has a Nash equilibrium. Second, one weakens the notion of Nash equilibrium, for instance by using probabilities. That is how Nash proved that every finite strategic game has a probabilistic Nash equilibrium.

All this work was done for games involving payoffs that are real numbers, a few years after this restriction was consciously made and justified by von Neumann and Morgenstern who also considered abstract payoffs without further studying them. Due to great success of the real-number restriction, it soon became a principle for most of the game theorists. Unfortunately, this shift from a conscious restriction to a dogma may prevent alternative approaches from emerging. Some of these alternative approaches may be not only interesting for themselves, but they also may help understand better the traditional approach.

This thesis proposes to suppress the "real-number-only dogma", and to consider alternative approaches. It introduces very abstract formalisms that generalise the notion of strategic games and the notion of Nash equilibrium. Subsequently, not all these abstract games have (abstract) Nash equilibria. This thesis exploits both Kuhn's technique and Nash's technique to cope with this issue. Along Kuhn and his precursors (*e.g.* Zermelo), this thesis introduces the notion of abstract sequential game and substantially generalises Kuhn's result, all of this being fully formalised using the proof assistant Coq. Then it generalises abstract sequential games in graphs and thus further generalises Kuhn's result. Along Nash and his precursors (*e.g.* Borel), this thesis considers ways of weakening the notion of Nash equilibrium to guarantee existence for every game. The probabilistic approach is irrelevant in this new settings, but depending on the new setting, either a notion of discrete non-determinism or the notion of sink for strongly connected component in a digraph will help solve the problem.

Acknowledgements

The topic of my PhD, generalisation and formalisation in game theory, was suggested by Pierre Lescanne. This topic sustained my excitement for three whole years, and I even enjoyed writing the dissertation. I also appreciate both the great freedom of research that I was allowed and Pierre's availability, especially for discussions that raised my interest in history of science.

In the beginning of my PhD, René Vestergaard's work on formal game theory inspired part of my work; in the end of my PhD, Martin Ziegler introduced me to computable analysis. In the course of our research on computable analysis, I benefited from the knowledge of Emmanuel Jeandel and Vincent Nesme from the information science department in ENS Lyon, as well as Etienne Ghys and Bruno Sévenec from the mathematics department in ENS Lyon.

During these three years, I held a teaching position in ENS Lyon. Victor Poupet and Damien Pous's explanations helped me a lot to fulfil my teaching duties. Generally speaking I also benefited from research courses taken at ENS Lyon, UCBL or JAIST, and from various scientific discussions with Philippe Audebaud, Florent Becker, Jean Duprat, Nick Galatos, Laurent Lyaudet, Guillaume Melquiond, Sylvain Perifel, Jean-Baptiste Rouquier, Eric Thierry, Matthias Wiesmann. Better memory and assessment ability would substantially increase the length of this list. (I hereby tackle the main issue of the acknowledgment section.)

Also, Jingdi Zeng's proofreading made my dissertation more readable than its early draft, and multiple rehearsals at ENS Lyon (in the team Plume) and at the MSR-INRIA joint centre (at Assia Mahboubi's suggestion) helped improve my PhD defense quickly.

Finally, most of the comments of the reviewers and jury members lead to useful adjustments in the dissertation.

Chapter 1

Introduction

This introduction is composed of four parts. First, it presents the thesis in a synthetic manner. Second, it discusses a few facets of proof theory that are useful to the thesis. Third, it presents the game theoretical context of the thesis. Fourth, it describes the contributions of the thesis.

1.1 Extended Abstract

This extended abstract presents the thesis in four points. First, it surveys briefly the relevant context of the thesis. Second, it mentions the subject and presents the methodology. Third, it categorises the contributions into chapters. Fourth, it suggests possible research directions.

Context

The title of the thesis is composed of three phrases, namely "abstraction", "formalisation", and "game theory". While these phrases may sound familiar to the reader, a quick explanation may be required nevertheless. The following therefore explains the meaning of these words within the scope of this thesis.

Abstraction is a classic mathematical operation, and more widely a scientific tool. It consists in suppressing assumptions and definitions that are useless with respect to a given theorem, while of course preserving the correctness of the theorem. This process yields object with fewer properties, which thus broadens the scope of the theorem. An abstraction is therefore a generalisation.

Formalisation was already a dream of Leibniz. It consists in writing proofs that can be verified automatically, *i.e.* with a computer. Actual projects intending to develop tools for such dreams started several decades ago. Thanks to a series of breakthroughs in the field of proof theory, proof formalisation is now possible. A few softwares share this niche market. Coq is one of them, and it is being developed by INRIA. Coq relies on the Curry-Howard correspondence (proposition=type and proof=programme). This allows for constructive proofs from which certified computer programs can be extracted "for free".

Game theory became a research area on its own after the work of von Neumann and Morgenstern in 1944. (Although a few game theoretic results were

stated earlier.) Game theory helps model situations in economics, politics, biology, information science, etc. These situations usually involve two or more agents, *i.e.* players. These agents have to take decisions when playing a game by given rules. In the course a play, often in the end, the agents are rewarded with payoffs. Despite its vague definition that allows for various usages, game theory seems to be very seldom the target of abstraction effort. For instance, agents' payoffs are most of the time assumed to be real numbers. As such, they are compared through the usual and implicit total order over the real numbers. Nevertheless, a fair amount of concrete situations would benefit from being modelled with alternative ordered structures.

Subject and Methodology

This thesis adopts an abstract and formal approach to game theory. It mostly discusses the concepts of game and Nash-like equilibrium in such a game. The main steps and features of the approach are described in the following.

First, basic concepts such as "game" or "Nash equilibrium" are abstracted as much as possible in order to identify their essence (more precisely, *one* essential understanding). Second, the concepts are described in a simple formalism, *i.e.* a formalism that invokes as little as possible mathematical background and that requires as little as possible mathematical development. The first two stages are often required before encoding the concepts and proving results about them using the constructive proof assistant Coq. This encoding provides an additional guarantee of proof correctness. Moreover, the process of formalisation, when done properly, requires extreme simplifications in terms of definitions, statements and proofs. This helps understand how the proofs articulate. In addition, constructive proofs give a better intuition on how things work, so they are preferred over non-constructive proofs. These proof clarifications permit generalising current game theoretic results and introducing new relevant concepts. Working at a high level of abstraction yields objects with few properties. However in this thesis, when a choice is required, discrete is preferred over continuous and finite over infinite, which seems to contradict the current mainstream research directions in game theory (but not in informatics). For instance, probabilities are not used in the definitions, statements, and proofs that are proposed in this thesis. (Probabilities might be useful in further research though.)

Despite the philosophical claim above, not all the results of this thesis are formalised using Coq. Actually, most of the results are not formalised, because it is not necessary (or even possible) to formalise everything. However, the experience of proof formalisation helps keep things simple, as if they were to be actually formalised.

Contributions

Below are listed contributions that mostly correspond to chapters of the thesis. These contributions sometimes relate to areas of (discrete) mathematics that are beyond the scope of game theory. Indeed, formalising a theorem requires to formalise lemmas on which the theorem relies, whereas these stand alone lemmas are not always connected directly to the area of the theorem.

Topological sorting

Topological sorting usually refers to linear extension of binary relations in a finite setting. Here, the setting is semi-finite. A topological sorting result is formalised using Coq. This chapter presents the proofs in detail, and it is readable by a mathematician who is not familiar with Coq. These results are invoked in the next chapter.

Generalisation of Kuhn's theorem

Traditional sequential games are labelled rooted trees. This chapter abstracts them together with their Nash equilibria and their subgame perfect equilibria. These abstractions enable a generalisation of Kuhn's theorem (stating existence of a Nash equilibrium for every sequential game). The proof of the generalisation proceeds by structural induction on the definition of abstract sequential games. Moreover, this yields an equivalence property instead of a simple implication. These results are fully formalised using Coq, and they refer to the Coq proof of topological sorting. The chapter is designed to be readable by a reader who is not interested in Coq, as well as a reader who demands details about the Coq formalisation.

An alternative proof to the generalisation of Kuhn's theorem

This chapter presents a second proof of the generalisation of Kuhn's theorem. Its structure and proof techniques are different from the ones that are involved in the previous chapter. Especially, the concept of subgame perfect equilibrium is not required in this proof that proceeds by induction on the size of the games rather than structural induction on the definition of games. In addition, the induction hypotheses are invoked at most twice, while they are invoked arbitrarily (although finitely) many times in the first proof.

Path optimisation in graphs

This chapter is a generic answer to the problem of path optimisation in graphs. For some notion of equilibrium, the chapter presents both a sufficient condition and a necessary condition for equilibrium existence in every graph. These two conditions are equivalent when comparisons relate to a total order. As an example, these results are applied to network routing.

Generalisation of Kuhn's theorem in graphs

Invoking the previous chapter, a few concepts are generalised in graphs: abstract sequential game, Nash equilibrium, and subgame perfect equilibrium. This enables a second and stronger generalisation of Kuhn's theorem. In passing, the chapter also shows that subgame perfect equilibria could have been named Nash equilibria.

Discrete and dynamic compromising equilibrium

This contribution corresponds to two chapters that introduce two abstractions of strategic game and (pure) Nash equilibrium. The first abstraction adopts

a "convertibility preference" approach; the second abstraction adopts a "best response" approach. Like strategic games, both abstractions lack the guarantee of Nash equilibrium existence. To cope with this problem in the strategic game setting, Nash introduced probabilities into the game. He weakened the definition of equilibrium, and thus guaranteed existence of a (weakened) equilibrium. However, probabilities do not seem to suit the above-mentioned two abstractions of strategic games. This suggests to "think different". For both abstractions, a discrete and dynamic notion of equilibrium is defined in lieu of the continuous, *i.e.* probabilistic, and static notion that was defined by Nash. Equilibrium existence is also guaranteed in these new settings, and computing these equilibria has a polynomial (low) algorithmic complexity. Beyond their technical contributions, these two chapters are also useful from a psychological perspective: now one knows that there are several relevant ways of weakening the definition of (pure) Nash equilibrium in order to guarantee equilibrium existence. One can thus "think different" from Nash without feeling guilty.

Discrete and static equilibrium

This chapter presents an approach that lies between Nash's probabilistic approach and the abstract approach above. Since mixing strategies with probabilities becomes irrelevant at an abstract level, a notion of discrete and non deterministic Nash equilibrium is introduced. Existence of these equilibria is guaranteed. Then, multi strategic games are defined. They generalise both strategic games and sequential games in graphs. The discrete and non deterministic approach provides multi strategic games with non deterministic equilibria that are guaranteed to exist.

Further exploration

The thesis suggests various research directions, such as:

- A few generalisations that are performed in the thesis are not optimal in the sense that they lead to implications that may not (yet) be equivalences.
- In strategic games, it may be possible to define a relevant notion of equilibrium that guarantees (pure) equilibrium existence.
- The notion of recommendation is still an open problem. It may be possible to define a relevant notion of recommendation thanks to the notion of discrete non-determinism that is used in the thesis.

1.2 Proof Theory

This section adopts a technical and historical approach. It presents the main ingredients of the proof assistant Coq in four subsections, namely inductive methods, constructivism, the Curry-De Bruijn-Howard correspondence, and constructive proof assistant in general.

A Historical View on Inductive Methods

Acerbi [2] identifies the following three stages in the history of proof by induction. First, an early intuition can be found in Plato's *Parmenides*. Second, in 1575, Maurolico [34] showed by an inductive argument that the sum of the first n odd natural numbers equals n^2 . Third, Pascal seems to have performed fully conscious inductive proofs. Historically, definitions by induction came long after proofs by induction. In 1889, even though the Peano's axiomatization of the natural numbers [42] referred to the successor of a natural, it was not yet an inductive definition but merely a property that had to hold on *pre-existing* naturals. Early XXth century, axiomatic set theory enabled inductive definitions of the naturals, like von Neumann [54], starting from the empty set representing zero. Beside the natural numbers, other objects also can be inductively/ recursively defined. According to Gochet and Gribomont [16], primitive recursive functions were introduced by Dedekind and general recursive functions followed works of Herbrand and Gödel; since then, it has been also possible to define sets by induction, as subsets of known supersets. However, the inductive definition of objects from scratch, *i.e.*, not as part of a greater collection, was mainly developed through recursive types (*e.g.*, lists or trees).

Constructivism in Proof Theory

Traditional mathematical reasoning is ruled by *classical logic*. First attempts to formalize this logic can be traced back to ancient Greeks like Aristotle [4] who discussed the principle of *proof by contradiction* among others: to prove a proposition by contradiction, one first derives an absurdity from the denial of the proposition, which means that the proposition can *not not* hold. From this, one eventually concludes that the proposition must hold. This principle is correct with respect to classical logic and it yields elegant and economical proof arguments. For example, a proof by contradiction may show the existence of objects complying with a given predicate without exhibiting a constructed witness: if such an object can *not not* exist then it must exist. At the beginning of the XXth century, many mathematicians started to think that providing an actual witness was a stronger proof argument. Some of them, like Brouwer, would even consider the proof by contradiction as a wrong principle. This mindset led to *intuitionistic logic* and, more generally, to *constructivist logics* formalized by Heyting, Gentzen, and Kleene among others. Instead of the principle of proof by contradiction, intuitionists use a stricter version stating only that an absurdity implies anything. Intuitionistic logic is smaller than classical logic in the sense that any intuitionistic theorem is also a classical theorem, but the converse does not hold. In [52], a counter-example shows that the intermediate value theorem is only classical, which implies the same for the Brouwer fixed point theorem. The principle of excluded middle states that any proposition is either "true" or "false". It is also controversial and it is actually equivalent, with respect to intuitionistic logic, to the principle of proof by contradiction. Adding any of those two principles to the intuitionistic logic yields the classical logic. In this sense, each of those principles captures the difference between the two logics.

The Curry-De Bruijn-Howard Correspondence

Nowadays, intuitionistic logic is also of interest due to practical reasons: the *Curry-De Bruijn-Howard* correspondence identifies intuitionistic proofs with functional computer programs and propositions with types. For example a program f of type $A \rightarrow B$ is an object requiring an input of type A and returning an output of type B . By *inhabiting* the type $A \rightarrow B$, the function f is also a proof of “ A implies B ”. This vision results from many breakthroughs in proof and type theories: type theory was first developed by Russell and Whitehead in [56] in order to cope with paradoxes in naive set theory. People like Brouwer, Heyting, and Kolmogorov had the intuition that a proof was a method (or an algorithm, or a function), but could not formally state it at that time. In 1958, Curry saw a connection between his combinators and Hilbert’s axioms. Later, Howard [47] made a connection between proofs and lambda terms. Eventually, De Bruijn [38] stated that the type of a proof was the proven proposition.

Constructive Proof Assistants

The Curry-De Bruijn-Howard Correspondence led to rather powerful *proof assistants*. Those pieces of software verify a proof by checking whether the program encoding the proof is well-typed. Accordingly, proving a given proposition amounts to providing a program of a given type. Some basic proof-writing steps are automated but users have to code the “interesting” parts of the proofs themselves. Each single step is verified, which gives an additional guarantee of the correctness of a mathematical proof. Of course this guarantee is not absolute: technology problems (such as software or hardware bugs) may yield validation of a wrong proof and human interpretations may also distort a formal result. Beside level of guarantee, another advantage is that a well-structured formal proof can be translated into natural language by mentioning all and only the key points from which a full formal proof can be easily retrieved. Such a reliable summary is usually different from the sketch of a “proof” that has not been actually written. An advantage of intuitionistic logic over classical logic is that intuitionistic proofs of existence correspond to search algorithms and some proof assistants, like *Coq*, are able to automatically extract an effective search program from an encoded proof, and the program is certified for free. Details can be found on the Coq website [1] and in the book by Bertot and Casteran [9].

1.3 Game Theory

This section first describes briefly game theory from an historical viewpoint. Second, it presents the notion of strategic game and Nash equilibrium. Third, it discusses Nash’s theorem. Fourth, it presents the notion of sequential game and (sequential) Nash equilibrium. Fifth, it discusses Kuhn’s theorem. Sixth, it mentions and questions the payoffs being traditionally real numbers. Seventh, it discusses graph structure for games that may involve both sequential and simultaneous decision making.

1.3.1 General Game Theory

Game theory embraces the theoretical study of processes involving (more or less conscious) possibly interdependent decision makers. Game theory originates in economics, politics, law, and also games dedicated to entertainment. Instances of game theoretic issues may be traced back to Babylonian times when the Talmud would prescribe marriage contracts that seem to be solutions of some relevant games described in [20]. In 1713, a simple card game raised questions and solutions involving probabilities, as discussed in [6]. During the XVIIth and XVIIIth centuries, philosophers such as Hobbes [19] adopted an early game theoretical approach to study political systems. In 1838, Cournot [14] introduced the notion of equilibrium for pricing in a duopoly, *i.e.* where two companies compete for the same market sector. Around 1920, Borel (*e.g.* [12]) also contributed to the field but it is said that game theory became a discipline on its own only in 1944, when von Neumann and Morgenstern [39] published a summary of prior works and a systematic study of a few classes of games. In 1950, the notion of equilibrium and the corresponding solution concept discussed by Cournot were generalised by Nash [36] for a class of games called *strategic games*. In addition to economics, politics and law, modern game theory is consciously involved in many other fields such as biology, computer science, and sociology.

1.3.2 Strategic Game and Nash Equilibrium

A strategic game involves a (non-empty) set of agents. Each agent has a (non-empty) set of available options. In game theory, these options are called strategies. A combination of agents' strategies, one strategy per agent, is called a strategy profile. To each strategy profile is attached a payoff function which rewards each agent with a payoff. The following example involves two agents named V , which stands for vertical, and H , which stands for horizontal. Agent V has two available strategies: v_1 and v_2 . Agent H has three available strategies: h_1 , h_2 , and h_3 . So there are two times three, *i.e.* six, strategy profiles. The top-left profile is (v_1, h_1) . To this profile corresponds a payoff function that rewards V with 0 and H with 1.

	h_1	h_2	h_3
v_1	0 1	0 0	1 2
v_2	2 2	1 0	1 1

The strategy profile (v_2, h_2) in the example above induces payoff 0 for agent H . Assume that agent V does not change his choice. By changing his own choice, *i.e.* by choosing strategy either h_1 or h_3 , agent H can convert the profile (v_2, h_2) into the profile either (v_2, h_1) or (v_2, h_3) . He would thus get a better payoff, *i.e.* 2 or 1. Therefore agent H is somehow unhappy with the strategy profile (v_2, h_2) . On the contrary, agent V is happy. Nash defined equilibria as strategy profiles that make all agents happy. The Nash equilibria of the game above are (v_2, h_1) and (v_1, h_3) . However, the example below depicts a game without any Nash equilibrium.

	h_1	h_2
v_1	1 0	0 1
v_2	0 1	1 0

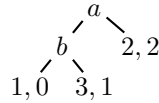
1.3.3 Nash's Theorem

As seen in the above subsection, not all finite strategic games have a Nash equilibrium. Since a guarantee of existence is a desirable property, Nash weakened his definition of equilibrium to assure the existence of a "compromising" equilibrium. More specifically, introducing probabilities into the game, he allowed agents to choose their individual strategies with a probability distribution rather than choosing a single strategy deterministically. Subsequently, instead of a single strategy profile chosen deterministically, Nash's probabilistic compromise involves a probability distribution over strategy profiles. So, expected payoff functions are involved instead of payoff functions. For instance, the sole probabilistic Nash equilibrium for the last game in the above subsection is the probability assignment $v_i \mapsto \frac{1}{2}, h_i \mapsto \frac{1}{2}$, and the expected payoff is $\frac{1}{2}$ for both agents. This compromise actually builds a new strategic game that is continuous. Nash proved that this new game always has a Nash equilibrium, which is called a probabilistic Nash equilibrium for the original game. A first proof of this result [36] invokes Kakutani's fixed point theorem [22], and a second proof [37] invokes (the proof-theoretically simpler) Brouwer's fixed point theorem.

1.3.4 Sequential Game and Nash Equilibrium

Another class of games is that of *sequential games*, also called *games in extensive form*. It traditionally refers to games where players play in turn till the play ends and payoffs are granted. For instance, the game of chess is often modelled by a sequential game where payoffs are "win", "lose" and "draw". Sequential games are often represented by finite rooted trees each of whose internal nodes is owned by a player, and each of whose external nodes, *i.e.* leaves, encloses one payoff per player. In 1912, Zermelo [57] proved about the game of chess that either white can win (whatever black may play), or black can win, or both sides can force a draw. This is sometimes considered as the first non-trivial theoretical results in game theory.

The following graphical example involves the two agents a and b . At every leaf, a payoff function is represented by two numbers separated by a comma: the payoff function maps agent a to the first number and agent b to the second number.



Such a game tree is interpreted as follows: A play of a game starts at the root of the game tree. If a chooses right, both he and b gets payoff 2. If a chooses left and b chooses left (resp. right), a gets 1 (resp. 3) and b gets 0 (resp. 1).

Although the concept of Nash equilibrium referred to strategic games in the first place, it is natural and relevant to extend that concept to sequential

games. The extension is made through an embedding of sequential games into strategic games. This embedding applied to the above example yields the following strategic game. The Nash equilibria of the strategic game image, namely (a_{right}, b_{left}) and (a_{left}, b_{right}) , are also called the Nash equilibria of the original sequential game.

	b_{left}	b_{right}
a_{left}	1 0	3 1
a_{right}	2 2	2 2

1.3.5 Kuhn's Theorem

In 1953, Kuhn [25] showed the existence of Nash equilibrium for sequential games. For this, he built a specific Nash equilibrium through what is called "backward induction" in game theory. In 1965, Selten ([48] and [49]) introduced the concept of subgame perfect equilibrium in sequential games. This is a refinement of Nash equilibrium that seems to be even more meaningful than Nash equilibrium for sequential games. The two concepts of "backward induction" and subgame perfect equilibrium happen to coincide, so Kuhn's result also guarantees existence of subgame perfect equilibrium in sequential games.

A subgame perfect equilibrium is a Nash equilibrium whose substrategy profiles are also subgame perfect equilibria. For instance on the left-hand sequential game below, (a_{right}, b_{left}) is not a subgame perfect equilibrium because b_{left} is not a Nash equilibrium for the subgame that is displayed on the right-hand side below. However, the Nash equilibrium (a_{left}, b_{right}) is a subgame perfect equilibrium because b_{right} is a Nash equilibrium for the subgame.



"Backward induction" is a recursive procedure that expects a sequential games and returns a strategy profile for this game. It works as follows. On the subgame above (on the right-hand side), agent b chooses b_{right} because he prefers 1 over 0. On the compound game above (on the left-hand side), a chooses a_{left} because he "knows" that b chooses b_{right} . This yields the profile (a_{left}, b_{right}) , which is the only subgame perfect equilibrium of the compound game.

In 2006, Vestergaard [53] formalised part of Kuhn's result with the proof assistant Coq, for the subclass of games represented by binary trees and whose payoffs range over the natural numbers. For this, he defined sequential games and corresponding strategy profiles inductively.

1.3.6 Ordering Payoffs

Game theory has mostly studied games with real-valued payoffs, perhaps for the following reason: In 1944, von Neumann and Morgenstern [39] suggested that the notion of payoff in economics could be reduced to real numbers. They argued that more and more physical phenomena were measurable; therefore,

one could reasonably expect that payoffs in economics, although not yet measurable, would become reducible to real numbers some day. However, game theory became popular soon thereafter, and its scope grew larger. As a result, several scientists and philosophers questioned the reducibility of payoffs to real numbers. In 1955, Simon [50] discussed games where agents are awarded (only partially ordered) vectors of real-valued payoffs instead of single real-valued payoffs. In 1956, Blackwell [10] proved a result involving vectors of payoffs. Those vectors model agents that take several non-commensurable dimensions into consideration; such games are sometimes called multi criteria games. More recent results about multi criteria games can be found in [44], for instance. In 1994, Osborne and Rubinstein [40] mentioned arbitrary preferences for strategic games, but without any further results. In 2003, Krieger [24] noticed that "backward induction" on sequential multi criteria games may not yield Nash equilibria, and yet showed that sequential multi criteria games have Nash equilibria. The proof seems to invoke probabilities and Nash's theorem for strategic games.

1.3.7 Graphs and Games, Sequential and Simultaneous

Traditional game theory seems to work mainly with strategic games and sequential games, *i.e.* games whose underlying structure is either an array or a rooted tree. These game can involve many players. On the contrary, combinatorial game theory studies games with various structures, for instance games in graphs. It seems that most of these combinatorial games involve two players only. Moreover, the possible outcomes at the end of most of these games are "win-lose", "lose-win", and "draw" only. The book [7] presents many aspects of combinatorial game theory.

Chess is usually thought as a sequential tree game. However, plays in chess can be arbitrarily long (in terms of number of moves) even with the fifty moves rules which says that a player can claim a draw if no capture has been made and no pawn has been moved in the last fifty consecutive moves. So, the game of chess is actually defined through a directed graph rather than a tree, since a play can enter a cycle. Every node of the graph is made of both the location of the pieces on the chessboard and an information about who has to move next. The arcs between the nodes correspond to the valid moves. So the game of chess is a bipartite digraph (white and black play in turn) with two agents. This may sound like a detail since the game of chess is well approximated by a tree. However, this is not a detail in games that may not end for intrinsic reasons instead of technical rules: for instance poker game or companies sharing a market. In both cases, the game can continue as long as there are at least two players willing to play. In the process, a sequence of actions can lead to a situation similar to a previous situation (in terms of poker chips or market share), hence a cycle.

Internet, too, can be seen as a directed graph whose nodes represent routers and whose arcs represent links between routers. When receiving a packet, a router has to decide where to forward it to: either to a related local network or to another router. Each router chooses according to "it's owner interest". Therefore Internet can be seen as a digraph with nodes labelled with owners of routers. This digraph is usually symmetric since a link from router A to router B can be easily transformed to a link between router b and router A.

Moreover, the interests of two different owners, *i.e.* Internet operators, may be contradictory since they are supposed to be competitors. Therefore the owners' trying to maximise their benefits can be considered a game. Local benefits (or costs) of a routing choice may be displayed on the arcs of the graph.

Many systems from the real world involve both sequential and simultaneous decision-making. For instance, a country with several political parties is a complex system. Unlike chess, poker, and Internet, rules may not be definable by a digraph, but at least the system may be modelled by a graph-like structure: The nodes represent the political situations of the country, *i.e.* which party is in power at which level, etc. At each node, simultaneous decisions are taken by the parties, *i.e.* vote a law, start a campaign, design a secret plan, etc. The combination of the decisions of the parties yields a short-term outcome (good for some parties, bad for some others) and leads to another node where other decisions are to be taken. This process may enter a cycle when a sequence of actions and elections leads to a political situation that is similar to a previous situation, *i.e.* the same parties are in power at the same levels as before. In such complex a setting, some decision-making processes are sequential, some are simultaneous, and some involve both facets.

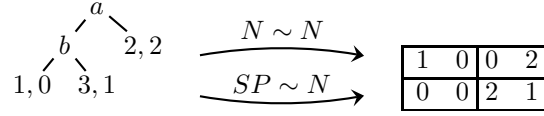
1.4 Contributions

This section accounts for the different contributions of the thesis. For presentation reasons, it does not follow exactly the structure of the dissertation. First, the section discusses extending the notion of Nash equilibrium from strategic games to sequential games. Second, it replaces real-valued payoff function with abstract outcomes. Third, it replaces specific structures such as tree and array with loose structures that still allows a game-theoretic definition of Nash equilibrium. Fourth, it discusses a necessary and sufficient condition for Nash equilibrium existence in abstract sequential tree games. Fifth, it mentions a different proof of the above result. Sixth, it defines sequential games on graphs. Seventh, it defines dynamic equilibria in the loose game structures as a compromise to guarantee existence of equilibrium. Eighth, it defines discrete non-determinism and uses it in a general setting.

1.4.1 Will Nash Equilibria Be Nash Equilibria?

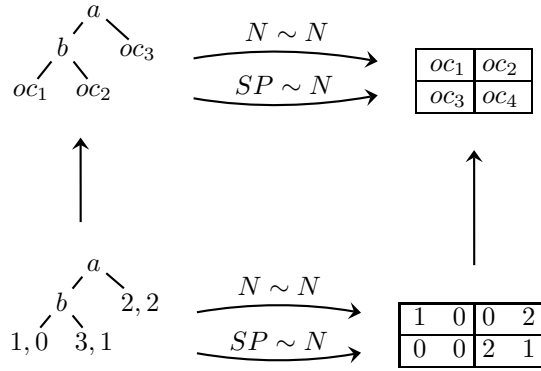
Subsection 1.3.4 explains how sequential games are embedded into strategic games, and how the Nash equilibria of a sequential game are defined as the pre-images of the Nash equilibria of the corresponding strategic game. This embedding sounds natural. Nevertheless, subsection 7.3.2 shows that there exists a similar embedding that performs a slightly different operation: the pre-images of the Nash equilibria of the corresponding strategic game are exactly the subgame perfect equilibria of the sequential game. This new embedding is also relevant, which suggests that the sequential strategy profiles that were baptised Nash equilibria were not the only candidates. The two embeddings are represented in the picture below, where the figures are arbitrary. The symbol $N \sim N$ represents the traditional correspondence between the Nash equilibria in sequential games and the Nash equilibria in strategic games. The symbol $SP \sim N$ represents the new correspondence between the subgame perfect

equilibria in sequential games and the Nash equilibria in strategic games.



1.4.2 Abstracting over Payoff Functions

Following subsection 1.3.6, this thesis replaces payoff functions with abstract objects named outcomes. In the world of real-valued payoff functions, an agent can compare different payoff functions by comparing the payoffs that are granted to him by the functions, which uses implicitly the usual total order over the real numbers. In the more abstract world, each agent has a preference over outcomes, which is given *via* an explicit and arbitrary binary relation. This abstraction yields abstract strategic games and abstract sequential games where Nash equilibria and subgame perfect equilibria are easily extended. The abstraction over the two traditional games and the embeddings between the resulting objects are represented below.

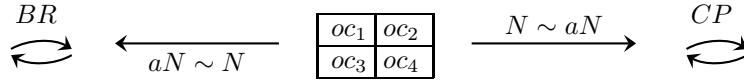


Note that this generalisation is done by abstraction over payoff functions only. More specifically, the structure of the games remains the same, *i.e.* tree-like and array-like. These abstractions are not merely performed for the fun of it, as discussed in subsection 1.4.4.

1.4.3 Abstracting over Game Structure

Chapter 2 rephrases the traditional definitions of strategic game and Nash equilibrium, which are illustrated by a few examples. These concepts are presented in a way that motivates the introduction of two different abstractions of strategic games and Nash equilibria. These abstractions are convertibility preference (CP) games and best response (BR) games. (The notion of abstract preference was already mentioned in works such as [40], but no further result seemed to follow.) Both abstractions allow defining the notion of happiness for agents participating in the game. Within both abstractions, abstract Nash equilibria are defined as objects that give happiness to all agents. An embedding

shows that CP (resp. BR) games are abstractions of strategic games and that abstract Nash equilibria in CP (resp. BR) games are abstractions of Nash equilibria. More specifically, CP and BR games are intended to be abstract and general while still allowing a game theoretic definition of Nash equilibrium. CP games are also defined in [30], and they are either re-explained or applied to biology in [13] and [31]. The picture below represents the two above-mentioned embeddings, where the symbol $N \sim aN$ means that Nash equilibria correspond to abstract Nash equilibria.



These new formalisms may not lead directly to difficult theoretical results or practical applications. However, they give an essential understanding of what is a Nash equilibrium. This thesis makes use of these viewpoints, and any notion of equilibrium discussed in the thesis is an instance of either CP or BR equilibrium. Actually, these viewpoints might be part of the folklore of game theory, but no document that was encountered while preparing this thesis seems to define anything similar to CP or BR games. In addition, this chapter presents the traditional notion of strict Nash equilibrium. This notion corresponds to abstract strict Nash equilibria that are also defined in the BR game formalism.

1.4.4 Acyclic Preferences, Nash and Perfect Equilibria: a Formal and Constructive Equivalence

Chapter 4 contributes at both the technical and the presentation level. There are five main technical contributions: First, an inductive formalism is designed to represent sequential games in the constructive proof assistant Coq ([1] and [9]), and all the results in this chapter are proved in Coq. Second, the new formalism allows representing abstract sequential games and a few related concepts. Third, Kuhn's result [25] is translated into the new formalism when agents' preferences are totally ordered. Fourth, the notion of "backward induction" is naturally generalised for arbitrary preferences. However, a simple example shows that a structure such as total ordering (more specifically, strict weak ordering) of preferences is needed for "backward induction" to guarantee subgame perfect equilibrium: both notions of "backward induction" and subgame perfect equilibrium coincide for total orders but not in general. Fifth, Kuhn's result is substantially generalised as follows. On the one hand, an intermediate result proves that smaller preferences, *i.e.*, binary relations with less arcs, yield more equilibria than bigger preferences. On the other hand, a topological sorting result was formally proved in [28] and chapter 3. By both results mentioned above, acyclicity of the preferences proves to be a necessary and sufficient condition for every game to have a Nash equilibrium/subgame perfect equilibrium.

This chapter deals with basic notions of game theory that are all exemplified and defined before they are used. Most of the time, these notions are explained in three different ways, with the second one helping make the connection between the two others: First, the notions are presented in a graphical formalism

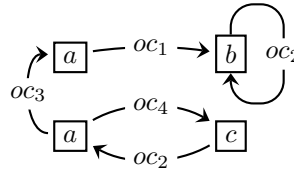
close to traditional game theory. Second, they are presented in a graphical formalism suitable for induction. Third, they are presented in a light Coq formalism close to traditional mathematics, so that only a basic understanding of Coq is needed. (A quick look at the first ten pages of [28] or chapter 3 will introduce the reader to the required notions.) The proofs are structured along the corresponding Coq proofs but are written in plain English.

1.4.5 Acyclic Preferences and Nash Equilibrium Existence: Another Proof of the Equivalence

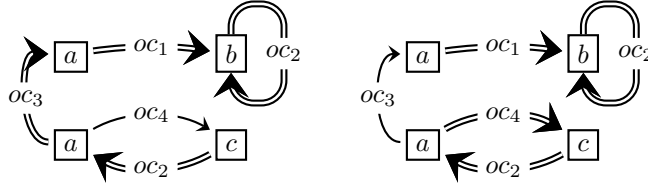
Chapter 5 proves that, when dealing with abstract sequential games, the following two propositions are equivalent: 1) Preferences over the outcomes are acyclic. 2) Every sequential game has a Nash equilibrium. This is a corollary of the triple equivalence mentioned in subsection ??, however the new proof invokes neither structural induction on games, nor “backward induction”, nor topological sorting. Therefore, this alternative argument is of proof theoretical interest. The proof of the implication $1) \Rightarrow 2)$ invokes three main arguments. First, if preferences are strict partial orders, then every game has a Nash equilibrium, by induction on the size of the game and a few cut-and-paste tricks on smaller games. Second, if a binary relation is acyclic, then its transitive closure is a strict partial order. Third, smaller preferences generate more equilibria, as seen in [29] and chapter 4. The converse $2) \Rightarrow 1)$ is proved as in [29] and chapter 4. It is worth noting that in [29], the implication $1) \Rightarrow 2)$ follows $1) \Rightarrow 3)$ and $3) \Rightarrow 2)$. However, the proof of this chapter is direct: it does not involve subgame perfect equilibria.

1.4.6 Sequential Graph Games

Chapter 7 introduces the notion of sequential graph game. Such a game involves agents and outcomes. A sequential graph game is a directed graph whose nodes are labelled with agents, whose arcs are labelled with outcomes, and each of whose node has an outgoing arc. The design choices are quickly justified, and an interpretation of these games is proposed through an informal notion of play.



A strategy profile for a sequential graph game amounts to choosing an outgoing arc at every node of the game. By changing these choices only at some nodes that he owns, an agent can convert a strategy profile into another one; this defines convertibility. The following two strategy profiles are convertible one to another by agent *a*.

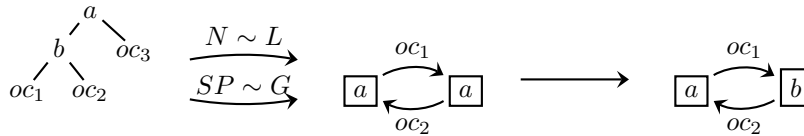


Starting from a given node, one can follow the arcs that are prescribed by a given strategy profile. This induces an infinite sequence of outcomes. For each agent, a binary relation accounts for the agent's preferences among infinite sequences of outcomes. Given a node of a sequential graph game, an agent can compare two strategy profiles for the game by comparing the induced infinite sequences at the given node; this defines preference. Having a notion of convertibility and preference for each agent, the local equilibria at given nodes are defined like the Nash equilibria of some derived CP game: they are strategy profiles that no agent can convert into a preferred profile. A global equilibrium is defined as a strategy profile that is a local equilibrium at every node of the underlying game.

It turns out that the global equilibria of a sequential graph game are exactly the Nash equilibria of some derived CP game that is different from the CP game mentioned above. In addition, the chapter defines an embedding of sequential tree games into sequential graph games. This embedding sends Nash equilibria to local equilibria and vice versa, and sends subgame perfect equilibria to global equilibria and vice versa. Therefore, local equilibrium is a generalisation of Nash equilibrium and global equilibrium is a generalisation of subgame perfect equilibrium.

In sequential tree games, subgame perfect equilibria can be built through "backward induction" following topological sorting. This chapter generalises the procedure of "backward induction" for a subclass of sequential graph games. This leads to a sufficient condition on the agents' preferences for global equilibrium existence in every game in the subclass mentioned above. It thus generalises the generalisation [29] of Kuhn's result [25], which states that every sequential game has a Nash (and subgame perfect) equilibrium. In addition, the chapter gives a necessary condition on the agents' preferences for global equilibrium existence in every game in the subclass mentioned above. For the necessary condition and the sufficient condition, which do not coincide in general, the chapter invokes some results about dalographs proved in chapter 6. However, the two conditions coincide when the preferences are total orders, which gives an equivalence property. In the same way, a sufficient condition is given for equilibrium existence in every sequential graph game.

The picture below represents sequential tree games to the left, sequential graph games to the right, and the above-mentioned subclass of sequential graph games in the centre. The symbol $N \sim L$ represents the correspondence between Nash equilibria and local equilibria, and the symbol $SP \sim G$ represents the correspondence between subgame perfect equilibria and global equilibria.



1.4.7 Abstract Compromising Equilibria

First, chapter 8 explores possible generalisations of Nash's result within the BR and CP formalisms. These generalisations invoke Kakutani's fixed point theorem, which is a generalisation of Brouwer's. However, there is a difference between Nash's theorem and its generalisations in the BR and CP formalisms: Nash's probabilised strategic games correspond to finite strategic games since they are derived from them. However, it seems difficult to introduce probabilities within a given finite BR or CP game, since BR and CP games do not have any Cartesian product structure. Therefore, one considers a class of already-continuous BR or CP games. Kakutani's fixed point theorem, which is much more appropriate than Brouwer's in this specific case, helps guarantee the existence of an abstract Nash equilibrium. However, these already-continuous BR or CP games do not necessarily correspond to relevant finite BR or CP games, so practical applications might be difficult to find.

Second, this chapter explores compromises that are completely different from Nash's probabilistic compromise. Two conceptually very simple compromises are presented in this section, one for BR games and one for CP games. The one for BR games is named or-best response strict equilibrium, and the one for CP games is named change-of-mind equilibrium, as in [30]. Both compromising equilibria are natural generalisations of Nash equilibria for CP games and strict Nash equilibria for BR games. It turns out that both generalisations are (almost) the same: they both define compromising equilibria as the sink strongly connected components of a relevant digraph. Informally, if nodes represent the microscopic level and strongly connected components the macroscopic level, then the compromising equilibria are the macroscopic equilibria of a microscopic world.

Since BR and CP games are generalisations of strategic games, the new compromising equilibria are relevant in strategic games too. This helps see that Nash's compromise and the new compromises are different in many respects. Nash's probabilistic compromise transforms finite strategic games into continuous strategic games. Probabilistic Nash equilibria for the original game are defined as the (pure) Nash equilibria for the continuous derived game. This makes the probabilistic setting much more complex than the "pure" setting. On the contrary, the new compromises are discrete in the sense that the compromising equilibria are finitely many among finitely many strongly connected components. While probabilistic Nash equilibria are static, the new compromising equilibria are dynamic. While Nash equilibria are fixed points obtained by Kakutani's (or Brouwer's) fixed point theorem, the new equilibria are fixed points obtained by a simple combinatorial argument (or Tarski's fixed point theorem if one wishes to stress the parallel between Nash's construction and this one). While probabilistic Nash equilibria are non-computable in general, the new compromising equilibria are computable with low algorithmic complexity. Finally, while probabilistic Nash equilibria are Nash equilibria of a derived continuous game, the new compromising equilibria seem not to be the Nash equilibria of any relevant derived game.

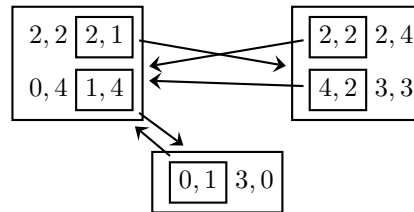
1.4.8 Discrete Non-Determinism and Nash Equilibria for Strategy-Based Games

Chapter ?? tries to do what Nash did for traditional strategic games: to introduce probabilities into abstract strategic games to guarantee the existence of a weakened kind of equilibrium. However, it is mostly a failure because there does not seem to exist any extension of a poset to its barycentres that is relevant to the purpose. So, instead of saying that "an agent chooses a given strategy with some probability", this chapter proposes to say that "the agent may choose the strategy", without further specification.

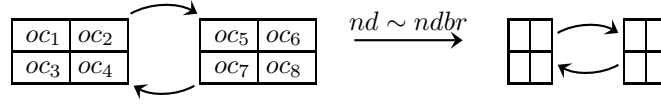
The discrete non-determinism proposed above is implemented in the notion of non deterministic best response (ndbr) multi strategic game. As hinted by the terminology, the best response approach is preferred over the convertibility preference approach for this specific purpose. (Note that discrete non-determinism for abstract strategic games can be implemented in a formalism that is more specific and simpler than ndbr multi strategic games, but this general formalism will serve further purposes.) This chapter defines the notion of ndbr equilibrium in these games, and a pre-fixed point result helps prove a sufficient condition for every ndbr multi strategic game to have an ndbr equilibrium.

This chapter also defines the notion of multi strategic game that is very similar to the notion of ndbr multi strategic game, while slightly less abstract. Informally, they are games where a strategic game takes place at each node of a graph. (A different approach to "games network" can be found in [33].) They can thus model within a single game both sequential and simultaneous decision-making mechanisms. An embedding of multi strategic games into ndbr multi strategic games provides multi strategic games with a notion of non deterministic (nd) equilibrium, and the sufficient condition for ndbr equilibrium existence is translated into a sufficient condition for nd equilibrium existence. The chapter details a concrete and simple example.

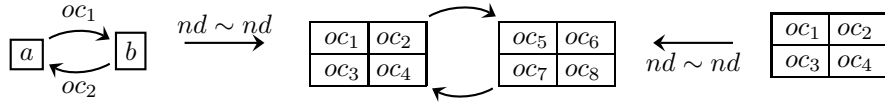
The following example depicts a multi strategic game that involves two agents, say *vertical* and *horizontal*. At each node *vertical* chooses the row and *horizontal* chooses the column. The game involves natural-valued payoff functions. The first figure corresponds to agent *vertical* and the second figure to agent *horizontal*. At a node, if a payoff function is enclosed in a small box with an arrow pointing from the box to another node, it means that the corresponding strategy profile leads to this other node. If a payoff function is not enclosed in a small box, it means that the corresponding strategy profile leads to the same node. For instance below, If the play start at the top-left node and if the agents choose the top-left profile, then both agents get payoff 2 and the same strategic game is played again. Whereas if the agents choose the top-right profile, *vertical* gets payoff 2 and *horizontal* gets payoff 1, and the agents have to play in the top-right node.



The picture below represents the multi strategic games on the left-hand side and non deterministic best response multi strategic games on the right-hand side. The symbol $nd \sim ndbr$ represents the correspondence between nd equilibria and ndbr equilibria.



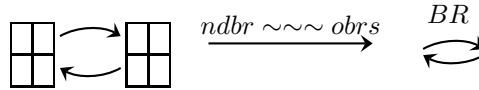
multi strategic games are actually a generalisation of both abstract strategic games and sequential graph games, as depicted below. Therefore any notion of nd equilibrium for multi strategic games can be translated to both sequential games and strategic games. This is represented in the following picture.



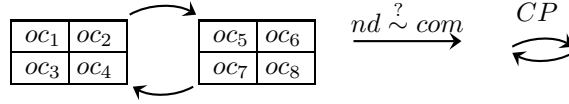
Direct embeddings of abstract strategic games into ndbr multi strategic games provide abstract strategic games with more subtle notions of non deterministic (nd) equilibrium that generalise the notion of Nash equilibrium. The chapter details a concrete and simple example. Since every abstract strategic game has an nd equilibrium, the discrete non deterministic approach succeeds where the probabilistic approach fails, *i.e.* is irrelevant. In addition, a numerical example shows that the constructive proof of nd equilibrium existence can serve as a recommendation to agents on how to play, while the notion of Nash equilibrium, as it stands, cannot lead to any kind of recommendation.

This new approach lies between Nash's approach, which is continuous and static, and the abstract approaches of CP and BR games, which are discrete and dynamic. Indeed, this notion of nd equilibrium is discrete and static. It is deemed static because it makes use of the Cartesian product structure, which allows interpreting an equilibrium as a "static state of the game".

A specific direct embedding even establishes a connection between nd equilibria in abstract strategic games and or-best response strict equilibria in BR games. This is represented in the following picture. The left-hand games are the ndbr multi strategic games. The right-hand games are the BR games. The symbol $ndbr \sim \sim \sim obrs$ represents the correspondence between or-best response strict equilibria and ndbr equilibria. The triple \sim means that this correspondence holds only for a specific embedding of abstract strategic games into ndbr multi strategic games, so it is weaker than a single \sim .



This correspondence between or-best response strict equilibria in BR games and nd equilibria in abstract strategic games is due to the . As for the change-of-mind equilibria in CP games, if there were a correspondence then it would be with nd equilibria there is no obvious correspondence with nd equilibria so far.



1.5 Convention

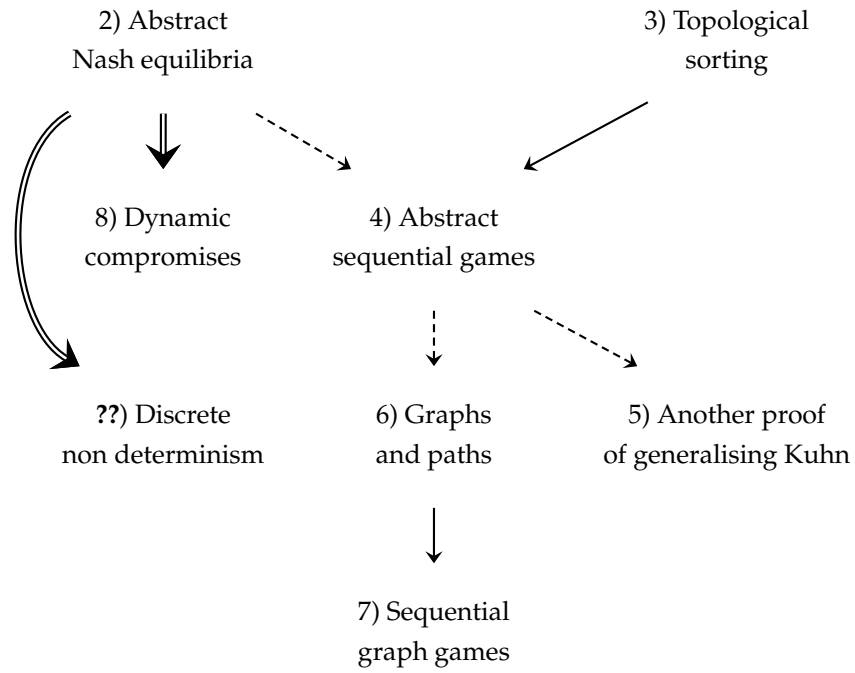
Let $E = \prod_{i \in I} E_i$ be a cartesian product.

- For e in E , let e_i be the E_i -component of e .
- Let E_{-i} denote $\prod_{j \in I - \{i\}} E_j$.
- For e in E , let e_{-i} be $(\dots, e_{i-1}, e_{i+1}, \dots)$ the projection of e on E_{-i} .
- For x in E_i and X in E_{-i} , define $X; x$ in E as $(X; x)_i = x$ and for all $j \neq i$, $(X; x)_j = X_j$.

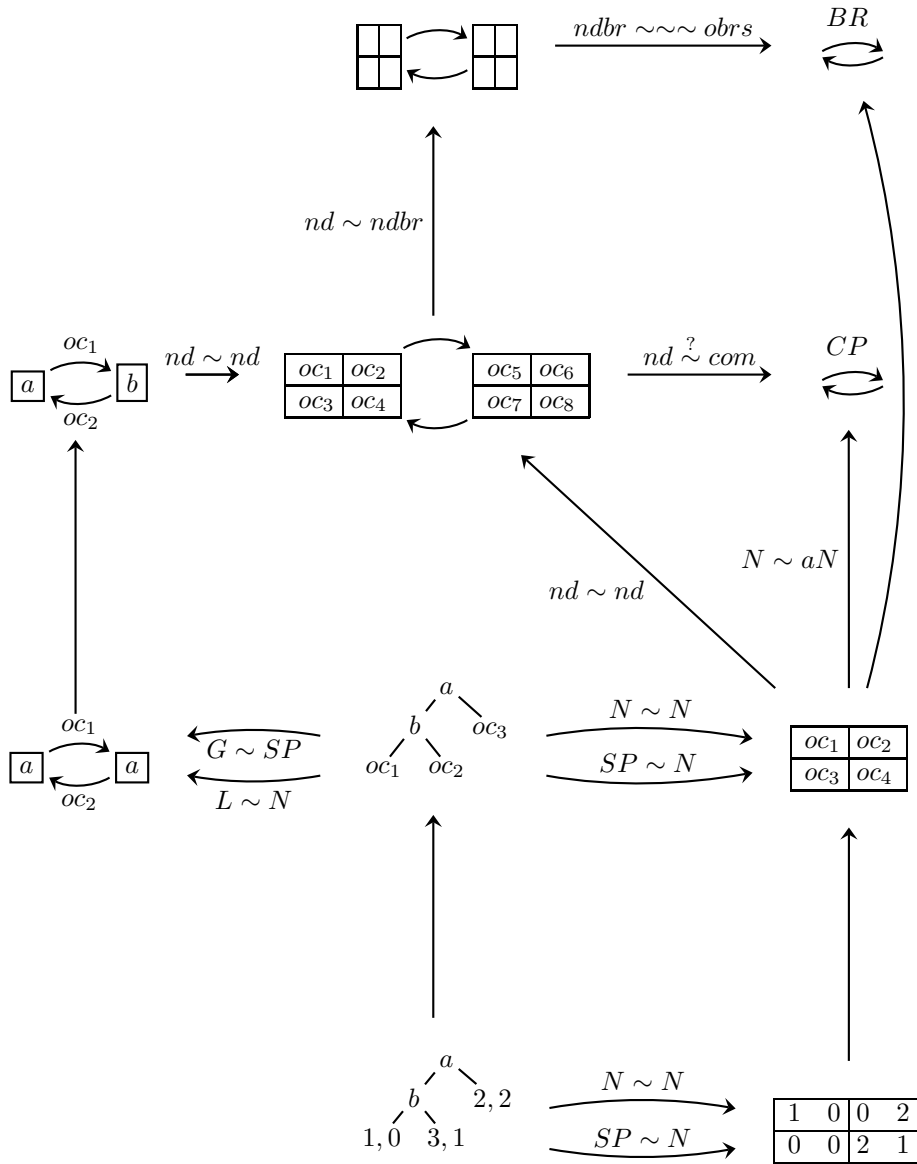
1.6 Reading Dependencies of Chapters

Although chapters are best read in the usual order, they were written to be as much as possible independent from each other. Therefore, the reader may start reading a chapter and jump back to a previous one when needed. Nevertheless, the following diagram intends to help the reader choose a reading strategy.

- $A \Rightarrow B$ means that B explicitly depends on A , e.g. by referring to definitions.
- $A \longrightarrow B$ means that B depends on A in a quasi-transparent way, e.g. by invoking a theorem.
- $A - \rightarrow B$ means that A and B are closely related and that A is best read first, although not necessarily.



1.7 Graphical Summary



Chapter 2

Abstract Nash Equilibria

2.1 Introduction

Nash introduced the concept of non-cooperative equilibrium, *i.e.* Nash equilibrium, for strategic games. This chapter shows that Nash equilibria are relevant beyond the scope of strategic games.

2.1.1 Contribution

This chapter rephrases the traditional definitions of strategic game and Nash equilibrium, which are illustrated by a few examples. These concepts are presented in a way that motivates the introduction of two different abstractions of strategic games and Nash equilibria. These abstractions are convertibility preference (CP) games (also defined in [30]) and best response (BR) games. Note that the notion of abstract preference was already mentioned in works such as [40] and that the terminology of best response already exists in game theory. Both abstractions allow defining the notion of happiness for agents participating in the game. Within both abstractions, abstract Nash equilibria are defined as objects that give happiness to all agents. An embedding shows that CP (resp. BR) games are abstractions of strategic games and that abstract Nash equilibria in CP (resp. BR) games are abstractions of Nash equilibria. More specifically, CP and BR games are intended to be abstract and general while still allowing a game theoretic definition of Nash equilibrium. These new formalisms may not lead directly to difficult theoretical results or practical applications, but they give an essential understanding of what a Nash equilibrium is. This thesis makes use of these viewpoints. Although no document that was encountered while preparing this thesis defines anything similar to CP or BR games, the informal version of the convertibility/preference viewpoint may be considered as part of the folklore. Indeed for instance, Rousseau [46] wrote: "Toute action libre a deux causes qui concourent à la produire, l'une morale, savoir la volonté qui détermine l'acte, l'autre physique, savoir la puissance qui l'exécute. Quand je marche vers un objet, il faut premièrement que j'y veuille aller ; en second lieu, que mes pieds m'y portent. Qu'un paralytique veuille courir, qu'un homme agile ne le veuille pas, tous deux resteront en place." Roughly, this may be translated as follows: Any free action has two causes,

the first one being moral, namely the will that specifies the action, the other one being physical, namely the power that executes it. When I walk towards an object, first of all it requires that I want to go there; second it requires that my feet bring me there. Whether a paralytic person wants to run or a agile man does not want to, both remain still.

In addition, this chapter presents the traditional notion of strict Nash equilibrium. This notion corresponds to abstract strict Nash equilibria that are also defined in the BR game formalism.

2.1.2 Contents

Section 2.2 rephrases the definitions of strategic game, Nash equilibrium, and strict Nash equilibrium. Section 2.3 introduces CP games and abstract Nash equilibria in CP games. Section 2.4 introduces BR games, abstract Nash equilibria, and strict abstract Nash equilibria in BR games.

2.2 Strategic Game and Nash Equilibrium

This section first rephrases the definition of strategic games, and gives a few examples using the array representation of strategic games. Then it rephrases the definition of Nash equilibria and gives examples of games with and without equilibria. Finally, it rephrases the definition of strict Nash equilibria and also gives a few examples showing that the existence of Nash equilibria and the existence of strict Nash equilibria are two orthogonal issues.

2.2.1 Strategic Game

A strategic game involves a (non-empty) set of agents. Each agent has a (non-empty) set of available options. In game theory, these options are called strategies. A combination of agents' strategies, one strategy per agent, is called a strategy profile. To each strategy profile is attached a payoff function which rewards each agent with a payoff. Strategic games are defined below.

Definition 1 (Strategic Games) *Strategic Games are 3-tuples $\langle \mathcal{A}, \mathcal{S}, P \rangle$, where:*

- \mathcal{A} is a non-empty set of agents.
- \mathcal{S} is the Cartesian product $\bigotimes_{a \in \mathcal{A}} \mathcal{S}_a$ of non-empty sets of individual strategies \mathcal{S}_a . Elements of \mathcal{S} are named strategy profiles.
- $P : \mathcal{S} \rightarrow \mathbb{R}$ is a function from strategy profiles to real-valued payoff function.

Let s range over \mathcal{S} , let s_a be the a -projection of s , and let s_{-a} be the $(\mathcal{A} - \{a\})$ -projection of s .

When only two agents are involved in a strategic game, it may be convenient to represent the game as an array. The following example involves two agents named V , which stands for vertical, and H , which stands for horizontal. Agent V has two available strategies: v_1 and v_2 . Agent H has three available strategies: h_1 , h_2 , and h_3 . So there are two times three, *i.e.* six, strategy profiles.

The top-left profile is (v_1, h_1) . To this profile corresponds a payoff function that rewards V with 0 and H with 1.

	h_1		h_2		h_3	
v_1	$V \mapsto 0$	$H \mapsto 1$	$V \mapsto 0$	$H \mapsto 0$	$V \mapsto 1$	$H \mapsto 2$
v_2	$V \mapsto 2$	$H \mapsto 2$	$V \mapsto 1$	$H \mapsto 0$	$V \mapsto 1$	$H \mapsto 1$

The representation above can be simplified into the array below. It becomes implicit that the first figure is associated with V , the agent choosing vertically, and that the second figure is associated with H , the agent choosing horizontally. This is consistent with the usual matrix representation: given M a matrix, $M_{i,j}$ is the element at row i and column j .

	h_1	h_2	h_3
v_1	0 1	0 0	1 2
v_2	2 2	1 0	1 1

Strategic games can be interpreted through the informal notion of play: at the beginning of the game each agent chooses secretly one of his options. Then all choices are frozen and disclosed. The combination of agents' strategies forms a strategy profile, which in turn induces a payoff function. Agents are rewarded accordingly. On one single schema, one can represent a strategic game, a related strategy profile, and the induced payoff function, as shown below. This may represent a play too.

	h_1	$[h_2]$	h_3
v_1	0 1	0 0	1 2
$[v_2]$	2 2	1 0	1 1

2.2.2 Nash Equilibrium

Consider the strategic game, related strategy profile, and induced payoffs defined in the end of subsection 2.2.1. The current strategy profile induces payoff 0 for agent H . Assume that agent V does not change his choice. By changing his own choice, *i.e.* by choosing strategy h_1 or h_3 , agent H could get a better payoff, *i.e.* 2 or 1. Therefore agent H is somehow unhappy with the current strategy profile. On the contrary, agent V is happy. A more formal definition of happiness is given below.

Definition 2 (Agent happiness) *Given a strategic game $g = \langle \mathcal{A}, \mathcal{S}, P \rangle$, an agent a being happy with a strategy profile s is defined as follows.*

$$\text{Happy}(a, s) \triangleq \forall s' \in \mathcal{S}, \quad \neg(s'_{-a} = s_{-a} \wedge P(s, a) < P(s', a))$$

Nash defined equilibria as strategy profiles that make all agents happy. A more formal definition is given below.

Definition 3 (Nash equilibrium) Given a strategic game $g = \langle \mathcal{A}, \mathcal{S}, P \rangle$, strategy profile s being a Nash equilibrium for game g is defined as follows.

$$\text{Eq}_g(s) \triangleq \forall a \in \mathcal{A}, \text{Happy}(a, s)$$

The definition above can be slightly rewritten to help decide whether or not a given profile is a Nash equilibrium.

Lemma 4

$$\text{Eq}_g(s) \Leftrightarrow \forall a \in \mathcal{A}, s' \in \mathcal{S}, s'_{-a} = s_{-a} \Rightarrow \neg(P(s, a) < P(s', a))$$

Given a strategic game, an agent, and a strategy profile, only part of the information contained in the game is relevant to the agent's happiness. Consider the left-hand example below. Only the information displayed on the right-hand side is necessary in order to deem happiness of agent H .

	h_1	$[h_2]$	h_3		h_1	$[h_2]$	h_3
v_1	0	1	0	0	1	2	
$[v_2]$	2	2	1	0	1	1	

$[v_2]$	$*$	2	$*$	0	$*$	1
---------	-----	---	-----	---	-----	---

The first example below has exactly one Nash equilibrium. The second and third examples have exactly two Nash equilibria each. In the last example, any strategy profile is a Nash equilibrium.

	h_1	h_2
v_1	1	1
v_2	0	1

	h_1	h_2
v_1	1	1
v_2	0	0

	h_1	h_2
v_1	1	0
v_2	0	0

	h_1	h_2
v_1	1	0
v_2	1	0

The left-hand example below shows a game without Nash equilibrium. The right-hand example shows a game where the only Nash equilibrium is sub-optimal in the sense that there exists a profile that grants better payoff for every agent. However, this other profile is not a Nash equilibrium.

	h_1	h_2
v_1	1	0
v_2	0	1

	h_1	h_2
v_1	1	1
v_2	0	3

Note that only finite strategic games have been considered so far. The following one-agent infinite strategic game has no Nash equilibrium either.

h_0	h_1	h_2	\dots	h_i	\dots
0	1	2	\dots	i	\dots

2.2.3 Strict Nash Equilibrium

Strict Nash equilibrium is a well-known refinement of Nash equilibrium. It refers to Nash equilibrium such that any agent would explicitly decrease his payoff when changing his own strategy.

Definition 5 Given a strategic game $g = \langle \mathcal{A}, \mathcal{S}, P \rangle$, strategy s being a strict Nash equilibrium for game g is defined as follows.

$$\text{Eq}_g^+(s) \triangleq \forall a \in \mathcal{A}, s' \in \mathcal{S} - \{s\}, \quad s'_{-a} = s_{-a} \Rightarrow P(s', a) < P(s, a)$$

The three examples below show that the existence of Nash equilibria and the existence of strict Nash equilibria are two orthogonal issues. The left-hand game has only one Nash equilibrium, and it is a strict one. The game in the centre has one strict Nash equilibrium and a non-strict Nash equilibrium. The game to the right has only non-strict Nash equilibria.

	h_1	h_2		h_1	h_2		h_1	h_2
v_1	2	2	1	0		v_1	0	0
v_2	0	1	0	0		v_2	0	0

2.3 Convertibility Preference Game

This section introduces the notion of convertibility preference (CP) game and abstract Nash equilibrium within CP games. An embedding shows that CP games are abstractions of strategic games and that abstract Nash equilibria in CP games are abstractions of Nash equilibria. More specifically, CP games are intended to be abstract and general while still allowing a game theoretic definition of Nash equilibria.

Definition 6 (CP Games) CP games are 4-tuples of the form

$\langle \mathcal{A}, \mathcal{S}, (\overset{+}{\rightarrow}^a)_{a \in \mathcal{A}}, (\overset{\circ}{\rightarrow}^a)_{a \in \mathcal{A}} \rangle$ and such that:

- \mathcal{A} is a non-empty set of agents.
- \mathcal{S} is a non-empty set of synopses.
- For $a \in \mathcal{A}$, $\overset{+}{\rightarrow}^a \subseteq \mathcal{S} \times \mathcal{S}$ says when agent a can convert from a synopsis (the left) to another (the right).
- For $a \in \mathcal{A}$, $\overset{\circ}{\rightarrow}^a \subseteq \mathcal{S} \times \mathcal{S}$ says when agent a prefers a synopsis (the right) to another (the left).

Agents of CP games are interpreted the same way as agents of strategic games: they are "players" taking part in the game. Synopses of CP games are abstractions over strategy profiles of strategic games, while nothing in CP games corresponds to the notion of strategy. The preference relations, one per agent, account for the preference of agents over synopses, in terms of outcome of the game. The notion of real-valued payoff function used in strategic games,

together with the implicit usual total order over the reals, is hereby split between and abstracted by the notions of synopses and preferences. The convertibility relations, one per agent, account for the ability of the agents to unilaterally convert a synopsis into another one. In strategic games, this convertibility was implicitly given by the Cartesian-product structure.

In a CP game, a given agent is happy with a given synopsis if he both *can* not and does not *want* to leave the synopsis. It is defined below.

Definition 7 (Happiness) Let $g = \langle \mathcal{A}, \mathcal{S}, (\overset{+}{\rightarrow}^a)_{a \in \mathcal{A}}, (\overset{\circ}{\rightarrow}^a)_{a \in \mathcal{A}} \rangle$ be a CP game.

$$Happy(a, s) \triangleq \forall s' \in \mathcal{S}, \neg(s \overset{+}{\rightarrow}^a s' \wedge s \overset{\circ}{\rightarrow}^a s')$$

The following defines abstract Nash equilibria as synopses that make all agents happy.

Definition 8 Let $g = \langle \mathcal{A}, \mathcal{S}, (\overset{+}{\rightarrow}^a)_{a \in \mathcal{A}}, (\overset{\circ}{\rightarrow}^a)_{a \in \mathcal{A}} \rangle$ be a CP game. Synopsis s being an abstract Nash equilibrium for game g is defined as follows.

$$Eq_g(s) \triangleq \forall a \in \mathcal{A}, Happy(a, s)$$

There is a natural embedding of strategic games into CP games. This embedding sends Nash equilibria to abstract Nash equilibria and vice versa, so both notions may be referred to as Nash equilibrium.

Lemma 9 Given a strategic game $g = \langle \mathcal{A}, \mathcal{S}, P \rangle$, let

$$\begin{aligned} s \overset{+}{\rightarrow} s' &\triangleq s'_{-a} = s_{-a} \\ s \overset{\circ}{\rightarrow} s' &\triangleq P(s, a) < P(s', a) \end{aligned}$$

$g' \triangleq \langle \mathcal{A}, \mathcal{S}, (\overset{+}{\rightarrow}^a)_{a \in \mathcal{A}}, (\overset{\circ}{\rightarrow}^a)_{a \in \mathcal{A}} \rangle$ is a CP game and

$$Eq_g(s) \Leftrightarrow Eq_{g'}(s)$$

Since there exist strategic games without Nash equilibrium, the embedding shows that there exist CP games without abstract Nash equilibrium.

The intersection of convertibility and preference plays an important role, and it will be helpful to define it. This conjunction of power and will of a single agent leads to an actual conversion of the agent's mindset. It is referred to as the change of mind.

Definition 10 The (free) change-of-mind relation for a is $\overset{a}{\rightarrow} \triangleq (\overset{+}{\rightarrow}^a \cap \overset{\circ}{\rightarrow}^a)$. Let $\overset{\Delta}{\rightarrow} \triangleq \bigcup_{a \in \mathcal{A}} \overset{a}{\rightarrow}$ be the change-of-mind relation.

Below, a synopsis being a (abstract) Nash equilibrium is rephrased as a node being of outdegree zero in the change-of-mind graph.

Lemma 11

$$Eq(s) \Leftrightarrow \forall s' \in \mathcal{S}, \neg(s \overset{\Delta}{\rightarrow} s')$$

Compare the proposition above with definition 8. The main difference is that the order of the two universal quantifications has been inverted. Indeed in definition 8, agents are first universally quantified, while the universal quantification over synopses is hidden within happiness. On contrary in definition 11, synopses are first universally quantified, while the universal quantification over agents is hidden within the change-of-mind relation.

Note that when dealing with (2-player) strategic games, the symbol $+$ may suggest convertibility along either horizontal or vertical axis, and the symbol \bigcirc may suggest possible preferences in any direction. Also note that the symbol \xrightarrow{a} is the graphic intersection of the two symbols $\xrightarrow{+a}$ and $\xrightarrow{\bigcirc a}$. It would also be possible to represent intersection of the two relations by the graphic union of their symbols, i.e. \oplus^a , but \xrightarrow{a} is simpler.

2.4 Best Response Game

This section introduces the notion of best response (BR) game and abstract Nash equilibrium within BR games. It also defines abstract strict Nash equilibria. An embedding shows that BR games are abstractions of strategic games, that abstract Nash equilibria are abstractions of Nash equilibria, and that strict abstract Nash equilibria are abstractions of strict Nash equilibria. More specifically, BR games are intended to be abstract and general while still allowing a game theoretic definition of Nash equilibria and strict Nash equilibria.

Definition 12 (BR Games) *BR games are 3-tuples of the form $\langle \mathcal{A}, \mathcal{S}, (BR_a)_{a \in \mathcal{A}} \rangle$ and such that:*

- \mathcal{A} is a non-empty set of agents.
- \mathcal{S} is a non-empty set of synopses.
- For $a \in \mathcal{A}$, BR_a has type $\mathcal{S} \rightarrow \mathcal{P}(\mathcal{S}) - \{\emptyset\}$, and $s' \in BR_a(s)$ says that s' is one of the best responses of agent a to the synopsis s .

Agents of BR games are interpreted the same way as agents of strategic games: they are "players" taking part in the game. Synopses of BR games are abstractions over strategy profiles of strategic games, while nothing in BR games corresponds to the notion of strategy. The best response functions, one per agent, account for the best reactions of a given agent to a given synopsis, among the reactions of the agent that are effectively possible. Note that every agent has at least one best response to every synopsis. The notion of real-valued payoff function used in strategic games, the implicit usual total order over the reals, and the Cartesian product structure inducing implicit available options are all abstracted by the concept of best response.

In a BR game, a given agent is happy with a synopsis if this synopsis is a best response to itself according to the agent.

Definition 13 (Happiness) *Let $g = \langle \mathcal{A}, \mathcal{S}, (BR_a)_{a \in \mathcal{A}} \rangle$ be a BR game.*

$$Happy(a, s) \stackrel{\Delta}{=} s \in BR_a(s)$$

The following defines abstract Nash equilibria as synopses that make all agents happy.

Definition 14 Let $g = \langle \mathcal{A}, \mathcal{S}, (BR_a)_{a \in \mathcal{A}} \rangle$ be a BR game. Synopsis s being an abstract Nash equilibrium for game g is defined as follows.

$$Eq_g(s) \triangleq \forall a \in \mathcal{A}, \text{Happy}(a, s)$$

In a BR game, a given agent is strictly happy with a synopsis if this synopsis is the only best response to itself according to the agent.

Definition 15 (Strict Happiness) Let $g = \langle \mathcal{A}, \mathcal{S}, (BR_a)_{a \in \mathcal{A}} \rangle$ be a BR game.

$$\text{Happy}^+(a, s) \triangleq \{s\} = BR_a(s)$$

The following defines abstract strict Nash equilibria as synopses that make all agents strictly happy.

Definition 16 Let $g = \langle \mathcal{A}, \mathcal{S}, (BR_a)_{a \in \mathcal{A}} \rangle$ be a BR game. Synopsis s being an abstract strict Nash equilibrium for game g is defined as follows.

$$Eq_g^+(s) \triangleq \forall a \in \mathcal{A}, \text{Happy}^+(a, s)$$

There is a natural embedding of strategic games into BR games. This embedding sends (strict) Nash equilibria to abstract (strict) Nash equilibria and vice versa.

Lemma 17 Given a strategic game $g = \langle \mathcal{A}, \mathcal{S}, P \rangle$, let

$$BR_a(s) \triangleq \{c \in \mathcal{S}_a \mid \forall c' \in \mathcal{S}_a, P(s_{-a}; c', a) \leq P(s_{-a}; c, a)\}$$

$g' \triangleq \langle \mathcal{A}, \mathcal{S}, (BR_a)_{a \in \mathcal{A}} \rangle$ is a BR game and

$$Eq_g(s) \Leftrightarrow Eq_{g'}(s)$$

$$Eq_g^+(s) \Leftrightarrow Eq_{g'}^+(s)$$

Since there exist strategic games without Nash equilibrium, the embedding shows that there exist BR games without abstract Nash equilibrium.

Since an abstract Nash equilibrium is a synopsis that is a best response to itself for all agents, it may be helpful to define a global and-best response function as the intersection of the best response functions. It will be also useful to define a global or-best response function as the union of the best response functions.

Definition 18 (And/or best responses) The and-best response and the or best response functions are defined as follows.

$$\begin{aligned} BR^\cap(s) &\triangleq \bigcap_{a \in \mathcal{A}} BR_a(s) \\ BR^\cup(s) &\triangleq \bigcup_{a \in \mathcal{A}} BR_a(s) \end{aligned}$$

Notice that $BR^\cap(s)$ may be empty for some s , although all $BR_a(s)$ are non-empty by definition. On contrary, $BR^\cup(s)$ is always non-empty. Moreover, the following inclusions always holds.

Lemma 19 *Let $g = \langle \mathcal{A}, \mathcal{S}, (BR_a)_{a \in \mathcal{A}} \rangle$ be a BR game.*

$$BR^\cap(s) \subseteq BR_a(s) \subseteq BR^\cup(s)$$

Below, a synopsis being a (abstract) Nash equilibrium is rephrased as a synopsis that is a global best response to itself.

Lemma 20 *Let $g = \langle \mathcal{A}, \mathcal{S}, (BR_a)_{a \in \mathcal{A}} \rangle$ be a BR game.*

$$Eq_g(s) \Leftrightarrow s \in BR^\cap(s)$$

Compare the proposition above with definition 14. The main difference is that the order of the universal quantification over agents and the and-best response membership of the synopsis has been inverted. Indeed in definition 14, agents are first universally quantified, while the and-best response membership of the synopsis is hidden within happiness. On contrary in lemma 20, the and-best response membership of the synopsis is first stated, while the universal quantification over agents is hidden within the and-best response function.

In the same way, strict Nash equilibria are characterised through the or-best response function. Note that membership of lemma 20 is replaced with equality.

Lemma 21 *Let $g = \langle \mathcal{A}, \mathcal{S}, (BR_a)_{a \in \mathcal{A}} \rangle$ be a BR game.*

$$Eq^+(s) \Leftrightarrow \{s\} = BR^\cup(s)$$

Proof Assume that $Eq^+(s)$. So $\{s\} = BR_a(s)$ for every agent a , by definition of Eq^+ . Therefore $\{s\} = BR^\cup(s)$ by definition of BR^\cup . Conversely, assume that $\{s\} = BR^\cup(s)$, so $BR_a(s) \subseteq \{s\}$ for every agent a , by definition of BR^\cup . But $BR_a(s) \neq \emptyset$ by definition of BR games, so $BR_a(s) = \{s\}$ for every agent a , who is therefore strictly happy with s . Hence $Eq^+(s)$. \square

Chapter 3

Acyclicity and Finite Linear Extendability: a Formal and Constructive Equivalence

Linear extension of partial orders emerged in the late 1920's. Its computer-oriented version, *i.e.*, topological sorting of finite partial orders, arose in the late 1950's. However, those issues have not yet been considered from a viewpoint that is both formal and constructive; this chapter discusses a few related claims formally proved with the constructive proof assistant Coq. For instance, it states that a given decidable binary relation is acyclic and equality is decidable on its domain *iff* an irreflexive linear extension can be computed uniformly for any of its finite restriction. A detailed introduction and proofs written in plain English shall help readers who are not familiar with constructive issues or Coq formalism.

3.1 Introduction

This section adopts an approach technical and historical. It presents the two notions of computability and linear extension, both involved in the results proved in Coq and discussed in this chapter. It also describes the main results and the contents of the chapter.

3.1.1 Decidability and Computability

In the middle of the 1930's, Church introduced the lambda calculus, and Turing and Post independently designed their very similar machines. Those three notions are by some means equivalent models for computer programs. A question is said to be decidable if there exists a computer program (equivalently lambda term or Post-Turing machine) requiring the parameters of the questions as input, returning (within finite time) “yes” or “no” as output, and thus

correctly answering the question. In this way, a binary relation over a set is said to be decidable if there exists a program expecting two elements in that set and returning “yes” if they are related or “no” if they are not. A sister notion is that of computable (or recursive) function, *i.e.*, mathematical function the images of which are computable within finite time by the same computer program although the domain and the codomain of the function may be infinite. Note that computability with two-element-codomain functions amounts to decidability. If several computable functions can be all computed by the same program, then these functions are said to be uniformly computable.

3.1.2 Transitive Closure, Linear Extension, and Topological Sorting

The calculus of binary relations was developed by De Morgan around 1860. The notion of transitive closure of a binary relation (smallest transitive binary relation including a given binary relation) was defined in different manners by different people about 1890. See Pratt [45] for a historical account. In 1930, Szpilrajn [51] proved that, assuming the axiom of choice, any partial order has a linear extension, *i.e.*, is included in some total order. The proof invokes a notion close to transitive closure. Szpilrajn acknowledged that Banach, Kuratowsky, and Tarski had found unpublished proofs of the same result. In the late 1950’s, The US Navy [3] designed PERT (Program Evaluation Research Task or Project Evaluation Review Techniques) for management and scheduling purposes. This tool partly consists in splitting a big project into small jobs on a chart and expressing with arrows when one job has to be done before another one can start up. In order to study the resulting directed graph, Jarnagin [35] introduced a finite and algorithmic version of Szpilrajn’s result. This gave birth to the widely studied topological sorting issue, which spread to the industry in the early 1960’s (see [26] and [21]). Some technical details and computer-oriented examples can be found in Knuth’s book [23].

3.1.3 Contribution

This chapter revisits a few folklore results involving transitive closure, excluded middle, computability, linear extension, and topological sorting. Most of the properties are logical equivalences instead of one-way implications, which suggests maximal generality. Claims have been fully formalized (and proved) in Coq and then slightly modified in order to fit in and be part of the Coq-related CoLoR library [11]. This chapter follows the structure of the underlying Coq development but some straightforward results are omitted. Arguments are constructive (therefore also classical) and usually simple. This chapter is meant to be read by a mathematician who is not familiar with constructivism. Concepts specific to Coq are introduced before they are used. Formal definitions and results are stated in a light Coq formalism that is very close to traditional mathematics and slightly different from the actual Coq code in order to ease the reading. Proofs are mostly written in plain English. The main result in this chapter relies on an intermediate one, and is itself invoked in a game theoretic proof (in Coq) not published yet.

In this chapter, a binary relation over an arbitrary set is said to be middle-

excluding if for any two elements in the set, either they are related or they are not. The intermediate result of this chapter implies that in an arbitrary set with decidable (resp. middle-excluding) equality, a binary relation is decidable (resp. middle-excluding) *iff* the transitive closures of its finite restrictions are uniformly decidable (resp. middle-excluding). The main result splits into two parts, one on excluded middle and one on computability: First, consider a middle-excluding relation. It is acyclic and equality on its domain is middle-excluding *iff* its restriction to any finite set has a middle-excluding irreflexive linear extension. Second, consider R a decidable binary relation over A . The following three propositions are equivalent. Note that computability of linear extensions is non-uniform in the second proposition but uniform in the third one.

- Equality on A is decidable and R is acyclic.
- Equality on A is decidable and every finite restriction of R has a decidable linear extension.
- There exists a computable function that expects finite restrictions of R and returns (decidable) linear extensions of them.

3.1.4 Contents

Section 3.2 gives a quick look at the Coq versions of types, binary relations, excluded middle, and computability. Through the example of lists, section 3.3 explains the principle of definition by induction in Coq, as well as the associated inductive proof principle and definition by recursion. In particular, subsection 3.3.2 details a simple proof by induction on list. Subsection 3.4.1 explains the inductive notion of transitive closure and the associated inductive proof principle. Subsections 3.4.2 and 3.4.3 discuss irreflexivity, representation of “finite sets” by lists, define finite restrictions of a binary relation, and detail a simple proof by induction on transitive closure. Section 3.5 defines paths with respect to a binary relation and proves their correspondence with transitive closure. It also defines bounded paths that are proved to preserve decidability and middle-exclusion properties of the original relation. Since bounded paths and paths are by some means equivalent on finite sets, subsection 3.5.4 states the intermediate result. Subsection 3.6.1 defines relation totality over finite sets. Subsections 3.6.2 to 3.6.5 define an acyclicity-preserving conditional single-arc addition (to a relation), and an acyclicity-preserving multi-stage arc addition over finite sets, which consists in repeating in turn single-arc addition and transitive closure. This procedure helps state linear extension equivalence in 3.6.6 and topological sorting equivalence in 3.6.7.

3.1.5 Convention

Let A be a *Set*. Throughout this chapter x, y, z , and t implicitly refer to objects of type A . In the same way R, R' , and R'' refer to binary relations over A ; l, l' , and l'' to lists over A , and n to natural numbers. For the sake of readability, types will sometimes be omitted according to the above convention, even in formal statements where Coq could not infer them. The notation $\neg P$ stands for $P \rightarrow \text{False}$, $x \neq y$ for $x = y \rightarrow \text{False}$, and $\nexists x, P$ for $(\exists x, P) \rightarrow \text{False}$.

3.2 Preliminaries

3.2.1 Types and Relations

Any Coq object has a type, which informs of the usage of the object and its possible interactions with other Coq objects. The Coq syntax $Obj : T$ means that Obj has type T . For example, $f : A \rightarrow B$ means that f requires an argument in the domain A and returns an output in the codomain B . If x has type A then f and x can be combined and yield $(f\ x)$, also written $f(x)$ or $f\ x$, of type B . A type is also a Coq object so it has a type too. The only types of types mentioned in this chapter are *Prop* and *Set*. The two propositions *True* and *False* are in *Prop* but in a constructive setting there are propositions, *i.e.*, objects in *Prop*, neither equivalent to *True* nor to *False*. Both the collection of all natural numbers and the collection of the two booleans *true* and *false* have type *Set*. Intuitively, proving propositions in *Prop* amounts to traditional (and intuitionistic) mathematical reasoning as proving objects in *Set* is computationally stronger since effective programs can be extracted from theorems in *Set*. Now consider $g : A \rightarrow (B \rightarrow C)$ where the parentheses are usually omitted by convention. The function g expects an argument in A and returns a function expecting an argument in B and returning an output in C . Therefore g can be seen as a function requiring a first argument in A , a second one in B , and returning an object in C . Binary relations over A can be represented by functions typed in $A \rightarrow A \rightarrow Prop$, *i.e.* requiring two arguments in A and returning a proposition (that may be interpreted as “the two arguments are related”). The returned proposition may be *True*, *False*, or something else that may or may not be equivalent to either *True* or *False*. For example if it returns always something absurd, *i.e.*, implying *False*, then it is “the” empty relation over A . The object *Identity_relation*, defined below in the light Coq formalism using this chapter’s convention, can be interpreted as the identity relation over A . Indeed, it requires two arguments in A and returns a proposition asserting that those arguments are equal.

Definition *Identity_relation* $x\ y : Prop := x=y$.

The actual Coq code would need to make it clear that x and y are in A .

Definition *Identity_relation* $(x\ y : A) : Prop := x=y$.

3.2.2 Excluded Middle and Decidability

The following two objects define middle-excluding equality on A and middle-excluding binary relations over A , respectively.

Definition *eq_midex* $:= \forall x\ y, x=y \vee x \neq y$.

Definition *rel_midex* $R := \forall x\ y, R\ x\ y \vee \neg R\ x\ y$.

Note that in general the proposition *eq_midex* is only a definition but not a theorem in Coq, *i.e.*, there is no proof of which the conclusion is the proposition $\forall x\ y, x=y \vee x \neq y$. Same remark for *rel_midex*.

This chapter widely uses the syntax $\forall v, \{B\} + \{C\}$ which is a $\forall v, B \vee C$ with a computational content. It means that for all v either B holds or C holds, and that, in addition, there exists a computable function expecting a v and pointing to one that holds. The next two definitions respectively say that equality on A is decidable and that a given binary relation over A is decidable.

Definition $eq_dec := \forall x y, \{x=y\} + \{x \neq y\}$.

Definition $rel_dec R := \forall x y, \{R x y\} + \{\neg R x y\}$.

The remainder of this subsection 3.2.2 justifies further the syntax $\forall v, \{B\} + \{C\}$ as correct representation for decidability. Roughly speaking, $\{Obj : T \mid P\}$ means the existence, with computational content, of an object of type T satisfying the predicate P , whereas $\exists Obj : T, P$ does the same without computational content. As shown by the two lemmas below, $rel_dec R$ is equivalent to “computable” existence of a function requiring two arguments in A and returning the boolean *true* if the two arguments are related or *false* if they are not, which amounts to decidability as discussed in subsection 3.1.1.

Lemma $rel_dec_bool : \forall R,$

$rel_dec R \rightarrow \{f : A \rightarrow A \rightarrow bool \mid \forall x y : A, \text{if } f x y \text{ then } R x y \text{ else } \neg R x y\}.$

Lemma $bool_rel_dec : \forall R,$

$\{f : A \rightarrow A \rightarrow bool \mid \forall x y : A, \text{if } f x y \text{ then } R x y \text{ else } \neg R x y\} \rightarrow rel_dec R.$

Therefore the syntax $\{P\} + \{\neg P\}$ is usually a convenient way to represent decidability in Coq. In terms of usage: while proving a proposition, both $x=y \vee x \neq y$ and $\{x=y\} + \{x \neq y\}$ allow case splitting (first case $x=y$ and second case $x \neq y$). When building a function, however, only $\{x=y\} + \{x \neq y\}$ allows case splitting, e.g., to perform a traditional “if-then-else”. It means that the computational content of $\{x=y\} + \{x \neq y\}$ is stronger, as suggested below. The same kind of result is available for binary relations.

Lemma $eq_dec_midex : eq_dec \rightarrow eq_midex.$

Lemma $rel_dec_midex : rel_dec \rightarrow rel_midex.$

3.3 On Lists

3.3.1 Lists in the Coq Standard Library

All the underlying Coq code of this subsection can be found in the Coq Standard Library [1]. Semi-formally, lists are defined by *induction* as follows. Let B be any set. Consider all L complying with the following two conditions:

- nil is in L and is called the empty list over B .
- If x is in B and l is in L then $(cons x l)$ is in L .

The lists over B are defined as the (unique) least such an L . Formally in Coq, lists over A can be defined as follows:

Inductive $list : Set :=$

| $nil : list$

| $cons : A \rightarrow list \rightarrow list.$

The first line defines lists over A by induction and considers the collection of all lists over A as a *Set*. The next line says that nil is a list over A and the last line says that giving an element of A and list over A as arguments to the constructor $cons$ yields a new list over A . The notation $x::l$ stands for $cons x l$. Call x the head of $x::l$ and l its tail.

Example 22 If x and y are in A then $\text{cons } x (\text{cons } y \text{ nil})$ is a list over A represented by $x::y::\text{nil}$.

In Coq, any definition by induction comes with an inductive proof principle, aka induction principle. For lists, it states that if a predicate holds for the empty list and is preserved by list construction, then it holds for all lists. Formally:

$$\forall (P : \text{list} \rightarrow \text{Prop}), P \text{ nil} \rightarrow (\forall a \, l, P \, l \rightarrow P (a :: l)) \rightarrow \forall l, P \, l.$$

There is a way to define functions over inductively defined objects, just along the inductive definition of those objects. This is called a definition by *recursion*. For example, appending two lists l and l' is defined by recursion on the first list argument l .

Fixpoint $\text{app } l \, l' \{ \text{struct } l \} : \text{list} :=$
 $\text{match } l \text{ with}$
 $\mid \text{nil} \Rightarrow l'$
 $\mid x :: l'' \Rightarrow x :: \text{app } l'' \, l'$
 $\text{end}.$

The function app requires two lists over A and returns a list over A . The command **Fixpoint** means that app is defined by recursion and $\{ \text{struct } l \}$ means that the recursion involves l , the first list argument. As lists are built using two constructors, the body of the function case splits on the structure of the list argument. If it is nil then the appending function returns the second list argument. If not then the appending function returns a list involving the same appending function with a strictly smaller first list argument, which ensures process termination. The notation $l++l'$ stands for $\text{app } l \, l'$.

Example 23 Let x, y, z , and t be in A . Computation steps are shown below.
 $(x::y::\text{nil})++(z::t::\text{nil}) \rightsquigarrow x::((y::\text{nil})++(z::t::\text{nil})) \rightsquigarrow$
 $x::y::(\text{nil}++(z::t::\text{nil})) \rightsquigarrow x::y::z::t::\text{nil}.$

The function length of a list is defined by recursion, as well as the predicate In saying that a given element occurs in a given list. The predicate incl says that all the elements of a first list occur in a second list. It is defined by recursion on the first list, using In .

3.3.2 Decomposition of a List

If equality is middle-excluding on A and if an element occurs in a list built over A , then the list can be decomposed into three parts: a list, one occurrence of the element, and a second list where the element does not occur.

Lemma $\text{In_elim_right} : \text{eq_midex} \rightarrow \forall x \, l,$
 $\text{In } x \, l \rightarrow \exists l', \exists l'', l = l' ++ (x :: l'') \wedge \neg \text{In } x \, l''.$

Proof Assume that equality is middle-excluding on A and let x be in A . Next, prove by induction on l the proposition $\forall l, \text{In } x \, l \rightarrow \exists l', \exists l'', l = l' ++ (x :: l'') \wedge \neg \text{In } x \, l''$. The base case, $l = \text{nil}$, is straightforward since x cannot occur in the empty list. For the inductive case, $l = y :: l1$, the induction hypothesis is $\text{In } x \, l1 \rightarrow \exists l', \exists l'', l1 = l' ++ (x :: l'') \wedge \neg \text{In } x \, l''$. Assume that x occurs in l and prove $\exists l', \exists l'', y :: l1 = l' ++ (x :: l'') \wedge \neg \text{In } x \, l''$ as follows: case split on x occurring in $l1$. If x occurs

in $l1$ then get l' and l'' from the induction hypothesis, and show that $y::l'$ and l'' are witnesses. If x does not occur in $l1$ then x equals y , so nil and $l1$ are witnesses. \square

3.3.3 Repeat-Free Lists

The predicate *repeat_free* says that no element occurs more than once in a given list. It is defined by recursion on its sole argument.

```
Fixpoint repeat_free l : Prop :=
  match l with
  | nil  $\Rightarrow$  True
  | x::l'  $\Rightarrow$   $\neg$ In x l'  $\wedge$  repeat_free l'
  end.
```

If equality is middle-excluding on A then a *repeat_free* list included in another list is not longer than the other list. This is proved by induction on the *repeat_free* list. For the inductive step, invoke *In_elim_right* to decompose the other list along the head of the *repeat_free* list.

Lemma *repeat_free_incl_length* : *eq_midex* $\rightarrow \forall l l'$,
repeat_free $l \rightarrow incl\ l\ l' \rightarrow length\ l \leq length\ l'$.

3.4 On Relations

3.4.1 Transitive Closure in the Coq Standard Library

Traditionally, the transitive closure of a binary relation is *the* smallest transitive binary relation including the original relation. The notion of transitive closure can be formally defined by induction, in the Coq Standard Library. The following function *clos_trans* expects a relation over A and yields its transitive closure, which is also a relation over A .

```
Inductive clos_trans R : A  $\rightarrow$  A  $\rightarrow$  Prop :=
  | t_step :  $\forall x\ y, R\ x\ y \rightarrow clos\_trans\ R\ x\ y$ 
  | t_trans :
     $\forall x\ y\ z, clos\_trans\ R\ x\ y \rightarrow clos\_trans\ R\ y\ z \rightarrow clos\_trans\ R\ x\ z.$ 
```

Informally, the *t_step* constructor guarantees that *clos_trans* R contains R and the *t_trans* constructor adds all “arcs” the absence of which would contradict transitivity.

Intuitively, two elements are related by the transitive closure of a binary relation if one can start at the first element and reach the second one in finitely many steps of the original relation. Therefore replacing *clos_trans* $R\ x\ y \rightarrow clos_trans\ R\ y\ z \rightarrow clos_trans\ R\ x\ z$ by $R\ x\ y \rightarrow clos_trans\ R\ y\ z \rightarrow clos_trans\ R\ x\ z$ or *clos_trans* $R\ x\ y \rightarrow R\ y\ z \rightarrow clos_trans\ R\ x\ z$ in the definition of *clos_trans* would yield two relations coinciding with *clos_trans*. Those three relations are yet different in intension: only *clos_trans* captures the meaning of the terminology “transitive closure”.

The Coq Standard Library also defines what a *transitive* relation is. In addition, this chapter needs the notion of subrelation.

Definition $sub_rel\ R\ R' : Prop := \forall x\ y, R\ x\ y \rightarrow R'\ x\ y$.

The notion of subrelation helps express the induction principle for $clos_trans$. It states that if a relation contains R and satisfies the following “weak transitivity” property then it also contains $clos_trans\ R$.

$$\begin{aligned} &\forall R', sub_rel\ R\ R' \rightarrow \\ &(\forall x\ y\ z, clos_trans\ R\ x\ y \rightarrow R'\ x\ y \rightarrow clos_trans\ R\ y\ z \rightarrow R'\ y\ z \rightarrow R'\ x\ z) \rightarrow sub_rel \\ &(clos_trans\ R)\ R' \end{aligned}$$

The next lemma asserts that a transitive relation contains its own transitive closure (they actually coincide).

Lemma $transitive_sub_rel_clos_trans : \forall R,$
 $transitive\ R \rightarrow sub_rel\ (clos_trans\ R)\ R$.

Proof Let R be a transitive relation over A . Prove the subrelation property by the induction principle of $clos_trans$. The base case is trivial and the inductive case is derived from the transitivity of R . \square

3.4.2 Irreflexivity

A relation is irreflexive if no element is related to itself. Therefore irreflexivity of a relation implies irreflexivity of any subrelation.

Definition $irreflexive\ R : Prop := \forall x, \neg R\ x\ x$.

Lemma $irreflexive_preserved : \forall R\ R',$
 $sub_rel\ R\ R' \rightarrow irreflexive\ R' \rightarrow irreflexive\ R$.

3.4.3 Restrictions

Throughout this chapter, finite “subsets” of A are represented by lists over A . For that specific use of lists, the number and the order of occurrences of elements in a list are irrelevant. Let R be a binary relation over A and l be a list over A . The binary relation $restriction\ R\ l$ relates elements that are both occurring in l and related by R . The predicate $is_restricted$ says that “the support of the given binary relation R is included in the list l ”. And the next lemma shows that transitive closure preserves restriction to a given finite set.

Definition $restriction\ R\ l\ x\ y : Prop := In\ x\ l \wedge In\ y\ l \wedge R\ x\ y$.

Definition $is_restricted\ R\ l : Prop := \forall x\ y, R\ x\ y \rightarrow In\ x\ l \wedge In\ y\ l$.

Lemma $restricted_clos_trans : \forall R\ l,$
 $is_restricted\ R\ l \rightarrow is_restricted\ (clos_trans\ R)\ l$.

Proof Assume that R is restricted to l . Let x and y in A be such that $clos_trans\ R\ x\ y$, and prove by induction on that last hypothesis that x and y are in l . The base case, where “ $clos_trans\ R\ x\ y$ comes from $R\ x\ y$ ”, follows by definition of restriction. For the inductive case, where “ $clos_trans\ R\ x\ y$ comes from $clos_trans\ R\ x\ z$ and $clos_trans\ R\ z\ y$ for some z in A ”, induction hypotheses are $In\ x\ l \wedge In\ z\ l$ and $In\ z\ l \wedge In\ y\ l$, which allows concluding. \square

If the support of a relation involves only two (possibly equal) elements, and if those two elements are related by the transitive closure, then they are also related by the original relation. By the induction principle for *clos_trans* and lemma *restricted_clos_trans*.

Lemma *clos_trans_restricted_pair* : $\forall R \ x \ y,$
is_restricted $R \ (x::y::nil) \rightarrow \text{clos_trans } R \ x \ y \rightarrow R \ x \ y.$

3.5 On Paths and Transitive Closure

3.5.1 Paths

The notion of path relates to one interpretation of transitive closure. Informally, a path is a list recording consecutive steps of a given relation. The following predicate says that a given list is a path between two given elements with respect to a given relation.

Fixpoint *is_path* $R \ x \ y \ l \ \{\text{struct } l\} : \text{Prop} :=$
match l *with*
 $\mid \text{nil} \Rightarrow R \ x \ y$
 $\mid z::l' \Rightarrow R \ x \ z \wedge \text{is_path } R \ z \ y \ l'$
end.

The following two lemmas show the correspondence between paths and transitive closure. The first is proved by the induction principle of *clos_trans* and an appending property on paths proved by induction on lists. For the second, let y be in A and prove $\forall l \ x, \text{is_path } R \ x \ y \ l \rightarrow \text{clos_trans } R \ x \ y$ by induction on l . Now consider the variable appearance order $\forall y \ l \ x$ in this lemma. Changing the order would yield a correct lemma as well, but the proof would be less workable. Indeed y can be fixed once for all but x needs to be universally quantified in the induction hypothesis, so x must appear after l on which the induction is performed. Also note that the two lemmas imply $\forall x \ y, \text{clos_trans } R \ x \ y \leftrightarrow \exists l, \text{is_path } R \ x \ y \ l.$

Lemma *clos_trans_path* : $\forall x \ y, \text{clos_trans } R \ x \ y \rightarrow \exists l, \text{is_path } R \ x \ y \ l.$

Lemma *path_clos_trans* : $\forall y \ l \ x, \text{is_path } R \ x \ y \ l \rightarrow \text{clos_trans } R \ x \ y.$

Assume that equality is middle-excluding on A and consider a path between two points. Between those two points there is a *repeat_free* path avoiding them and (point-wise) included in the first path. The inclusion is also arc-wise by construction, but it is not needed in this chapter.

Lemma *path_repeat_free_length* : $\text{eq_midex} \rightarrow \forall y \ l \ x,$
 $\text{is_path } R \ x \ y \ l \rightarrow$
 $\exists l', \neg \text{In } x \ l' \wedge \neg \text{In } y \ l' \wedge \text{repeat_free } l' \wedge$
 $\text{length } l' \leq \text{length } l \wedge \text{incl } l' \ l \wedge \text{is_path } R \ x \ y \ l'.$

Proof Assume that equality is middle-excluding on A , let y be in A , and perform an induction on l . For the inductive step, call a the head of l . If a equals y then the empty list is a witness for the existential quantifier. Now assume that a and y are distinct. Use the induction hypothesis with a and get a list l' . Case split on x occurring in l' . If x occurs in l' then invoke lemma *In_elim_right*

and decompose l' along x , and get two lists. In order to prove that the second list, where x does not occur, is a witness for the existential quantifier, notice that splitting a path yields two paths (*a priori* between different elements) and that appending reflects the *repeat_free* predicate (if the appending of two lists is *repeat_free* then the original lists also are). Next, assume that x does not occur in l' . If x equals a then l' is a witness for the existential quantifier. If x and a are distinct then $a::l'$ is a witness. \square

3.5.2 Bounded Paths

Given a relation and a natural number, the function *bounded_path* returns a relation saying that there exists a path of length at most the given natural number between two given elements.

Inductive *bounded_path* $R n : A \rightarrow A \rightarrow \text{Prop} :=$
 $\mid \text{bp_intro} : \forall x y l, \text{length } l \leq n \rightarrow \text{is_path } R x y l \rightarrow \text{bounded_path } R n x y.$

Below, two lemmas relate *bounded_path* and *clos_trans*. The first lemma is derived from *path_clos_trans* and the second one is derived from *clos_trans_path*, *path_repeat_free_length*, *repeat_free_incl_length*, and a path of a restricted relation being included in the support of the relation. Especially, the second lemma says that in order to know whether two elements are related by the transitive closure of a restricted relation, it suffices to check whether there is, between those two elements, a path of length at most the “cardinal” of the support of the relation.

Lemma *bounded_path_clos_trans* : $\forall R n,$
 $\text{sub_rel } (\text{bounded_path } R n) (\text{clos_trans } R).$

Lemma *clos_trans_bounded_path* : $\text{eq_midex} \rightarrow \forall R l,$
 $\text{is_restricted } R l \rightarrow \text{sub_rel } (\text{clos_trans } R) (\text{bounded_path } R (\text{length } l)).$

3.5.3 Restriction, Decidability, and Transitive Closure

The following lemma says that it is decidable whether or not one step of a given decidable relation from a given starting point to some point z in a given finite set and one step of another given decidable relation from the same point z can lead to another given ending point. Moreover such an intermediate point z is computable when it exists, hence the syntax $\{z : A \mid \dots\}$.

Lemma *dec_lem* : $\forall R' R'' x y l, \text{rel_dec } R' \rightarrow \text{rel_dec } R'' \rightarrow$
 $\{z : A \mid \text{In } z l \wedge R' x z \wedge R'' z y\} + \{\neg \exists z : A, \text{In } z l \wedge R' x z \wedge R'' z y\}.$

The following lemma is the middle-excluding version of the previous lemma.

Lemma *midex_lem* : $\forall R' R'' x y l, \text{rel_midex } R' \rightarrow \text{rel_midex } R'' \rightarrow$
 $(\exists z : A, \text{In } z l \wedge R' x z \wedge R'' z y) \vee (\neg \exists z : A, \text{In } z l \wedge R' x z \wedge R'' z y).$

Proof By induction on l . For the inductive step, call a the head of l . Then case split on the induction hypothesis. In the case of existence, any witness for the induction hypothesis is also a witness for the wanted property. In the case of non-existence, case split on $R' x a$ and $R'' a y$. \square

By unfolding the definition *rel_midex*, the next result implies that given a restricted and middle-excluding relation, a given natural number and two given points, either there is a path of length at most that number between those points or there is no such path. Replacing *midex* by *dec* in the lemma yields a correct lemma about decidability.

Lemma *bounded_path_midex* : $\forall R\ l\ n,$
 $is_restricted\ R\ l \rightarrow rel_midex\ R \rightarrow rel_midex\ (bounded_path\ R\ n).$

Proof First prove three simple lemmas relating *bounded_path*, *n*, and *S n*. Then let *R* be a middle-excluding relation restricted to *l* and *x* and *y* be in *A*. Perform an induction on *n*. For the inductive step, case split on the induction hypothesis with *x* and *y*. If *bounded_path R n x y* holds then it is straightforward. If its negation holds then case split on *em_lem* with *R*, *bounded_path R n*, *x*, *y*, and *l*. In the existence case, just notice that a path of length less than *n* is of length less than *S n*. In the non-existence case, show the negation of *bounded_path* in the wanted property. \square

Let equality and a restricted relation be middle-excluding over *A*, then the transitive closure of the relation is also middle-excluding. The proof invokes *bounded_path_midex*, *bounded_path_clos_trans*, and *clos_trans_bounded_path*. The decidability version of it is also correct.

Lemma *restricted_midex_clos_trans_midex* : $eq_midex \rightarrow \forall R\ l,$
 $rel_midex\ R \rightarrow is_restricted\ R\ l \rightarrow rel_midex\ (clos_trans\ R).$

3.5.4 Intermediate Results

The following theorems state the equivalence between decidability of a relation and uniform decidability of the transitive closures of its finite restrictions. The first result invokes *clos_trans_restricted_pair* and the second implication uses *restricted_dec_clos_trans_dec*. Note that decidable equality is required only for the second implication. These results remain correct when considering excluded middle instead of decidability.

Theorem *clos_trans_restriction_dec_R_dec* : $\forall R$
 $(\forall l, rel_dec\ (clos_trans\ (restriction\ R\ l))) \rightarrow rel_dec\ R.$

Theorem *R_dec_clos_trans_restriction_dec* : $eq_dec \rightarrow \forall R$
 $rel_dec\ R \rightarrow \forall l, rel_dec\ (clos_trans\ (restriction\ R\ l)).$

3.6 Linear Extension and Topological Sorting

Consider *R* a binary relation over *A* and *l* a list over *A*. This section presents a way of preserving acyclicity of *R* while “adding arcs” to the restriction of *R* to *l* in order to build a total and transitive relation over *l*. In particular, if *R* is acyclic, then its image by the relation completion procedure must be a strict total order. The basic idea is to compute the transitive closure of the restriction of *R* to *l*, add an arc *iff* it can be done without creating any cycle, taking the transitive closure, adding an arc if possible, etc. All those steps preserve existing arcs. Since *l* is finite, there are finitely many eligible arcs,

which ensures termination of the process. This is not the fastest topological sort algorithm but its fairly simple expression leads to a simple proof of correctness.

3.6.1 Total

R is said to be total on l if any two distinct elements in l are related either way. Such a trichotomy property for a relation implies trichotomy for any bigger relation.

Definition *trichotomy* $R\ x\ y : Prop := R\ x\ y \vee x=y \vee R\ y\ x$.

Definition *total* $R\ l : Prop := \forall\ x\ y, In\ x\ l \rightarrow In\ y\ l \rightarrow trichotomy\ R\ x\ y$.

Lemma *trichotomy_preserved* : $\forall\ R\ R'\ x\ y,$
 $sub_rel\ R\ R' \rightarrow trichotomy\ R\ x\ y \rightarrow trichotomy\ R'\ x\ y$.

3.6.2 Try Add Arc

If x and y are equal or related either way then define the relation *try_add_arc* $R\ x\ y$ as R , otherwise define it as the disjoint union of R and the arc (x,y) .

Inductive *try_add_arc* $R\ x\ y : A \rightarrow A \rightarrow Prop :=$
 $| keep : \forall\ z\ t, R\ z\ t \rightarrow try_add_arc\ R\ x\ y\ z\ t$
 $| try_add : x \neq y \rightarrow \neg R\ y\ x \rightarrow try_add_arc\ R\ x\ y\ x\ y$.

Prove by induction on l and a few case splittings that, under some conditions, a path with respect to an image of *try_add_arc* is also a path with respect to the original relation.

Lemma *path_try_add_arc_path* : $\forall\ R\ t\ x\ y\ l\ z,$
 $\neg(x=z \vee In\ x\ l) \vee \neg(y=t \vee In\ y\ l) \rightarrow$
 $is_path\ R\ (try_add_arc\ R\ x\ y)\ z\ t\ l \rightarrow is_path\ R\ z\ t\ l$.

The next three lemmas lead to the conclusion that the function *try_add_arc* does not create cycles. The first one is derived from a few case splittings and the last one highly relies on the second one but also invokes *clos_trans_path*.

Lemma *trans_try_add_arc_sym* : $\forall\ R\ x\ y\ z\ t,$
 $transitive\ R \rightarrow try_add_arc\ R\ x\ y\ z\ t \rightarrow try_add_arc\ R\ x\ y\ t\ z \rightarrow R\ z\ z$.

Lemma *trans_bounded_path_try_add_arc* : $eq_midex \rightarrow \forall\ R\ x\ y\ z\ n,$
 $transitive\ R \rightarrow bounded_path\ (try_add_arc\ R\ x\ y)\ n\ z\ z \rightarrow R\ z\ z$.

Proof By induction on n . The base case requires only *trans_try_add_arc_sym*. For the inductive case, consider a path of length less than or equal to $n+1$ and build one of length less than $n+1$ as follows. By *path_repeat_free_length* the path may be *repeat_free*, i.e., without circuit. Proceed by case splitting on the construction of the path: when the path is *nil*, it is straightforward. If the length of the path is one then invoke *sub_rel_try_add_arc trans_try_add_arc_sym*; else perform a 4-case splitting (induced by the disjunctive definition of *try_add_arc*) on the first two (*try_add_arc* $R\ x\ y$)-steps of the path. Two cases out of the four need lemmas *transitive_sub_rel_clos_trans*, *path_clos_trans*, and *path_try_add_arc_path*. \square

Lemma $try_add_arc_irrefl : eq_midex \rightarrow \forall R x y,$
 $transitive R \rightarrow irreflexive R \rightarrow irreflexive (clos_trans (try_add_arc R x y)).$

3.6.3 Try Add Arc (One to Many)

The function $try_add_arc_one_to_many$ recursively tries to (by preserving acyclicity) add all arcs starting at a given point and ending in a given list.

Fixpoint $try_add_arc_one_to_many R x l \{struct l\} : A \rightarrow A \rightarrow Prop :=$
 $match l with$
 $| nil \Rightarrow R$
 $| y::l' \Rightarrow clos_trans (try_add_arc (try_add_arc_one_to_many R x l') x y)$
 $end.$

The following three lemmas prove preservation properties about the function $try_add_arc_one_to_many$: namely, arc preservation, restriction preservation, and middle-exclusion preservation. Decidability preservation is also correct, although not formally stated here.

Lemma $sub_rel_try_add_arc_one_to_many : \forall R x l,$
 $sub_rel R (try_add_arc_one_to_many R x l).$

Proof By induction on l . For the inductive step, call a the head of l and l' its tail. Use transitivity of sub_rel with $try_add_arc_one_to_many x l'$ and $try_add_arc (try_add_arc_one_to_many x l') x a$. Also invoke $clos_trans$ and a similar arc preservation property for try_add_arc . \square

Lemma $restricted_try_add_arc_one_to_many : \forall R l x l', In x l \rightarrow incl l' l \rightarrow$
 $is_restricted R l \rightarrow is_restricted (try_add_arc_one_to_many R x l') l.$

Proof By induction on l' , $restricted_clos_trans$, and a similar restriction preservation property for try_add_arc . \square

Lemma $try_add_arc_one_to_many_midex :$
 $eq_midex \rightarrow \forall R x l l', In x l \rightarrow incl l' l \rightarrow is_restricted R l \rightarrow$
 $rel_midex R \rightarrow rel_midex (try_add_arc_one_to_many R x l').$

Proof By induction on l' , invoking $restricted_try_add_arc_one_to_many$, lemma $restricted_midex_clos_trans_midex$ with l , and a similar middle-exclusion preservation property for try_add_arc . \square

Next, a step towards totality.

Lemma $try_add_arc_one_to_many_trichotomy : eq_midex \rightarrow \forall R x y l l',$
 $In y l' \rightarrow In x l \rightarrow incl l' l \rightarrow is_restricted R l \rightarrow rel_midex R \rightarrow$
 $trichotomy (try_add_arc_one_to_many R x l') x y.$

Proof By induction on l' . For the inductive step, invoke $trichotomy_preserved$, case split on y being the head of l' or y occurring in the tail of l' . Also refer to a similar trichotomy property for try_add_arc . \square

3.6.4 Try Add Arc (Many to Many)

The function $try_add_arc_many_to_many$ requires a relation and two lists. Then, using $try_add_arc_one_to_many$, it recursively tries to safely add all arcs starting in first list argument and ending in the second one.

Fixpoint *try_add_arc_many_to_many* $R\ l\ l' : \{ \text{struct } l' \} : A \rightarrow A \rightarrow \text{Prop} :=$
match l' *with*
 | *nil* $\Rightarrow R$
 | $x::l'' \Rightarrow \text{try_add_arc_one_to_many } (\text{try_add_arc_many_to_many } R\ l''\ l) \ x\ l$
end.

The following three results proved by induction on the list l' state arc, restriction, and decidability preservation properties of *try_add_arc_many_to_many*. For the inductive case of the first lemma, call l'' the tail of l' , apply the transitivity of *sub_rel* with $(\text{try_add_arc_many_to_many } R\ l''\ l)$, and invoke lemma *sub_rel_try_add_arc_one_to_many*. Use *restricted_try_add_arc_one_to_many* for the second lemma. For the third one invoke *try_add_arc_one_to_many_dec* and also *restricted_try_add_arc_many_to_many*. Middle-exclusion preservation is also correct, although not formally stated here.

Lemma *sub_rel_try_add_arc_many_to_many* : $\forall R\ l\ l',$
sub_rel $R\ (\text{try_add_arc_many_to_many } R\ l'\ l).$

Lemma *restricted_try_add_arc_many_to_many* : $\forall R\ l\ l', \text{incl } l'\ l \rightarrow$
is_restricted $R\ l \rightarrow \text{is_restricted } (\text{try_add_arc_many_to_many } R\ l'\ l)\ l.$

Lemma *try_add_arc_many_to_many_dec* : $\rightarrow \forall R\ l\ l', \text{incl } l'\ l \rightarrow$
is_restricted $R\ l \rightarrow \text{rel_dec } R \rightarrow \text{rel_dec } (\text{try_add_arc_many_to_many } R\ l'\ l).$

The next two results state a trichotomy property and also that the function *try_add_arc_many_to_many* does not create any cycle.

Lemma *try_add_arc_many_to_many_trichotomy* : *eq_midex* $\rightarrow \forall R\ l\ x\ y\ l',$
incl $l'\ l \rightarrow \text{In } y\ l \rightarrow \text{In } x\ l' \rightarrow \text{restricted } R\ l \rightarrow \text{rel_midex } R \rightarrow$
trichotomy $(\text{try_add_arc_many_to_many } R\ l'\ l)\ x\ y.$

Proof By induction on l' . Start the inductive step by case splitting on x being the head of l' or occurring in its tail l'' . Conclude the first case by invoking *try_add_arc_one_to_many_trichotomy*, *try_add_arc_many_to_many_midex*, and *restricted_try_add_arc_many_to_many*. Use *trichotomy_preserved*, the induction hypothesis, and *sub_rel_try_add_arc_one_to_many* for the second case. \square

Lemma *try_add_arc_many_to_many_irrefl* : *eq_midex* $\rightarrow \forall R\ l\ l',$
incl $l'\ l \rightarrow \text{is_restricted } R\ l \rightarrow \text{transitive } A\ R \rightarrow$
irreflexive $R \rightarrow \text{irreflexive } (\text{try_add_arc_many_to_many } R\ l'\ l).$

Proof By induction on l' . For the inductive step, first prove a similar irreflexivity property for *try_add_arc_one_to_many* by induction on lists and by *try_add_arc_irrefl*. Then invoke *restricted_try_add_arc_many_to_many*. Both this proof and the one for *try_add_arc_one_to_many* also require transitivity of the transitive closure and an additional case splitting on l' being *nil* or not. \square

3.6.5 Linear Extension/Topological Sort Function

Consider the restriction of a given relation to a given list. The following function tries to add all arcs both starting and ending in that list to that restriction while still preserving acyclicity.

Definition *LETS* $R\ l : A \rightarrow A \rightarrow \text{Prop} :=$

try_add_arc_many_to_many (*clos_trans* (*restriction* *R l*)) *l l*.

The next three lemmas are proved by *sub_rel_try_add_arc_many_to_many*, *transitive_clos_trans*, and *restricted_try_add_arc_many_to_many* respectively.

Lemma *LETS_sub_rel* : $\forall R l,$
sub_rel (*clos_trans* (*restriction* *R l*)) (*LETS* *R l*).

Lemma *LETS_transitive* : $\forall R l,$ *transitive* (*LETS* *R l*).

Lemma *LETS_restricted* : $\forall R l,$ *is_restricted* (*LETS* *R l*) *l*.

Under middle-excluding equality, the finite restriction of *R* to *l* has no cycle iff *LETS* *R l* is irreflexive. Prove left to right by *try_add_arc_many_to_many_irrefl*, and right to left by *irreflexive_preserved* and *LETS_sub_rel*.

Lemma *LETS_irrefl* : *eq_midex* $\rightarrow \forall R l,$
irreflexive (*clos_trans* (*restriction* *R l*)) \leftrightarrow *irreflexive* (*LETS* *R l*).

If *R* and equality on *A* are middle-excluding then *LETS* *R l* is total on *l*. This is proved by lemmas *R_midex_clos_trans_restriction_midex* (in 3.5.4) and *try_add_arc_many_to_many_trichotomy*.

Lemma *LETS_total* : *eq_midex* $\rightarrow \forall R l,$ *rel_midex* *R* \rightarrow *total* (*LETS* *R l*) *l*.

The next two lemmas show that if *R* and equality on *A* are middle-excluding (resp. decidable) then so is *LETS* *R l*: by *try_add_arc_many_to_many_midex* (resp. *try_add_arc_many_to_many_dec*) and *R_midex_clos_trans_restriction_midex* (resp. *R_dec_clos_trans_restriction_dec*).

Lemma *LETS_midex* : *eq_midex* $\rightarrow \forall R l,$
rel_midex *R* \rightarrow *rel_midex* (*LETS* *R l*).

Lemma *LETS_dec* : *eq_dec* $\rightarrow \forall R,$ *rel_dec* *R* $\rightarrow \forall l,$ *rel_dec* (*LETS* *R l*).

3.6.6 Linear Extension

Traditionally, a linear extension of a partial order is a total order including the partial order. Below, a linear extension (over a list) of a binary relation is a strict total order (over the list) that is bigger than the original relation (restricted to the list).

Definition *linear_extension* *R l R'* := *is_restricted* *R' l* \wedge
sub_rel (*restriction* *R l*) *R'* \wedge *transitive* *A R'* \wedge *irreflexive* *R'* \wedge *total* *R' l*.

The next two lemmas say that a relation “locally” contained in some acyclic relation is “globally” acyclic and that if for any list over *A* there is a middle-excluding total order over that list, then equality is middle-excluding on *A*.

Lemma *local_global_acyclic* : $\forall R,$
 $(\forall l, \exists R', \text{sub_rel } (\text{restriction } R l) R' \wedge \text{transitive } R' \wedge \text{irreflexive } R') \rightarrow$
irreflexive (*clos_trans* *R*).

Proof Let *R* be a relation over *A*. Assume that any finite restriction of *R* is included in some strict partial order. Let *x* be in *A* such that *clos_trans* *R x x*. Then derive *False* as follows. Invoke *clos_trans_path* and get a path. It is still a path for the restriction of *R* to the path itself (the path is a list seen as a subset of *A*). Use *path_clos_trans*, then the main assumption, *transitive_sub_rel_clos_trans*, and the monotonicity of *clos_trans* with respect to *sub_rel*. \square

Lemma *total_order_eq_midex* :

$(\forall l, \exists R, \text{transitive } R \wedge \text{irreflexive } R \wedge \text{total } R \mid \wedge \text{rel_midex } R) \rightarrow \text{eq_midex}.$

Proof Assume the left conjunct, let x and y be in A , use the assumption with $x::y::\text{nil}$, get a relation, and double case split on x and y being related either way. \square

Consider a middle-excluding relation on A . It is acyclic and equality is middle-excluding on A iff for any list over A there exists, on the given list, a decidable strict total order containing the original relation.

Theorem *linearly_extendable* : $\forall R, \text{rel_midex } R \rightarrow$

$(\text{eq_midex} \wedge \text{irreflexive } (\text{clos_trans } R) \leftrightarrow$

$\forall l, \exists R', \text{linear_extension } R \mid R' \wedge \text{rel_midex } R').$

Proof Left to right: by the relevant lemmas of subsection 3.6.5, $(\text{LETS } R \mid)$ is a witness for the existential quantifier. Right to left by *local_global_acyclic* and *total_order_eq_midex*. \square

3.6.7 Topological Sorting

In this subsection, excluded-middle results of subsection 3.6.6 are translated into decidability results and augmented: as there is only one concept of linear extension in subsection 3.6.6, this section presents three slightly different concepts of topological sort. Instead of the equivalence of theorem *linearly_extendable*, those three definitions yield a quadruple equivalence.

From now on a decidable relation may be represented by a function to booleans instead of a function to *Prop* satisfying the definition *rel_dec*. However, those two representations are “equivalent” thanks to lemmas *rel_dec_bool* and *bool_rel_dec* in subsection 3.2.2.

In this chapter, a given relation over A is said to be non-uniformly (topologically) sortable if the restriction of the relation to any list has a decidable linear extension.

Definition *non_uni_topo_sortable* $R :=$

$\forall l, \exists R' : A \rightarrow A \rightarrow \text{bool}, \text{linear_extension } R \mid (\text{fun } x \ y \Rightarrow R' \ x \ y = \text{true}).$

In the definition above, R' represents a decidable binary relation that intends to be a linear extension of R over the list l . But R' has type $A \rightarrow A \rightarrow \text{bool}$ so it cannot be used with the predicate *linear_extension* $R \mid$ that expects an object of type $A \rightarrow A \rightarrow \text{Prop}$, which is the usual type for representing binary relations in Coq. The function $\text{fun } x \ y \Rightarrow (R' \ x \ y) = \text{true}$ above is the translation of R' in the suitable type/representation. It expects two elements x and y in A and returns the proposition $R' \ x \ y = \text{true}$, in *Prop*.

In this chapter, a given relation over A is said to be uniformly sortable if there exists a computable function expecting a list over A and producing, over the list argument, a (decidable) linear extension of the original relation.

Definition *uni_topo_sortable* $R := \{F : \text{list } A \rightarrow A \rightarrow A \rightarrow \text{bool} \mid$

$\forall l, \text{linear_extension } R \mid (\text{fun } x \ y \Rightarrow (F \ l \ x \ y) = \text{true})\}.$

The third definition of topological sort uses the concept of *asymmetry*, which is now informally introduced; from an algorithmic viewpoint: given a way of

representing binary relations, different objects may represent the same binary relation; from a logical viewpoint: two binary relations different in intension, *i.e.* their definitions intend different things, may still coincide, *i.e.* may be logically equivalent. In an arbitrary topological sort algorithm, the returned linear extension may depend on which object has been chosen to represent the original binary relation. For example, applying the empty relation on a given two-element set to a topological sort algorithm may produce the two possible linear extensions depending on the order in which the two elements constituting the set are given. This remark leads to the definition below. It is not clear what this notion of asymmetry would be useful for, but a result follow.

Definition $asym\ R\ G := \forall x\ y : A,$
 $x \neq y \rightarrow \neg R\ x\ y \rightarrow \neg R\ y\ x \rightarrow \neg(G\ (x::y::nil)\ x\ y \wedge G\ (y::x::nil)\ x\ y).$

Next comes the definition of asymmetry for a topological sort of a binary relation. The syntax *let variable := formula in formula'* avoids writing *formula* several times in *formula'*.

Definition $asym_topo_sortable\ R := \{F : list\ A \rightarrow A \rightarrow A \rightarrow bool \mid$
 $let\ G := (fun\ l\ x\ y \Rightarrow F\ l\ x\ y = true)\ in$
 $asym\ R\ G \wedge \forall l, linear_extension\ R\ l\ (G\ l)\}.$

Given a binary relation R over A , the remainder of this subsection proves that the four following assertions are equivalent:

1. Equality on A is decidable, and R is decidable and acyclic.
2. R is middle-excluding and asymmetrically sortable.
3. R is decidable and uniformly sortable.
4. Equality on A is decidable, and R is decidable and non-uniformly sortable.

The following lemma says that if there exists a computable function expecting a list over A and producing a (decidable) strict total order over A , then equality on A is decidable. The proof is similar to the one for *total_order_eq_midex*.

Lemma $total_order_eq_dec :$
 $\{F : list\ A \rightarrow A \rightarrow A \rightarrow bool \mid \forall l, let\ G := fun\ x\ y \Rightarrow F\ l\ x\ y = true\ in$
 $transitive\ A\ G \wedge irreflexive\ G \wedge total\ G\ l\} \rightarrow eq_dec\ A.$

Next lemma shows that *LETS* yields asymmetric topological sort.

Lemma $LETS_asym : \forall R, asym\ R\ (LETS\ R).$

Proof Assume all possible premises, especially let x and y be in A . As a preliminary: the hypotheses involve one relation image of *restriction* and four relations images of *try_add_arc*. Prove that all of them are restricted to $x::y::nil$. Then perform a few cases splittings and apply *clos_trans_restricted_pair* seven times. \square

The quadruple equivalence claimed above is derived from *rel_dec_midex* and the six theorems below. The proofs are rather similar to the middle-excluding case in subsection 3.6.6. The first theorem proves $1 \rightarrow 2$ by the relevant lemmas of subsection 3.6.5 and *LETS* producing a witness for the computational existence. The second (straightforward) and the third show $2 \rightarrow 3$. The fourth (straightforward) and the fifth, proved by *total_order_eq_dec*, yield $3 \rightarrow 4$. The last shows $4 \rightarrow 1$ by invoking *local_global_acyclic*.

Theorem *possible_asym_topo_sorting* : $\forall R,$
 $eq_dec A \rightarrow rel_dec R \rightarrow irreflexive (clos_trans A R) \rightarrow asym_topo_sortable R.$

Theorem *asym_topo_sortable_uni_topo_sortable* : $\forall R,$
 $asym_topo_sortable R \rightarrow uni_topo_sortable R.$

Theorem *asym_topo_sortable_rel_dec* : $\forall R,$
 $rel_midex R \rightarrow asym_topo_sortable R \rightarrow rel_dec R.$

Proof First notice that R is acyclic by *local_global_acyclic* and that equality on A is decidable by *total_order_eq_dec*. Then let x and y be in A . By decidable equality, case split on x and y being equal. If they are equal then they are not related by acyclicity. Now consider that they are distinct. Thanks to the assumption, get TS an asymmetric topological sort of R . Case split on x and y being related by TS ($x::y::nil$). If they are not then they cannot be related by R by subrelation property. If they are related then case split again on x and y being related by TS ($y::x::nil$). If they are not then they cannot be related by R by subrelation property. If they are then they also are by R by the asymmetry property. \square

Theorem *uni_topo_sortable_non_uni_topo_sortable* : $\forall R,$
 $uni_topo_sortable R \rightarrow non_uni_topo_sortable R.$

Theorem *rel_dec_uni_topo_sortable_eq_dec* : $\forall R,$
 $rel_dec R \rightarrow uni_topo_sortable R \rightarrow eq_dec A.$

Theorem *rel_dec_non_uni_topo_sortable_acyclic* : $\forall R,$
 $rel_dec R \rightarrow non_uni_topo_sortable R \rightarrow irreflexive (clos_trans A R).$

3.7 Conclusion

This chapter has given a detailed account on a few facts related to linear extensions of acyclic binary relations. The discussion is based on a formal proof developed with the proof assistant Coq. Since arguments are constructive, they are also correct with respect to traditional mathematical reasoning. The chapter aims to be understandable to mathematicians new to Coq or more generally by readers unfamiliar with constructive issues. The three main results are stated again below. First, a binary relation over a set with decidable/middle-excluding equality is decidable/middle-excluding *iff* transitive closures of its finite restrictions are also decidable/middle-excluding. This theorem is involved in the proof of the second and third main results. Second, consider a middle-excluding relation over an arbitrary domain. It is acyclic and equality

on its domain is middle-excluding *iff* any of its finite restriction has a middle-excluding linear extension. Third, consider R a decidable binary relation over A . The following three propositions are equivalent:

- Equality on A is decidable and R is acyclic.
- Equality on A is decidable and R is *non-uniformly* sortable.
- R is *uniformly* sortable.

The proofs of the last two main results rely on the constructive function *LETS* that is actually (similar to) a basic topological sort algorithm. An effective program could therefore be extracted from the Coq development (related to computability). The original proof would in turn serve as a formal proof of correctness for the program.

Chapter 4

Acyclicity of Preferences, Nash Equilibria, and Subgame Perfect Equilibria: a Formal and Constructive Equivalence

Sequential game and Nash equilibrium are basic key concepts in game theory. In 1953, Kuhn showed that every sequential game has a Nash equilibrium. The two main steps of the proof are as follows: First, a procedure expecting a sequential game as an input is defined as “backward induction” in game theory. Second, it is proved that the procedure yields a Nash equilibrium. “Backward induction” actually yields Nash equilibria that define a proper subclass of Nash equilibria. In 1965, Selten named this proper subclass subgame perfect equilibria. In game theory, payoffs are rewards usually granted at the end of a game. Although traditional game theory mainly focuses on real-valued payoffs that are implicitly ordered by the usual total order over the reals, there is a demand for results dealing with non totally ordered payoffs. In the mid 1950’s, works of Simon or Blackwell already involved partially ordered payoffs. This chapter further explores the matter: it generalises the notion of sequential game by replacing real-valued payoff functions with abstract atomic objects, called outcomes, and by replacing the usual total order over the reals with arbitrary binary relations over outcomes, called preferences. This introduces a general abstract formalism where Nash equilibrium, subgame perfect equilibrium, and “backward induction” can still be defined. Using a lemma on topological sorting, this chapter proves that the following three propositions are equivalent: 1) Preferences over the outcomes are acyclic. 2) Every sequential game has a Nash equilibrium. 3) Every sequential game has a subgame perfect equilibrium. The result is fully computer-certified using the (highly reliable) constructive proof assistant called Coq. Beside the additional guarantee of correctness provided by the proof in Coq, the activity of formalisation also helps clearly identify the useful definitions and the main articulations of the proof.

4.1 Introduction

4.1.1 Contribution

This chapter contributes at both the technical and the presentation level. There are five main technical contributions:

- An *inductive formalism* is designed to represent sequential games in the constructive proof assistant Coq ([1] and [9]), and all the results in this chapter are proved in Coq.
- The new formalism allows the chapter to introduce an *abstraction of traditional sequential games* and of a few related concepts. The abstraction preserves the tree structure of the game but replaces the real-valued pay-off functions, enclosed in the leaves of the trees, by arbitrary outcomes. Each agent has a preference over outcomes, which is given *via* an explicit and arbitrary binary relation. This preference replaces the implicit and traditional “usual total order” over the real numbers. Nash equilibria and subgame perfect equilibria are defined accordingly.
- Kuhn’s result [25] is translated into the new formalism when agents’ preferences are totally ordered.
- The notion of “backward induction” is naturally generalised for arbitrary preferences. However, a simple example shows that total ordering of preferences is needed for “backward induction” to guarantee subgame perfect equilibrium: both notions of “backward induction” and subgame perfect equilibrium coincide for total orders but not in general.
- Kuhn’s result is *substantially generalised* as follows. On the one hand, an intermediate result proves that smaller preferences, *i.e.*, binary relations with less arcs, yield more equilibria than bigger preferences. On the other hand, a topological sorting result was formally proved in [28] and chapter 3. By both results mentioned above, *acyclicity of the preferences proves to be a necessary and sufficient condition for every game to have a Nash equilibrium/subgame perfect equilibrium*.

This chapter deals with basic notions of game theory that are all exemplified and defined before they are used. Most of the time, these notions are explained in three different ways, with the second one helping make the connection between the two others:

- The notions are presented in a graphical formalism close to traditional game theory.
- They are presented in a graphical formalism suitable for induction.
- They are presented in a light Coq formalism close to traditional mathematics, so that only a basic understanding of Coq is needed. (A quick look at the first ten pages of [28] or chapter 3 will introduce the reader to the required notions.)

Moreover, the proofs are structured along the corresponding Coq proofs but are written in plain English.

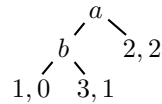
4.1.2 Contents

Section 4.2 gives an intuitive and informal presentation of traditional sequential games through graphical examples. Section 4.3 explores further the relevance of non-totally ordered payoffs. Section 4.4 discusses general concepts that are not specially related to game theory but required in the remainder of the chapter. Section 4.5 presents the above-mentioned abstraction of sequential games and their new formalism. Section 4.6 presents the notion of strategy profile at the same level of abstraction as for sequential games. Section 4.7 defines convertibility between strategy profiles. Section 4.8 discusses the notion of preference, happiness, Nash equilibrium, and subgame perfect equilibrium. Section 4.9 generalises “backward induction”, translates Kuhn’s result into the new formalism, and proves the triple equivalence between acyclicity of preferences and existence of Nash equilibrium/subgame perfect equilibrium.

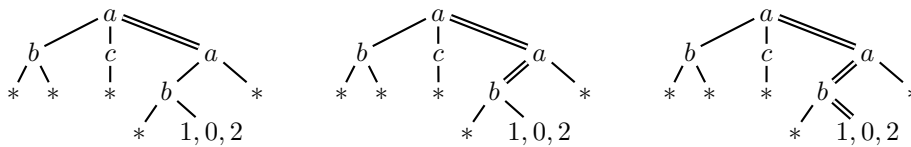
4.2 Traditional Sequential Game Theory

Through graphical examples and explanations in plain English, this section gives an intuitive and informal presentation of traditional sequential games and of a few related concepts such as Nash equilibrium.

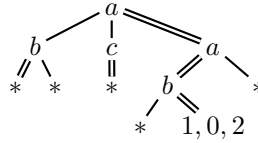
A traditional sequential game involves finitely many agents and payoffs. Payoffs usually are real numbers. A payoff function is a function from the agents to the payoffs. A sequential game is a rooted finite tree with internal nodes labelled with agents and external nodes, *i.e.*, leaves, labelled with payoff functions. Consider the following graphical example of such games. It involves the two agents a and b . At every leaf, a payoff function is represented by two numbers separated by a comma: the payoff function maps agent a to the first number and agent b to the second number.



Such game trees are interpreted as follows: A play of a game starts at the root of the game tree. If the tree is a leaf then agents are rewarded according to the enclosed payoff function. Otherwise, the agent owning the root chooses the next node among the children of the root. The subtree rooted at the chosen child is considered and the play continues from there. In the game above, if a chooses to go right then both agents get 2. If a chooses left then b has to choose too, etc. Now, a specific play of a game is described and drawn. Double lines represent choices made during the play. Agent a first chooses to go right, then left, then b chooses to go right. Eventually, a gets 1, b gets 0, and c gets 2. The stars $*$ represent arbitrary payoff functions irrelevant to the discussion.



In game theory, the strategy of an agent is an object that accounts for the decisions of the agent in all situations that the agent might encounter. A strategy profile is a tuple combining one strategy per agent. So, for sequential games, a strategy profile amounts to choices made at all internal nodes of the tree. Below is an example of a strategy profile. Double lines between nodes represent choices and the stars * represent arbitrary payoff functions irrelevant to the discussion. The choice of b at the leftmost internal node may seem rather ineffective, but it can be interpreted as b 's choice if a play ever reach this very node.



Given a strategy profile, starting from the root and following the agents' consecutive choices leads to one leaf. The payoff function enclosed in that specific leaf is called the induced payoff function. The induced payoff function of the strategy profile above is: a gets 1, b gets 0, and c gets 2.

The (usually implicit) preference of agents for strictly greater payoffs induces a (usually implicit) preference of an agent for payoff functions that grants him strictly greater payoffs. This, in turn, yields a (usually implicit) preference of an agent for strategy profiles inducing preferred payoff functions. Below, agent a prefers the left-hand strategy profile to the right-hand one since 3 is greater than 2, but it is the opposite for agent b since 1 is less than 2.



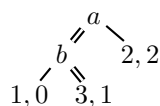
An agent is (usually implicitly) granted the ability to change his choices at all nodes he owns. For instance, below, the agent b can convert the strategy profile on the left to the one on the right by changing his choices exactly at the nodes where b is displayed in bold font. The stars * represent arbitrary payoff functions irrelevant to the discussion.



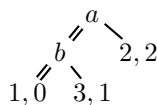
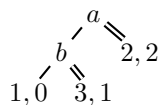
An agent is said to be happy with a strategy profile if he cannot convert it to another strategy profile that he prefers. A Nash equilibrium is a strategy profile that makes all agents happy. Below, the strategy profile to the left is not a Nash equilibrium since its sole player gets 0 but can convert it to the right-hand strategy profile and get 1. However, the right-hand strategy profile is a Nash equilibrium since a cannot convert it and get a payoff strictly greater than 1.



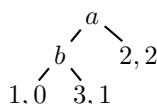
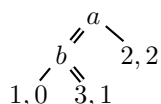
Here is another example of Nash equilibrium.



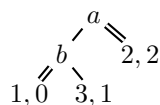
Indeed, agent a could only convert the strategy profile above to the left-hand strategy profile below, and would get 2 instead of 3. Therefore a is happy. Agent b could only convert the strategy profile above to the right-hand strategy profile below, and would get 0 instead of 1. Therefore b is happy too. The strategy profile above makes all players happy; it is a Nash equilibrium.



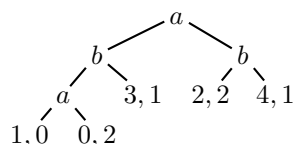
The underlying game of a strategy profile is computed by forgetting all the choices made at the internal nodes of the strategy profile. The next picture displays a strategy profile, to the left, and its underlying game, to the right.



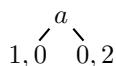
Two Nash equilibria inducing different payoff functions may have the same underlying game: indeed, consider the previous Nash equilibrium and the following one, where b 's choice is ineffective in terms of induced payoff function.



A given game is said to have a Nash equilibrium if there exists a Nash equilibrium whose underlying game is the given game. In order to prove that every sequential game has a Nash equilibrium, one can use a construction called "backward induction" in game theory. It consists in building a strategy profile from a sequential game. Performing "backward induction" on the following example will help describe and interpret the idea of the construction.

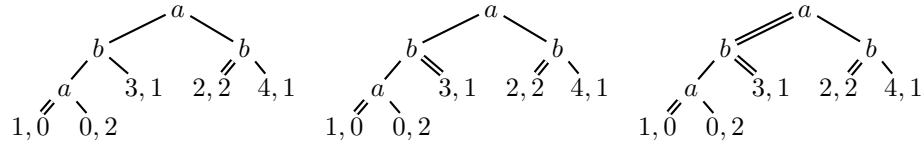


If a play starts at the leftmost and lowest node of the game above, then agent a faces the following game:



So, provided that a is "rational" in some informal sense, he chooses left and get 1 instead of 0. In the same way, if the play starts at the rightmost node, b chooses left and get 2 instead of 1. These two remarks correspond to the leftmost picture below. Provided that agent b is aware of a 's "rational"

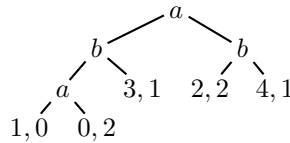
behaviour, if a play starts at the left node owned by b , then b chooses right and get 1 instead of 0, as shown on the second picture below. The last picture shows that when a play starts at the root, as in the usual interpretation of a sequential game, a chooses left and gets 3 instead of 2. In such a process, an agent facing several options equivalent in terms of payoffs may choose either of them.



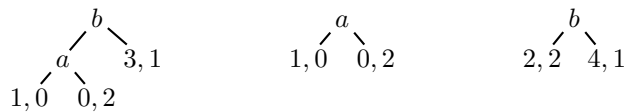
A strategy profile built by “backward induction” is a Nash equilibrium whose underlying game is the original game. (A formal proof relying on formal definitions is presented in a later section.) This way, it is proved that all sequential games have Nash equilibria. However, the next example shows that not all Nash equilibria are obtained by “backward induction”. Even stronger, a Nash equilibrium may induce a payoff function induced by no “backward induction” strategy profile. Indeed, the left-hand strategy profile below is a Nash equilibrium that is not a “backward induction”, and the only “backward induction” on the same underlying game is shown on the right-hand side.



Traditional game theory uses the following definition of subtree: all the descendants of a node of a tree define a subtree. For instance consider the following game.



In addition to the leaves, the proper subtrees of the game above are listed below.



With this definition of subtree, a subgame perfect equilibrium is defined as a Nash equilibrium all of whose substrategy profiles are also Nash equilibria. In traditional game theory, the notions of subgame perfect equilibrium and “backward induction” strategy profile coincide.

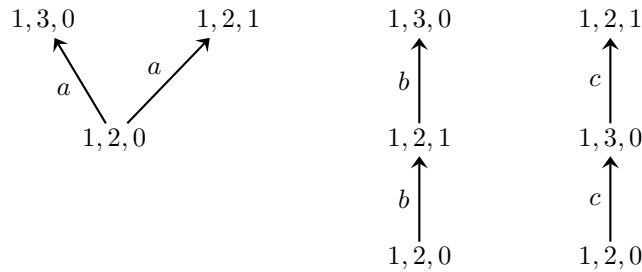
4.3 Why Total Order?

Section 4.2 only deals with real-valued, totally ordered payoffs. However, as mentioned in subsection 1.3.6, there is a need for game theoretic results involving non totally ordered payoffs. This section adds simple and informal arguments to the discussion. In particular, it shows that the class of traditional sequential games naturally induces two classes of games slightly more general than itself. For these classes of games, the question whether Nash equilibria/subgame perfect equilibria exist or not is still relevant, and has yet not been addressed by Kuhn's result.

4.3.1 Selfishness Refinements

An agent that gives priority to his own payoffs without taking other agents into consideration is called selfish. It is the case in traditional game theory. Now consider a benevolent agent that takes all agents, including him, into account¹. More specifically, consider two payoff functions p and p' . If for each agent, p grants a payoff greater than or equal to the one granted by p' , and if there exists one agent to whom p grants strictly greater payoff than p' , then the benevolent agent prefers p to p' . For instance, consider three agents a , b and c , and three payoff functions $(1, 3, 0)$, $(1, 2, 1)$, and $(1, 2, 0)$. (The first component corresponds to a , the second to b , and the third to c .) A benevolent agent prefers the first two to the last one, but has no preference among the first two.

An agent is selfish-benevolent if his preference is the union of selfish and benevolent preferences. Put otherwise, an agent prefers a payoff function to another one if he prefers it either selfishly or benevolently. For instance, consider the previous example. Assume that all agents are selfish-benevolent. So, b prefers $(1, 3, 0)$ to $(1, 2, 1)$ since 3 is greater than 2, prefers $(1, 2, 1)$ to $(1, 2, 0)$ by benevolence, and prefers $(1, 3, 0)$ to $(1, 2, 0)$ by both selfishness and benevolence. Selfishness-benevolence induces a partial order over real-valued payoff functions. These partial orders are represented below.



The following shows that using real-valued payoff functions with selfish-benevolence cannot be modelled by using real-valued payoff functions with the usual preference (agents only consider their own payoffs). To show this, assume that there exists an embedding sending the formers into the latters, *i.e.* for 3-agent payoff functions, $(x, y, z) \mapsto (f(x, y, z), g(x, y, z), h(x, y, z))$ such that $(x, y, z) <_a (x', y', z') \Leftrightarrow f(x, y, z) <_{\mathbb{R}} f(x', y', z')$ (and the same kind of property must hold for agents b and c). Indeed, for the embedding to be a modelling, it must preserve and reflect the preference order between payoff

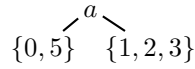
¹*e.g.* in a Pareto style

functions. So, all the following formulas must hold: $f(0, 1, 0) <_{\mathbb{R}} f(0, 2, 0)$ and $\neg(f(0, 0, 1) <_{\mathbb{R}} f(0, 2, 0))$ and $\neg(f(0, 1, 0) <_{\mathbb{R}} f(0, 0, 1))$. However, the first two formulas imply $f(0, 1, 0) <_{\mathbb{R}} f(0, 0, 1)$, hence a contradiction.

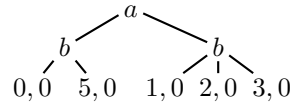
In the same way, an agent may be selfish-malevolent. More specifically, consider two payoff functions p and p' . If p grants a selfish-malevolent agent a payoff greater than the one granted by p' then the agent prefers p to p' . If p and p' grant the same payoff to the selfish-malevolent agent, and if p grants every other agent a payoff lesser than or equal to the one granted by p' , and if there exists one agent to whom p grants strictly lesser payoff than p' , then the selfish-malevolent agent prefers p to p' . Selfishness-malevolence induces a partial order over real-valued payoff functions too.

4.3.2 Lack of Information

Consider an agent that prefers greater payoffs and that is “rational”, *i.e.*, acts according to his preferences. Imagine the following 1-player game played by the above-mentioned agent: when a play starts, the agent has two options, say left and right. If he chooses left then he gets either 0 or 5, and if he chooses right then he gets either 1 or 2 or 3. After payoffs are granted, the play ends. This game is represented below.



The wording “either... or...”, in the phrase “either 0 or 5” does not refer to any procedure whatsoever. Therefore, in each case the agent has no clue how payoffs are going to be granted. It is worth stressing that, in particular, “either 0 or 5” does not refer to probability half for 0 and probability half for 5. It does not refer to probabilities at all. As a result, the agent cannot dismiss for sure any of his options in the game above. The two options are not comparable; the payoffs are not totally ordered. This type of game can even help understand traditional sequential games better. Indeed, consider the traditional sequential game below, where a and b are “rational”, and therefore only care about maximising their own payoffs. Also assume that a knows b ’s being “rational”.



Whatever agent a may choose, b ’s options are equivalent since they yield the same payoff. Therefore a has no clue how b is going to choose: Go left when left and right are equivalent? Toss a coin? Phone a friend? So, from agent a ’s viewpoint, the game above reduces to the 1-player game discussed before. In this subsection, non totally ordered payoffs represents lack of information without invoking a formalism dedicated to knowledge representation such as epistemic logic.

When payoffs are non-empty sets of real numbers instead of single real numbers, as in the first example above, there are several ways to define relevant preferences. For instance, agents can focus either on the lower bound of

the set, which amounts to guaranteeing a minimum, or on the upper bound of the set, which amounts to letting hope for a maximum, or both at once. Most of these yield partial orders over non-empty sets of reals.

4.4 Preliminaries

Prior to the game theoretic development presented in Coq in later sections, a few general concepts and related results are needed. Part of them are mentioned in [28] and chapter 3: list inclusion, subrelation, restriction of a relation to the elements of a list, etc. This section completes the inventory of the required notions, in the Coq formalism.

A first useful result reads as follows: given a binary relation and two lists, one included in the other, the restriction of the relation to the smaller list is a subrelation of the restriction of the relation to the bigger list. The proof is a straightforward unfolding of the definitions, and the result is formally written below.

Lemma *sub_rel_restriction_incl* : $\forall (B : \text{Set})(l' : \text{list } B) R,$
 $\text{incl } l' \rightarrow \text{sub_rel } (\text{restriction } R \ l) (\text{restriction } R \ l').$

The remainder of this section presents the extension to lists of four concepts usually pertaining to one or two objects only.

4.4.1 Extension of Predicates to Lists

Let A be a *Set*. The function *listforall* expects a predicate on A , i.e., an object of type $A \rightarrow \text{Prop}$, and returns a predicate on lists, i.e., an object of type $\text{list } A \rightarrow \text{Prop}$, stating that all the elements in the list comply with the original predicate. It is recursively defined along the inductive structure of the list argument.

Fixpoint *listforall* ($Q : A \rightarrow \text{Prop}$)($l : \text{list } A$){*struct* l } : $\text{Prop} :=$
match l *with*
 | *nil* $\Rightarrow \text{True}$
 | $x::l' \Rightarrow Q \ x \wedge \text{listforall } Q \ l'$
end.

This paragraph is intended to the reader who is not familiar with Coq: In the first line above, *Fixpoint* starts the recursive definition, *listforall* is the name of the defined function, Q is the predicate argument, l is the list argument, {*struct* l } means that the recursion involves l , and Prop is the type of the output. The *match* performs a case splitting on the structure of l . The line thereafter specifies that the function returns *True* for empty list arguments. The last line is the core of the recursion: in order to compute the result for the list, it refers to the result of the computation involving the tail, which is a strictly smaller argument. This ensures termination of the computation, therefore the function is well defined. An example of computation of *listforall* is given below. The symbol \rightsquigarrow represents a computation step.

$\text{listforall } Q \ (x::y::z::\text{nil}) \rightsquigarrow Q \ x \wedge \text{listforall } Q \ (y::z::\text{nil}) \rightsquigarrow$
 $Q \ x \wedge Q \ y \wedge \text{listforall } Q \ (z::\text{nil}) \rightsquigarrow Q \ x \wedge Q \ y \wedge Q \ z \wedge \text{listforall } Q \ \text{nil} \rightsquigarrow$
 $Q \ x \wedge Q \ y \wedge Q \ z \wedge \text{True}$

Note that $Q\ x \wedge Q\ y \wedge Q\ z \wedge \text{True}$ is equivalent to $Q\ x \wedge Q\ y \wedge Q\ z$, which is what the function *listforall* is meant for.

The following four lemmas involve the notion of appending ($++$), also called concatenation, of two lists. It is defined in the Coq Standard Library. The four lemmas express basic properties of the *listforall* function. They are all proved by induction on the list l .

Lemma *listforall_app* : $\forall Q\ l'\ l,$
 $\text{listforall } Q\ l \rightarrow \text{listforall } Q\ l' \rightarrow \text{listforall } Q\ (l++l').$

Lemma *listforall_appl* : $\forall Q\ l'\ l, \text{listforall } Q\ (l++l') \rightarrow \text{listforall } Q\ l.$

Lemma *listforall_appr* : $\forall Q\ l'\ l, \text{listforall } Q\ (l++l') \rightarrow \text{listforall } Q\ l'.$

Lemma *listforall_In* : $\forall Q\ x\ l, \text{In } x\ l \rightarrow \text{listforall } Q\ l \rightarrow Q\ x.$

4.4.2 Extension of Functions to Lists

The Coq Standard Library provides a function *map* that, given a list and a function f , returns a list with the images by f of the elements of the original list. It is defined by recursion.

Fixpoint *map* ($A\ B : \text{Set}$)($f : A \rightarrow B$)($l : \text{list } A$) : $\text{list } B :=$
 $\text{match } l \text{ with}$
 $\mid \text{nil} \Rightarrow \text{nil}$
 $\mid a::t \Rightarrow (f\ a)::(\text{map } A\ B\ f\ t)$
 $\text{end}.$

Consider the simplified computation example below, where domains and codomains of f are omitted for better readability.

$\text{map } f\ (x::y::z::\text{nil}) \rightsquigarrow \dots \rightsquigarrow (f\ x)::(f\ y)::(f\ z)::\text{nil}$

The next two lemmas state *map*'s preserving two functions being inverse and commutativity between *map* and appending. Both lemmas are proved by induction on the list l . The second one comes from the Coq Standard Library [1].

Lemma *map_inverse* : $\forall (A\ B : \text{Set})(f : A \rightarrow B)\ g,$
 $(\forall x, g\ (f\ x)=x) \rightarrow \forall l, \text{map } g\ (\text{map } f\ l)=l.$

Lemma *map_app* : $\forall (A\ B : \text{Set})\ l\ l'\ (f : A \rightarrow B), \text{map } f\ (l++l') = (\text{map } f\ l)++(\text{map } f\ l').$

4.4.3 Extension of Binary Relations to Lists

Let A be a *Set*. Given a binary relation $P : A \rightarrow A \rightarrow \text{Prop}$, the function *rel_vector* expects two lists over A and states that they are component-wise related by P . Note that if they are component-wise related then their lengths are the same.

Fixpoint *rel_vector* ($P : A \rightarrow A \rightarrow \text{Prop}$)($l\ l' : \text{list } A$) {*struct* l }: $\text{Prop} :=$
 $\text{match } l \text{ with}$
 $\mid \text{nil} \Rightarrow l'=\text{nil}$
 $\mid x::l2 \Rightarrow \text{match } l' \text{ with}$


```

| nil  $\Rightarrow$  False
| x'::l2'  $\Rightarrow$  P x x'  $\wedge$  rel_vector P l2 l2'
end

```

end.

The following examples describe three typical computations of the *rel_vector* predicate.

$$\text{rel_vector } P (x::y::\text{nil}) (x'::y'::\text{nil}) \rightsquigarrow P x x' \wedge \text{rel_vector } P (y::\text{nil}) (y'::\text{nil}) \rightsquigarrow P x x' \wedge P y y' \wedge \text{rel_vector } P (\text{nil}) (\text{nil}) \rightsquigarrow P x x' \wedge P y y' \wedge \text{nil}=\text{nil}$$

Note that $P x x' \wedge P y y' \wedge \text{nil}=\text{nil}$ is equivalent to $P x x' \wedge P y y'$.

$$\text{rel_vector } P (x::y::\text{nil}) (x'::\text{nil}) \rightsquigarrow P x x' \wedge \text{rel_vector } P (y::\text{nil}) (\text{nil}) \rightsquigarrow P x x' \wedge \text{False}$$

Note that $P x x' \wedge \text{False}$ is equivalent to *False*.

$$\text{rel_vector } P (x::\text{nil}) (x'::y'::\text{nil}) \rightsquigarrow P x x' \wedge \text{rel_vector } P (\text{nil}) (y'::\text{nil}) \rightsquigarrow P x x' \wedge y'::\text{nil}=\text{nil}$$

Note that $P x x' \wedge y'::\text{nil}=\text{nil}$ is equivalent to *False* since $y'::\text{nil}=\text{nil}$ is equivalent to *False*.

The following lemma states that if two lists are component-wise related, then two elements occurring at the same place in each list are also related.

Lemma *rel_vector_app_cons_same_length* : $\forall P a a' m m' l l',$
 $\text{rel_vector } P (l++a::m) (l'+a'::m') \rightarrow \text{length } l=\text{length } l' \rightarrow P a a'.$

Proof Let P be a binary relation over A , let a and a' be in A , and let m and m' be lists over A . Prove $\forall l l', \text{rel_vector } P (l++a::m) (l'+a'::m') \rightarrow \text{length } l=\text{length } l' \rightarrow P a a'$ by induction on l . For the inductive case, implying that l' is non-empty, apply the induction hypothesis with the tail of l' . \square

The next result shows that if two lists are component-wise related, then given one element in the second list, one can compute an element of the first list such that both elements are related.

Lemma *rel_vector_app_cons_exists* : $\forall P a q l m,$
 $\text{rel_vector } P l (m++a::q) \rightarrow \{x : A \mid \text{In } x l \wedge P x a\}.$

Proof By induction on l and case splitting on m being empty. For the inductive case, if m is empty then the head of l is a witness, if m is not empty then use the induction hypothesis with the tail of m ; the computable element is a witness. \square

The following lemma says that if two lists are component-wise related, then given one element in the first list and one in the second list, but at different places, there is another element in the first list, either before or after the element mentioned first, that is related to the element of the second list.

Lemma *rel_vector_app_cons_different_length* : $\forall P l a m l' a' m',$
 $\text{rel_vector } P (l++a::m) (l'+a'::m') \rightarrow \text{length } l \neq \text{length } l' \rightarrow \{x : A \mid (\text{In } x l \vee \text{In } x m) \wedge P x a'\}.$

Proof By induction on l . For the base case, l is empty, if l' is empty then it is straightforward, if l' is not empty then applying *rel_vector_app_cons_exists* gives a witness. For the inductive case, l is not empty, case split on l' being empty and use the induction hypothesis when l' is not empty. \square

4.4.4 No Successor

Let A be a *Set*. Given a binary relation, the predicate *is_no_succ* returns a proposition saying that a given element is the predecessor of no element in a given list.

Definition *is_no_succ* ($P : A \rightarrow A \rightarrow \text{Prop}$)($x : A$)($l : \text{list } A$) :=
listforall ($\text{fun } y \rightarrow \neg P \ x \ y$) l .

The next two results show a transitivity property and decidability of *is_no_succ* when built on a decidable binary relation. Both are proved by induction on the list l .

Lemma *is_no_succ_trans* : $\forall P \ x \ y \ l$,
transitive $A \ P \rightarrow P \ x \ y \rightarrow \text{is_no_succ } P \ x \ l \rightarrow \text{is_no_succ } P \ y \ l$.

Lemma *is_no_succ_dec* : $\forall P, \text{rel_dec } P \rightarrow \forall x \ l$,
 $\{\text{is_no_succ } P \ x \ l\} + \{\neg \text{is_no_succ } P \ x \ l\}$.

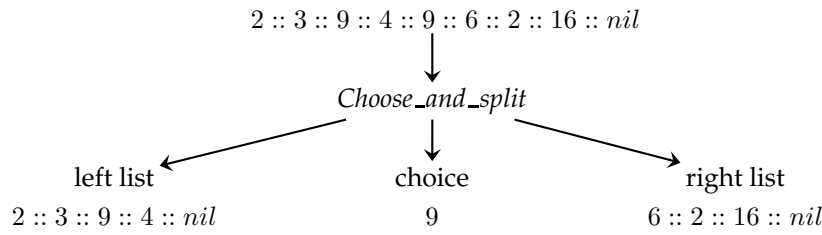
The following lemma helps generalise the notion of “backward induction” in section 4.9. Assume P a decidable binary relation over A , and $x::l$ a non-empty list over A . The list $x::l$ can be computably split into a left list, a chosen element, and a right list such that 1) the chosen element has no P -successor in the right list, 2) the chosen element is the first (from left to right) element with such a property, and moreover 3) if P is irreflexive and transitive then the chosen element has no P -successor in the left list either. The form of the statement has being slightly simplified, as compared to the actual Coq code. The first conjunct corresponds to the splitting of the list $x::l$, and the last three conjuncts correspond to the points 1), 2) and 3) as mentioned above.

Lemma *Choose_and_split* : $\forall P, \text{rel_dec } P \rightarrow \forall (l : \text{list } A)(x : A)$,
 $\{(\text{left}, \text{choice}, \text{right}) : \text{list } A \times A \times \text{list } A \mid$

$$\begin{aligned} & x::l = (\text{left} ++ (\text{choice}::\text{right})) \\ & \wedge \\ & \text{is_no_succ } P \ \text{choice } \text{right} \\ & \wedge \\ & (\forall \text{left}' \ \text{choice}' \ \text{right}', \text{left} = \text{left}' ++ (\text{choice}'::\text{right}') \rightarrow \\ & \neg \text{is_no_succ } P \ \text{choice}' \ (\text{right}' ++ (\text{choice}::\text{right}))) \\ & \wedge \\ & (\text{irreflexive } P \rightarrow \text{transitive } P \rightarrow \text{is_no_succ } P \ \text{choice } \text{left}) \}. \end{aligned}$$

Proof Assume that P is decidable and proceed by induction on l , the tail of the non-empty list. For the base case where the tail is empty, *nil*, x , and *nil* are witnesses for the required left list, choice element, and right list. For the step case, $l = a::l'$, case split on x having a successor in l . If not, then *nil*, x , and l are a witness. If x has a successor in l then use the induction hypothesis with a , which splits the list l into a left list, a choice element, and a right list. Put x on the top of the left list, here are the three witnesses. \square

For example, consider the partial order induced by divisibility of natural numbers by natural numbers. For the list $2 :: 3 :: 9 :: 4 :: 9 :: 6 :: 2 :: 16 :: \text{nil}$, the *choice* is 9, the left list is $2 :: 3 :: 9 :: 4 :: \text{nil}$, and the right list is $6 :: 2 :: 16 :: \text{nil}$.



Indeed, the first 2 can divide 4 written on its right, 3 can divide 9 written on its right, the first 9 can divide the other 9 written on its right, the 4 can divide 16 written on its right, but the second 9 cannot divide any number written on its right.

4.5 Sequential Games

By abstracting over payoff functions, subsection 4.5.1 generalises the notion of sequential game. In the remainder of this chapter, the expression “sequential game” refers to the new and abstract sequential games, unless stated otherwise. In addition, subsection 4.5.1 introduces a new formalism for sequential games. In subsection 4.5.2, an inductive proof principle is designed for these sequential games. Last, subsection 4.5.3 defines a function related to sequential games.

4.5.1 Definition of Sequential Games

This subsection first presents sequential games informally in the way traditional sequential games were presented in section 4.2. Then it describes sequential games in a both inductive and graphical formalism, and makes a link with the traditional formalism. Last, the inductive and graphical formalism is naturally translated into a definition of sequential games in Coq.

The Traditional Way:

Informally, consider a collection of outcomes that are the possible end-of-the-game situations, and a collection of agents that are the possible stake-holders and decision makers. Roughly speaking, an abstract sequential game is a traditional sequential game where each real-valued payoff function (enclosed in a leaf) has been replaced by an outcome. Below, the left-hand picture represents a traditional sequential game, and the right-hand picture represents an abstract sequential game on the same tree structure.



The Inductive and Graphical Way:

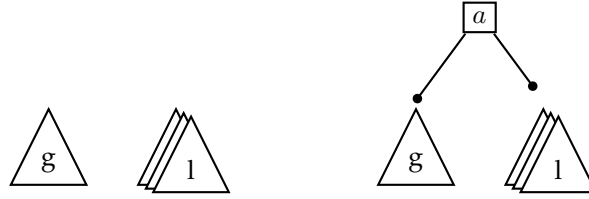
In what follows, a generic game g will be represented by the left-hand picture below and a generic (possibly empty) list of games by the right-hand picture.



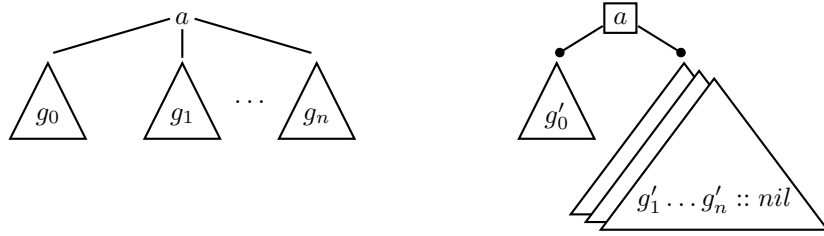
Sequential games are inductively defined in two steps. First step: If oc is an outcome then the object below is a game.



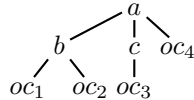
Second step: if a is an agent and if the first two objects below are a game and a list of games, then the rightmost object is also a game.



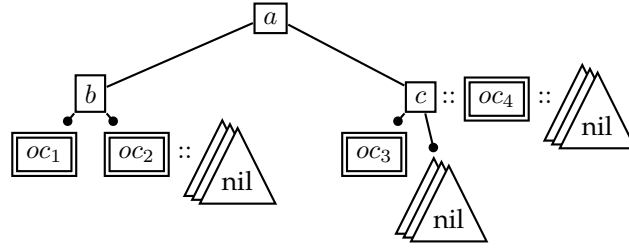
The major difference between the traditional formalism and the new formalism is as follows: In the traditional formalism, an internal node had one or arbitrarily many more children, which are games. In the new formalism instead, an internal node has one left child, which is a game, and one right child, which is a list of games. Since the list could be empty, the left-hand game ensures that at least one choice is available to the agent owning the node. The next two pictures represent the same game in both formalisms, where for all i between 0 and n , the object g'_i is the translation of g_i from the traditional formalism into the new formalism.



For instance, let a , b and c be three agents, and oc_1 , oc_2 , oc_3 and oc_4 be four outcomes. Consider the following game in the traditional formalism.



The game above is represented by the following game in the new formalism.



The Inductive and Formal Way:

Formally in Coq, games are defined as follows. First declare *Outcome* and *Agent*, two arbitrary types of type *Set*.

Variable *Outcome* : *Set*.

Variable *Agent* : *Set*.

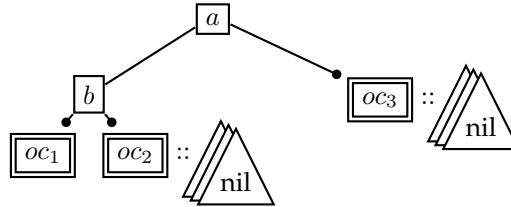
Next, the object *Game* is defined by induction.

Inductive *Game* : *Set* :=

| *gL* : *Outcome* → *Game*

| *gN* : *Agent* → *Game* → list *Game* → *Game*.

gL stands for “game leaf” and *gN* for “game node”. If *oc* is an *Outcome*, then *gL oc* is a *Game*. If *a* is an *Agent*, *g* is a *Game*, and *l* is a list of objects in *Game*, then *gN a g l* is a *Game*. The interpretation of such an object is as follows: the agent *a* “owns” the root of *gN a g l*, and *g* and *l* represent *a*’s options, i.e., the subgames *a* can decide to continue the play in. The structure ensures that any node of the tree meant to be internal has a non-zero number of children. The inductive Coq formalism and the inductive graphical formalism are very close to each other. For instance, compare the game *gN a (gL b oc1 oc2::nil) oc3::nil* with its representation in the inductive graphical formalism below.



Note that the inductive definition of games involves lists, which are already an inductive structure. Therefore, the game structure is inductive “horizontally and vertically”.

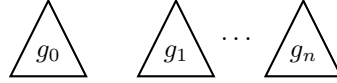
4.5.2 Induction Principle for Sequential Games

This subsection first discusses what would be an inductive proof principle for games in the traditional formalism. The failure of the former leads to an in-

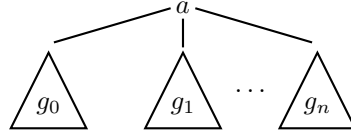
duction principle in the inductive graphical formalism. Then, the induction principle is easily translated in Coq formalism.

The Traditional Way:

In the traditional formalism, an informal induction principle would read as follows. In order to check that a predicate holds for all games, check two properties: First, check that the predicate holds for all games that are leaves (enclosing an outcome). Second, check that if the predicate holds for all the following games,



then, for any agent a , it holds for the following game.



However in a formal proof, dots do not mean much. In the new formalism proposed for games, they are formalised by lists. So, an induction principle for games must take lists into account.

The Inductive and Graphical Way:



In order to prove that the predicate P holds for all games, it suffices to design a predicate Q expecting a game and a list of games, and then to check that all the properties listed below hold. (In most of the proofs using this induction principle so far, Q morally means "for the game and all elements of the list, P holds")



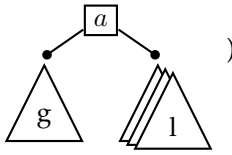
- For all outcome oc , $P(\boxed{\boxed{oc}})$.

- For all game $\triangle g$, if $P(\triangle g)$ then $Q(\triangle g, \text{nil})$

- For all game $\triangle g$ and $\triangle g'$ and all list of game $\text{list } l$,

if $P(\triangle g)$ and $Q(\triangle g', \text{list } l)$ then $Q(\triangle g', \text{list } g::l)$

- For all agent a , all game  and all list of game ,

if Q (, ) then P ()

The Inductive and Formal Way:

The induction principle that Coq automatically associates to the inductively defined structure of games is not efficient. A convenient principle has to be built (and hereby proved) manually, *via* a recursive function and the command *fix*. That leads to the following induction principle, which is therefore a theorem in Coq.

$Game_ind2 : \forall (P : Game \rightarrow Prop) (Q : Game \rightarrow list\ Game \rightarrow Prop),$
 $(\forall oc, P (gL\ oc)) \rightarrow$
 $(\forall g, P\ g \rightarrow Q\ g\ nil) \rightarrow$
 $(\forall g, P\ g \rightarrow \forall g' l, Q\ g'\ l \rightarrow Q\ g'\ (g :: l)) \rightarrow$
 $(\forall g l, Q\ g\ l \rightarrow \forall a, P (gN\ a\ g\ l)) \rightarrow$
 $\forall g, P\ g$

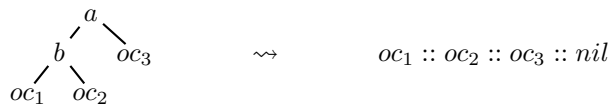
Two facts are worth noting: First, this principle corresponds to the one stated above in the inductive graphical formalism. Second, in order to prove a property $\forall g : Game, P\ g$ with the induction principle $Game_ind2$, the user has to imagine a workable predicate Q .

4.5.3 Outcomes Used in a Sequential Game

This subsection discusses the notion of outcomes used in a sequential game and defines a function that returns the “left-to-right” list of all the outcomes involved in a given game. Two lemmas follow the definition of the function in the Coq formalism.

The Traditional Way:

The example below explains what function is intended.



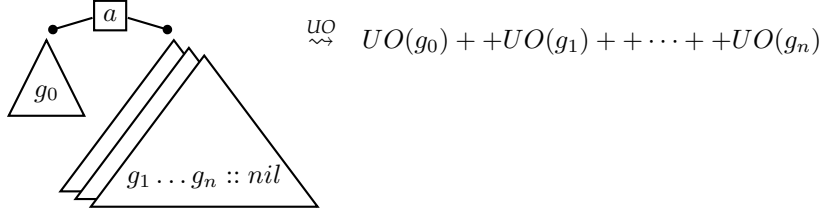
The Inductive and Graphical Way:

To prepare the full formalisation of the idea above, the intended function is defined recursively with the inductive graphical formalism, in two steps along

the structure of games. First step: the only outcome used in a leaf is the outcome enclosed in the leaf.

$$\boxed{oc} \stackrel{UO}{\rightsquigarrow} oc :: nil$$

Second step: recall that ++ refers to lists appending/concatenation.



The Inductive and Formal Way:

The intended function, called *UsedOutcomes*, is inductively and formally defined in Coq.

```

Fixpoint UsedOutcomes (g : Game) : list Outcome :=
match g with
| gL oc => oc::nil
| gN - g' l => ((fix ListUsedOutcomes (l' : list Game) : list Outcome :=
    match l' with
    | nil => nil
    | x::m => (UsedOutcomes x)++(ListUsedOutcomes m)
    end) (g':l))
end.

```

The following lemma states that the outcomes used in a game are also used in a structurally bigger game.

Lemma *UsedOutcomes_gN* : $\forall a g g' l,$
 $In g (g':l) \rightarrow incl (UsedOutcomes g) (UsedOutcomes (gN a g' l)).$

Proof By induction on l . For the inductive step case split as follows. If g equals the head of l then invoke *incl_appr*, *incl_appl*, and *incl_refl*. If g either equals g' or occurs in the tail of l then the induction hypothesis says that the outcomes used by g are used by g' and l together. The remainder mainly applies *in_app_or* and *in_or_app*. \square

The next result shows that if the outcomes used in a game all occur in a given list of outcomes (called *loc*), then so do the outcomes used in any subgame of the original game.

Lemma *UsedOutcomes_gN_listforall* : $\forall a g loc l,$
 $incl (UsedOutcomes (gN a g l)) loc \rightarrow listforall (fun g' : Game \Rightarrow incl (UsedOutcomes g') loc) (g::l).$

Proof All the following cases invoke *in_or_app*. It is straightforward for g . For elements of l proceed by induction on l . For the inductive step, it is straightforward for the head of l , and use the induction hypothesis for elements of the tail of l . \square

4.6 Strategy Profiles

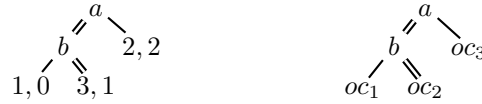
Consistent with subsection 4.5.1, subsection 4.6.1 generalises the notion of strategy profile by abstracting over payoff functions, and introduces a new formalism for them. In the remainder of this chapter, the expression “strategy profile” refers to the new and abstract strategy profiles, unless stated otherwise. In subsection 4.6.2, an inductive proof principle is designed for these strategy profiles. Subsection 4.6.3 defines the underlying game of a strategy profile. Last, subsection 4.6.4 defines the induced outcome of a strategy profile.

4.6.1 Definition of Strategy Profiles

This subsection first presents strategy profiles informally in the way traditional strategy profiles were presented in section 4.2. Then it describes strategy profiles in a both inductive and graphical formalism consistent with the one used to define sequential games. A correspondence between the traditional formalism and the new one is established. Finally, the inductive and graphical formalism is naturally translated into a definition of strategy profiles in Coq.

The Traditional Way:

Roughly speaking, an abstract strategy profile is a traditional strategy profile where each real-valued payoff function (enclosed in a leaf) has been replaced by an outcome. Below, the left-hand picture represents a traditional strategy profile, and the right-hand picture represents an abstract strategy profile on the same tree structure.



The Inductive and Graphical Way:

A generic strategy profile s will be represented by the left-hand picture below and a generic (possibly empty) list of strategy profiles by the right-hand picture.



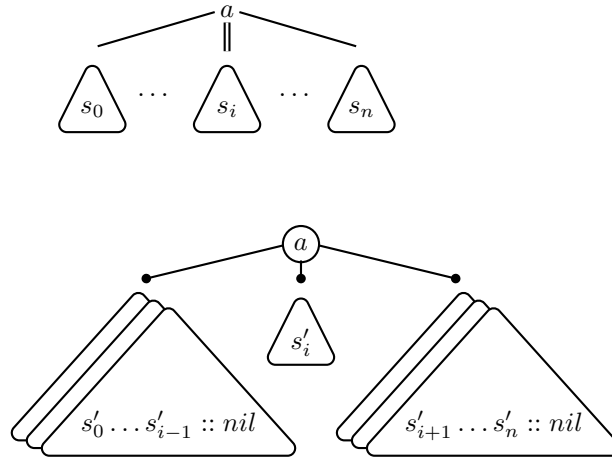
Strategy profiles are inductively defined in two steps. First step: If oc is an outcome then the object below is a strategy profile.



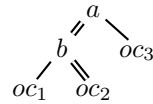
Second step: if a is an agent and the first three objects below are a strategy profile and two lists of strategy profiles, then the rightmost object is also a strategy profile.



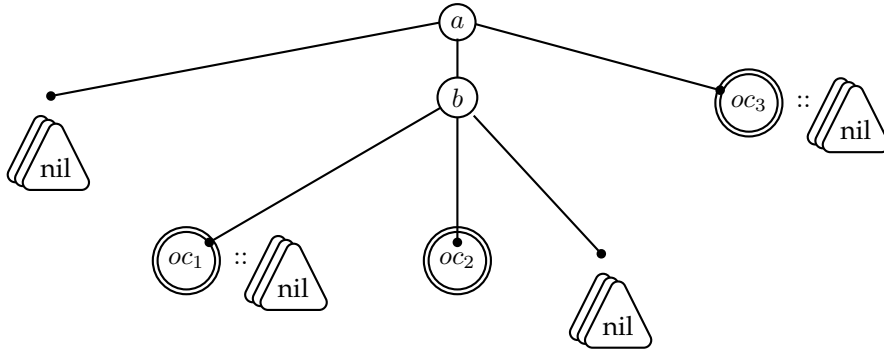
The major difference between the traditional formalism and the new formalism is as follows: In the traditional formalism, an internal node has one or arbitrarily many more children, which are strategy profiles, and in addition an internal node is linked to one and only one of its children by a double line. In the new formalism, an internal node has one left child, which is a list of strategy profiles, one central child, which is a strategy profile corresponding to the double line, and one right child, which is a list of strategy profiles. The next two pictures represent the same strategy profile in both formalisms, where for all i between 0 and n , the object s'_i is the translation of s_i from the traditional formalism into the new formalism.



For instance, let a and b be two agents, and oc_1 , oc_2 and oc_3 be three outcomes. Consider the following strategy profile in the traditional formalism.



The strategy profile above is represented by the following strategy profile in the new formalism.



The Inductive and Formal Way:

Formally in Coq, strategy profiles are defined as follows.

Inductive Strat : Set :=

| *sL* : Outcome → Strat

| *sN* : Agent → list Strat → Strat → list Strat → Strat.

sL stands for “strategy profile leaf” and *sN* for “strategy profile node”. If *oc* is an Outcome, then *sL oc* is a Strat. If *a* is an Agent, if *s* is a Strat, and if *l* and *l'* are lists of objects in Strat, then *sN a l s l'* is a Strat. The interpretation of such an object is as follows: as for sequential games, the agent *a* “owns” the root of *sN a l s l'*. Moreover, *s* is the substrategy profile *a* has decided the play shall continue in, and *l* and *l'* represent the options dismissed by *a* on the left and on the right of *s*. The structure ensures that any node of the tree meant to be internal has a non-zero number of children, and that one and only one child has been chosen. Like for sequential games, the inductive graphical formalism and the inductive Coq formalism are very close to each other for strategy profiles. For instance, compare the strategy profile *sN a nil (sN b oc1::nil oc2 nil) oc3::nil* with its representation in the inductive graphical formalism above.

4.6.2 Induction Principle for Strategy Profiles

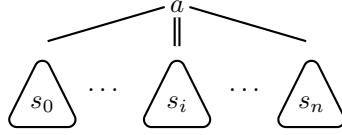
This subsection discusses what would be an inductive proof principle for strategy profiles in the traditional formalism. The failure of the former leads to an induction principle in the inductive Coq formalism.

The Traditional Way:

In the traditional formalism, a so-called induction principle would read as follows. In order to check that a predicate holds for all strategy profiles, check two properties: First, check that the predicate holds for all strategy profiles that are leaves (enclosing an outcome). Second, check that if the predicate holds for all the following strategy profiles,



then, for any agent a and any i between 0 and n , it holds for the following strategy profile.



However, dots, etc's and so on, very seldom suit formal proofs. While some dots may be easily formalised, some others cannot. In the inductive formalism proposed in this chapter, they are formalised by lists. So, an induction principle for games must take lists into account.

The Inductive and Formal Way:

The induction principle that Coq automatically associates to the inductively defined structure of strategy profiles is not efficient. A convenient principle has to be built (and hereby proved) manually, *via* a recursive function and the command *fix*. That leads to the following induction principle, which is therefore a theorem in Coq. In order to prove that a predicate P holds for all strategy profiles, it suffices to design a predicate Q expecting a strategy profile and a list of strategy profiles, and then to check that several fixed properties hold, as formally stated below.

$$\begin{aligned}
 \text{Strat_ind2} : & \forall (P : \text{Strat} \rightarrow \text{Prop}) (Q : \text{list Strat} \rightarrow \text{Strat} \rightarrow \text{list Strat} \rightarrow \text{Prop}), \\
 & (\forall oc, P (sL oc)) \rightarrow \\
 & (\forall sc, P sc \rightarrow Q \text{ nil } sc \text{ nil}) \rightarrow \\
 & (\forall s, P s \rightarrow \forall sc sr, Q \text{ nil } sc sr \rightarrow Q \text{ nil } sc (s :: sr)) \rightarrow \\
 & (\forall s, P s \rightarrow \forall sc sl sr, Q sl sc sr \rightarrow Q (s :: sl) sc sr) \rightarrow \\
 & (\forall sc sl sr, Q sl sc sr \rightarrow \forall a, P (sN a sl sc sr)) \rightarrow \\
 & \forall s, P s
 \end{aligned}$$

4.6.3 Underlying Game of a Strategy Profile

This subsection discusses the notion of underlying game of a given strategy profile and defines a function that returns such a game. One lemma follows the definition of the function in the Coq formalism.

The Traditional Way:

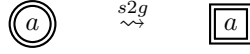
The example below explains what function is intended. It amounts to forgetting all the choices.



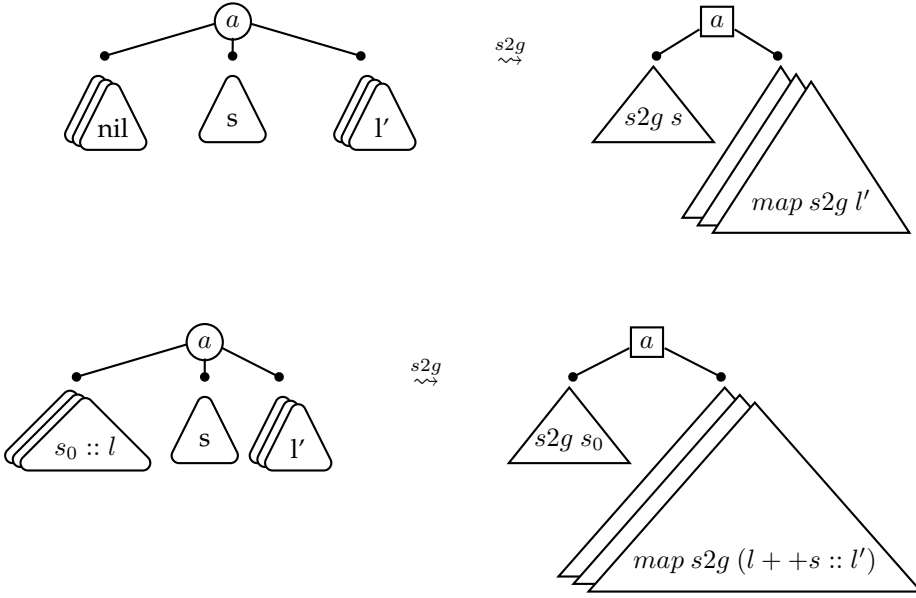
The Inductive and Graphical Way:

To prepare the full formalisation of the idea above, the intended function named $s2g$ (stands for strategy profile to game) is defined recursively with the inductive graphical formalism, in two steps along the structure of strategy profiles.

First step: the underlying game of a strategy profile where no choice has been made, *i.e.*, a leaf strategy profile, is a game where no choice is required, *i.e.*, a leaf game.



The second step needs case splitting along the structure of the first list, *i.e.*, whether it is empty or not.



The Inductive and Formal Way:

The intended function is inductively and formally defined in Coq, and called *s2g*, which stands for “strategy profile to game”.

```

Fixpoint s2g (s : Strat) : Game :=
  match s with
  | sL oc => gL oc
  | sN a sl sc sr =>
    match sl with
    | nil => gN a (s2g sc) (map s2g sr)
    | s'::sl' => gN a (s2g s') ((map s2g sl')++(s2g sc)::(map s2g sr))
    end
  end.

```

The next result states that if the two lists of substrategy profiles of two strategy profiles have, component-wise, the same underlying games, then the two original strategy profiles also have the same underlying game.

Lemma *map_s2g_sN_s2g* : $\forall a \text{ sl sc sr sl' sc' sr'}$,
 $\text{map s2g (sl ++ sc :: sr)} = \text{map s2g (sl' ++ sc' :: sr')} \rightarrow$
 $\text{s2g (sN a sl sc sr)} = \text{s2g (sN a sl' sc' sr')}.$

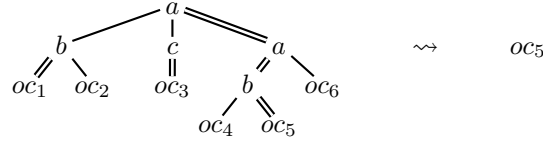
Proof Double case split on *sl* and *sl'* being empty and use *map_app*. \square

4.6.4 Induced Outcome of a Strategy Profile

This subsection discusses the notion of outcome induced by a strategy profile, and defines a function that computes such an outcome. Three lemmas follow the definition of the function in the Coq formalism.

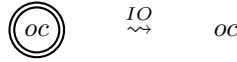
The Traditional Way:

Starting at the root of a strategy profile and following the agents' consecutive choices leads to a leaf, and hereby to an outcome. The following example explains what function is intended.

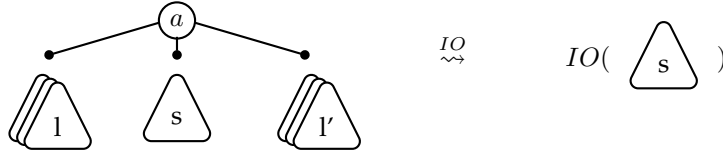


The Inductive and Graphical Way:

To prepare the full formalisation of the idea above, the intended function is defined inductively with the inductive graphical formalism, in two steps along the structure of strategy profiles. First step: the outcome induced by a leaf strategy profile is the enclosed outcome.



Second step: follow the choice at the root and consider the chosen substrategy profile.



The Inductive and Formal Way:

The intended function, called *InducedOutcome*, is recursively and formally defined in Coq.

```

Fixpoint InducedOutcome (s : Strat) : Outcome :=
  match s with
  | sL oc => oc
  | sN a sl sc sr => InducedOutcome sc
  end.

```

The following lemma, proved by induction on *sl*, states that the outcomes used by the underlying game of a strategy profile are all used by the underlying-

ing game of a bigger strategy profile whose chosen substrategy profile is the original strategy profile.

Lemma *UsedOutcomes_sN_incl* : $\forall a \text{ sl sc sr},$
 $\text{incl} (\text{UsedOutcomes} (s2g \text{ sc})) (\text{UsedOutcomes} (s2g (sN a \text{ sl sc sr}))).$

The next result says that the outcome induced by a strategy profile is also an outcome used in the underlying game.

Lemma *Used_Induced_Outcomes* : $\forall s : \text{Strat},$
 $\text{In} (\text{InducedOutcome } s) (\text{UsedOutcomes} (s2g s)).$

Proof Write the claim as a predicate on s and apply the induction principle *Strat_ind2* where $Q \text{ l s l'}$ is $\text{In} (\text{InducedOutcome } s) (\text{UsedOutcomes} (s2g s))$. For the last induction case invoke *UsedOutcomes_sN_incl*. \square

Note that if the underlying game of a strategy profile is a leaf game enclosing a given outcome, then the strategy profile is a leaf strategy profile enclosing the same outcome, so that the outcome induced by the strategy profile is also the same outcome: if $s2g \text{ s} = gL \text{ oc}$ then $\text{InducedOutcome } s = oc$. The following lemma is the list version of the remark above. It is prove by induction on ls .

Lemma *map_s2g_gL_InducedOutcome* : $\forall ls \text{ loc},$
 $\text{map } s2g \text{ ls} = \text{map} (\text{fun } y \Rightarrow gL \text{ y}) \text{ loc} \rightarrow \text{map } \text{InducedOutcome } ls = \text{loc}.$

4.7 Convertibility

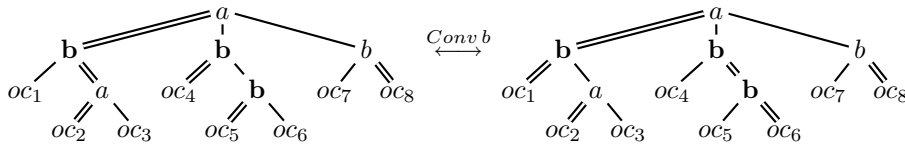
The first subsection defines convertibility, which will be used to formally define the notion of Nash equilibrium in section 4.8. The second subsection designs an inductive proof principle for convertibility.

4.7.1 Definition of Convertibility

This subsection discusses the notion of convertibility, which is the ability of an agent to convert a strategy profile into another strategy profile, and defines a predicate accounting for it. Four lemmas follow its definition in Coq.

The Traditional Way:

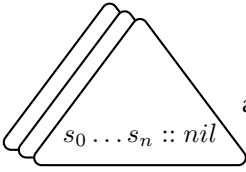
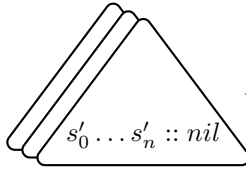
An agent is (usually implicitly) granted the ability to simultaneously change his choices at all nodes that he owns. The following example shows one possible such change: agent b changes his choices exactly at the nodes where b is displayed in bold font.





The Inductive and Graphical Way:

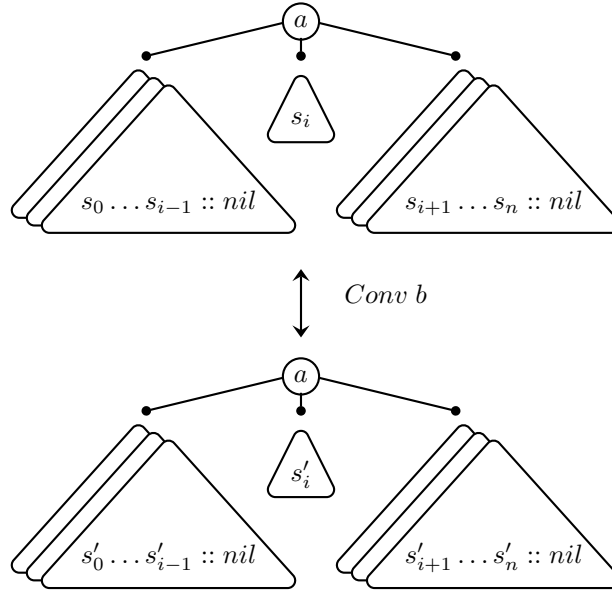
To prepare the full formalisation of the idea above, the intended function is defined inductively with the inductive graphical formalism, in two steps along the structure of strategy profiles. Let b be an agent. First step: Let oc be an outcome. Agent b can convert the leaf strategy profile enclosing oc into itself by changing some of (actually none of) his choices since no choice has been made.

$$\textcircled{oc} \xleftrightarrow{\text{Conv } b} \textcircled{oc}$$

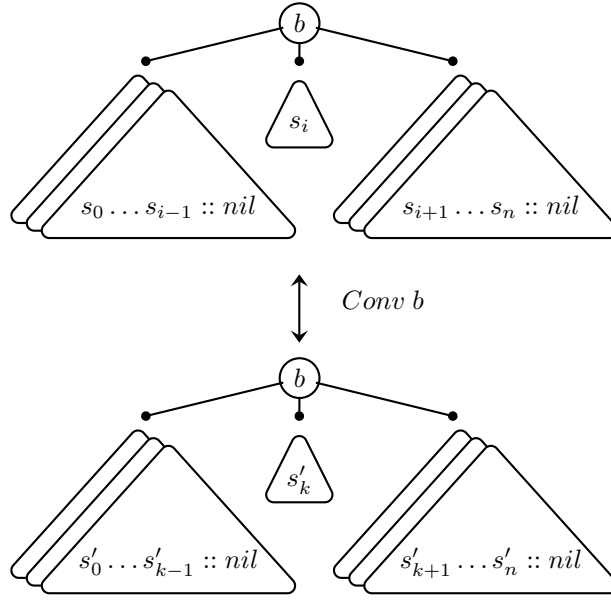
Second step: let  and  be two lists

of strategy profiles such that for all i between 0 and n ,  $\xleftrightarrow{\text{Conv } b}$ 

The second step involves compound strategy profiles, and needs case splitting on the “converting agent” owning the roots or not. Also let a be another agent. For all i between 0 and n , agent b can perform the following conversion by combining at once all his conversion abilities in all the substrategy profiles.



When a equals b , the agent b can also change his choice at the root owned by a . In that case, the agent b can perform the following conversion for all i and k between 0 and n .



The Inductive and Formal Way:

The following definition accounts for agents being able to unilaterally change part of their choices. *Conv*, which means strategy profile convertibility, and *ListConv*, which means component-wise convertibility of list of strategy profiles, are defined by mutual induction *via* the word *with*, within the same definition.

Inductive $Conv : Agent \rightarrow Strat \rightarrow Strat \rightarrow Prop :=$
 $| convLeaf : \forall b oc, Conv b (sL oc)(sL oc)$
 $| convNode : \forall b a sl sl' sc sc' sr sr', (length sl = length sl' \vee a = b) \rightarrow$
 $ListConv b (sl ++ (sc :: sr)) (sl' ++ (sc' :: sr')) \rightarrow$
 $Conv b (sN a sl sc sr)(sN a sl' sc' sr')$

with

ListConv $: Agent \rightarrow list Strat \rightarrow list Strat \rightarrow Prop :=$
 $| lconvnil : \forall b, ListConv b nil nil$
 $| lconvcons : \forall b s s' tl tl', Conv b s s' \rightarrow ListConv b tl tl' \rightarrow$
 $ListConv b (s :: tl)(s' :: tl')$

The formula $length sl = length sl' \vee a = b$ ensures that only the owner of a node can change his choice at that node. It corresponds to the case splitting in the inductive and graphical definition above. $ListConv b (sl ++ (sc :: sr)) (sl' ++ (sc' :: sr'))$ guarantees that this property holds also in the substrategy profiles. *ListConv* corresponds to the dots in the inductive and graphical definition above²

²One may think of avoiding a mutual definition for *Conv* by using the already defined *rel_vector* instead of *ListConv*. However, the Coq versions 8.0 and 8.1 do not permit it, presumably because it would require to pass the whole *Conv* object as an argument (to *rel_vector*) while not yet fully defined.

The following two lemmas establish the equivalence between *ListConv* on the one hand and *rel_vector* and *Conv* on the other hand. They are both proved by induction on the list l and by a case splitting on l' .

Lemma *ListConv_rel_vector* : $\forall a \ l \ l',$
ListConv a l l' \rightarrow rel_vector (Conv a) l l'.

Lemma *rel_vector_ListConv* : $\forall a \ l \ l',$
rel_vector (Conv a) l l' \rightarrow ListConv a l l'.

The next two results state reflexivity property of *Conv* and *ListConv*.

Lemma *Conv_refl* : $\forall a \ s, \text{Conv } a \ s \ s.$

Proof Let a be an agent and s a strategy profile. It suffices to prove that $P \ s$ where $P \ s'$ is *Conv a s' s'* by definition. Apply the induction principle *Strat_ind2* where $Q \ s \ l \ s \ r$ is *ListConv a (sl++sc::sr) (sl++sc::sr)* by definition. Checking all cases is straightforward. \square

Lemma *ListConv_refl* : $\forall a \ l, \text{ListConv } a \ l \ l.$

Proof By *rel_vector_ListConv*, induction on the list l , and *Conv_refl*. \square

4.7.2 Induction Principle for Convertibility

The Inductive and Formal Way:

A suitable induction principle for convertibility is automatically generated in Coq with the command *Scheme*, and hereby is a theorem in Coq. The principle states that under four conditions, *Conv a* is a subrelation of the binary relation $P \ a$ for all agents a . Note that in order to prove such a property, the user has to imagine a workable predicate $P0$.

conv_lconv_ind : $\forall (P : \text{Agent} \rightarrow \text{Strat} \rightarrow \text{Strat} \rightarrow \text{Prop})$
 $(P0 : \text{Agent} \rightarrow \text{list Strat} \rightarrow \text{list Strat} \rightarrow \text{Prop}),$
 $(\forall b \ oc, P \ b \ (sL \ oc) \ (sL \ oc)) \rightarrow$
 $(\forall b \ a \ sl \ sl' \ sc \ sc' \ sr \ sr', \text{length } sl = \text{length } sl' \vee a = b \rightarrow \text{ListConv } b \ (sl \ ++ \ sc :: sr) \ (sl' \ ++ \ sc' :: sr') \rightarrow P0 \ b \ (sl \ ++ \ sc :: sr) \ (sl' \ ++ \ sc' :: sr') \rightarrow P \ b \ (sN \ a \ sl \ sc \ sr) \ (sN \ a \ sl' \ sc' \ sr')) \rightarrow$
 $(\forall b, P0 \ b \ nil \ nil) \rightarrow$
 $(\forall b \ s \ s' \ tl \ tl', \text{Conv } b \ s \ s' \rightarrow P \ b \ s \ s' \rightarrow \text{ListConv } b \ tl \ tl' \rightarrow P0 \ b \ tl \ tl' \rightarrow P0 \ b \ (s :: tl) \ (s' :: tl')) \rightarrow$
 $\forall a \ s \ s0, \text{Conv } a \ s \ s0 \rightarrow P \ a \ s \ s0$

This induction principle is used below to prove that two convertible strategy profiles have the same underlying game.

Lemma *Conv_s2g* : $\forall (a : \text{Agent})(s \ s' : \text{Strat}), \text{Conv } a \ s \ s' \rightarrow s2g \ s = s2g \ s'.$

Proof Assume s and s' convertible by a . Write $s2g \ s = s2g \ s'$ as $P \ a \ s \ s'$ for some P and proceed by the induction principle *conv_lconv_ind* where $P0 \ b \ l \ l'$ is $(\text{ListConv } b \ l \ l' \rightarrow \text{map } s2g \ l = \text{map } s2g \ l')$ by definition. The remainder invokes *map_s2g_sN_s2g*. \square

4.8 Concepts of Equilibrium

This section defines the notions of preference, happiness, Nash equilibrium, and subgame perfect equilibrium in the new and abstract formalism. Two lemmas follow these definitions.

In traditional game theory, the agents implicitly prefer strictly greater pay-offs, and thus also prefer payoff functions granting them strictly greater pay-offs. In the abstract formalism, the agents' preferences for outcomes are explicitly represented by binary relations over the outcomes, one relation per agent. Below, $OcPref\ a$ is the preference of agent a .

Variable $OcPref : Agent \rightarrow Outcome \rightarrow Outcome \rightarrow Prop$.

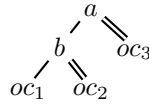
Since every strategy profile induces an outcome, the preferences over outcomes yield preferences over strategy profiles. For instance, if agent a prefers oc_1 to oc_3 , then he prefers the following right-hand strategy profile to the left-hand one. As to agent b , it could be either way since no specific property is assumed about preferences.



Formally in Coq, preference over strategy profiles is defined as follows.

Definition $StratPref\ (a : Agent)(s\ s' : Strat) : Prop :=$
 $OcPref\ a\ (InducedOutcome\ s)(InducedOutcome\ s')$.

If an agent cannot convert a given strategy profile to any preferred one, then he is said to be happy with respect to the given strategy profile. For instance the following strategy profile makes agent a happy iff agent a does not prefer oc_2 to oc_3 , whatever his other preferences are.



Formally in Coq, happiness of an agent is defined as follows.

Definition $Happy\ (s : Strat)(a : Agent) : Prop := \forall\ s',$
 $Conv\ a\ s\ s' \rightarrow \neg StratPref\ a\ s\ s'.$

A strategy profile that makes every agent happy is called a Nash equilibrium.

Definition $Eq\ (s : Strat) : Prop := \forall\ a, Happy\ s\ a.$

A subgame perfect equilibrium is a Nash equilibrium such that all of its substrategy profiles are subgame perfect equilibria. Compare this definition with the informal and more complicated one in section 4.2.

Fixpoint $SPE\ (s : Strat) : Prop := Eq\ s \wedge$
 $match\ s\ with$
 $| sL\ oc \Rightarrow True$
 $| sN\ a\ sl\ sc\ sr \Rightarrow (listforall\ SPE\ sl) \wedge SPE\ sc \wedge (listforall\ SPE\ sr)$
 $end.$

Therefore, a subgame perfect equilibrium is a Nash equilibrium.

Lemma *SPE_is_Eq* : $\forall s : \text{Strat}, \text{SPE } s \rightarrow \text{Eq } s$.

The following provides a sufficient condition for a strategy profile to be a Nash equilibrium: at the root of a compound strategy profile, if the chosen substrategy profile is a Nash equilibrium, and if the owner of the root cannot convert any of his other options into a substrategy profile that he prefers to his current choice, then the compound strategy profile is also a Nash equilibrium. This is stated in Coq below.

Lemma *Eq_subEq_choice* : $\forall a \text{ sl } sc \text{ sr},$
 $(\forall s \text{ s}', \text{In } s \text{ sl} \vee \text{In } s \text{ sr} \rightarrow \text{Conv } a \text{ s s}' \rightarrow \neg \text{StratPref } a \text{ sc s}') \rightarrow$
 $\text{Eq } sc \rightarrow \text{Eq } (sN \text{ a sl } sc \text{ sr}).$

Proof Let be $a, \text{sl}, \text{sc}, \text{sr}$ and assume the two premises. Also assume that an agent a' can convert $sN \text{ a sl } sc \text{ sr}$ to s' that he prefers. Now try to derive a contradiction from the hypothesis. For this, note that s' equals $sN \text{ a sl' } sc' \text{ sr'}$ for some $\text{sl}', \text{sc}',$ and sr' . Case split on sl and sl' having or not the same length. If they have the same length then use the equilibrium assumption together with *rel_vector_app_cons_same_length* and *ListConv_rel_vector*. If sl and sl' have different lengths then a' equals a . The remainder of the proof invokes *rel_vector_app_cons_different_length* and *ListConv_rel_vector*. \square

The converse of this lemma also holds, but it is omitted in this chapter.

4.9 Existence of Equilibria

This section generalises the notion of “backward induction” for abstract sequential games, and shows that it yields a subgame perfect equilibrium when preferences are totally ordered. But it also shows that it may not always yield a subgame perfect equilibrium for arbitrary preferences. However, this section eventually proves that acyclicity of (decidable) preferences is a necessary and sufficient condition for guaranteeing (computable) existence of Nash equilibrium/subgame perfect equilibrium.

4.9.1 “Backward Induction”

This subsection starts with an informal discussion, and continues with definitions in the inductive graphical formalism. Eventually, a “backward induction” function is defined in Coq, and one lemma follows.

The Traditional Way:

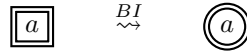
Informally, the idea is to perform “backward induction” on all subgames first, and then to let the owner of the root choose one strategy profile that suits him best among the newly built strategy profiles. When preferences are partially ordered, the agent can choose in order to maximise his preference; when they are not partially ordered, a procedure slightly more general may be needed. Lemma *Choose_and_split* defined in subsection 4.4.4 relates to such a procedure. (More specifically, the proof of the lemma is such a procedure.) For instance, let $oc_1 \dots oc_6$ be six outcomes such that a prefers oc_5 to oc_2 , and b prefers oc_2

to oc_1 and oc_1 to oc_2 , and nothing else. A “backward induction” process is detailed below. On the left-hand picture, b chooses by *Choose_and_split*. On the right-hand picture, agent a being “aware” of b ’s choice procedure chooses accordingly, also by *Choose_and_split*.

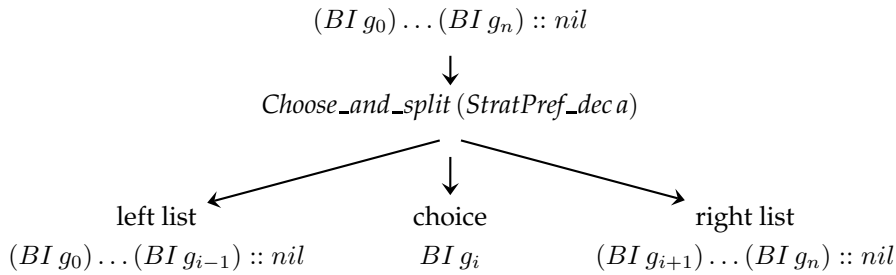


The Inductive and Graphical Way:

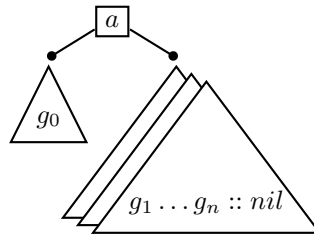
To prepare the full formalisation of the idea above, the intended function is defined inductively with the inductive graphical formalism, in two steps along the structure of strategy profiles. First step: performing “backward induction” on a leaf game that encloses an outcome yields a leaf strategy profile that encloses the same outcome.



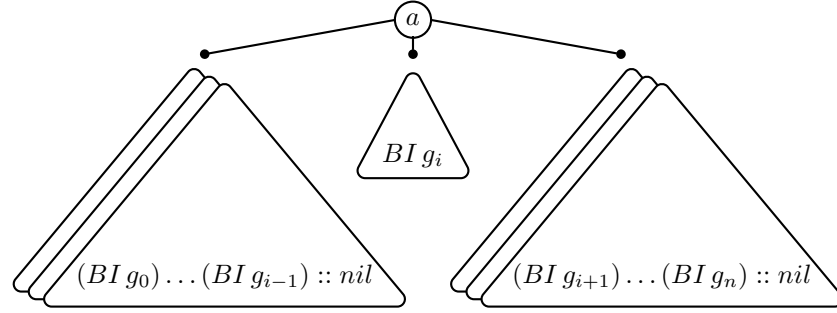
Second step: Assume that “backward induction” is defined for the games g_0, \dots, g_n . An agent a can choose one strategy profile among $BI\ g_0, \dots, BI\ g_n$ by *Choose_and_split* and his own preference, as below.



Then, “backward induction” can be defined on the following compound game.



↓ BI



The Inductive and Formal Way:

The definition using the inductive graphical formalism above is translated into Coq formalism. First, assume that preferences over outcomes are decidable.

Hypothesis *OcPref_dec* : $\forall (a : \text{Agent}), \text{rel_dec } (\text{OcPref } a)$.

Subsequently, preferences over strategy profiles are also decidable.

Lemma *StratPref_dec* : $\forall (a : \text{Agent}), \text{rel_dec } (\text{StratPref } a)$.

Next, the generalisation of “backward induction”, with respect to the preferences above, is defined by recursion. For the sake of readability, the definition displayed below is a slight simplification of the actual Coq code.

```

Fixpoint BI (g : Game) : Strat :=
match g with
| gL oc  $\Rightarrow$  sL oc
| gN a g l  $\Rightarrow$  let (sl,sc,sr):=
    Choose_and_split (StratPref_dec a) (map BI l) (BI g) in
    sN a sl sc sr
end.
```

As stated below, the underlying game of the image by *BI* of a given game is the same game.

Lemma *BI_s2g* : $\forall g : \text{Game}, \text{s2g } (\text{BI } g) = g$.

Proof Rewrite the claim as $\forall g, P g$ for some *P*. Apply the induction principle *Game_ind2* where *Q g l* is $\text{s2g } (\text{BI } g) = g \wedge \text{map s2g } (\text{map BI } l) = l$. For the last induction case, invoke *Choose_and_split* and *map_app*. \square

4.9.2 The Total Order Case

In this subsection only, assume transitivity and irreflexivity of preferences over outcomes. Subsequently, those properties also hold for preferences over strategy profiles.

Hypothesis *OcPref_irrefl* : $\forall (a : \text{Agent}), \text{irreflexive } (\text{OcPref } a)$.

Hypothesis *OcPref_trans* : $\forall (a : \text{Agent}), \text{transitive } (\text{OcPref } a)$.

Lemma *StratPref_irrefl* : $\forall (a : \text{Agent}), \text{irreflexive } (\text{StratPref } a)$.

Lemma *StratPref_trans* : $\forall (a : \text{Agent}), \text{transitive } (\text{StratPref } a)$.

Irreflexivity of preferences guarantees that leaf strategy profiles are Nash equilibria.

Lemma *Leaf_Eq* : $\forall oc : Outcome, Eq (sL oc)$.

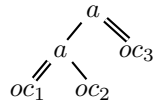
If preferences are total over a given list of outcomes, then for any sequential game using only outcomes from the list, “backward Induction” yields subgame perfect equilibrium. This is the translation of Kuhn’s result into abstract sequential game formalism.

Lemma *BI_SPE* : $\forall loc, (\forall (a : Agent), total (OcPref a) loc) \rightarrow \forall g : Game, incl (UsedOutcomes g) loc \rightarrow SPE (BI g)$.

Proof Assume *loc*, a list of outcomes, and the totality property. Write *incl (UsedOutcomes g) loc* \rightarrow *SPE (BI g)* as *P g* and proceed by the induction principle *Game_ind2* where $Q g l$ is *listforall (fun g' \Rightarrow incl (UsedOutcomes g') loc) (g::l) \rightarrow SPE (BI g) \wedge listforall SPE (map BI l)*. The first three cases are straightforward. For the fourth and last case assume *g*, *l*, and the other premises, such as an agent *a*. In order to prove that the “backward induction” of the compound game *gN a g l* is a subgame perfect equilibrium, first note that all substrategy profiles of this “backward induction” are subgame perfect equilibria, by the induction hypotheses, *listforall_appr* and *listforall_appr*. Then, the main difficulty is to prove that it is a Nash equilibrium. For this, invoke *Eq_subEq_choice* after proving its two premises: The induction hypothesis and lemma *SPE_is_Eq* shows that the substrategy profile chosen by *a* is a Nash equilibrium. For the other premise required for invoking *Eq_subEq_choice*, assume a substrategy profile not chosen by *a*. The induction hypothesis and lemma *SPE_is_Eq* show that it is a Nash equilibrium. Next, assume that this Nash equilibrium is convertible by *a* into another strategy profile, and show that *a* does not prefer this new strategy profile to his current choice. For this, invoke decidability, irreflexivity, transitivity, and totality of preferences, as well as lemmas *listforall_In*, *UsedOutcomes_gN*, *map_inverse*, *Used_Induced_Outcomes* and *Conv_s2g*. \square

4.9.3 Limitation

Now, no property is assumed about the preferences. Consider the three outcomes *oc*₁, *oc*₂ and *oc*₃, and an agent *a* that prefers *oc*₂ to *oc*₃, and nothing else. The strategy profile below is obtained by the “backward induction” function define in subsection 4.9.1. However, it is not a Nash equilibrium since the current induced outcome is *oc*₃, but it can be converted by *a* into the preferred *oc*₂.



Informally, when dealing with totally ordered preferences, the notions of “backward induction” and subgame perfect equilibrium coincide although their definitions differ; they are the same in extension, but are different in intension. This difference in intension is critical when dealing with partially ordered preferences: as shown by the example above, “backward induction” no longer yields Nash equilibrium, let alone subgame perfect equilibrium.

4.9.4 General Case

Until this subsection, equilibrium concepts and related notions have been defined with respect to given preferences: a preference binary relation was associated with an agent once for all. In the following, equilibrium concepts and related notions are abstracted over preferences. It means that preferences become a parameter of the definitions and lemmas. For instance, instead of writing $Eq\ s$ to say that s is a Nash equilibrium, one shall write $Eq\ OcPref\ s$ to say that s is a Nash equilibrium with respect to the family of preferences defined by $OcPref$.

As formally stated in the two lemmas below, given two families of preferences and given a strategy profile, if for every agent the restriction of his first preference to the outcomes used by the strategy profile is a subrelation of his second preference, and if the strategy profile is a Nash equilibrium/subgame perfect equilibrium with respect to the second preferences then it is also a Nash equilibrium/subgame perfect equilibrium with respect to the first preferences. Informally, the less arcs an agent's preference has, the more strategy profiles make the agent happy.

Lemma $Eq_order_inclusion : \forall OcPref\ OcPref'\ s,$
 $(\forall a : Agent,$
 $sub_rel\ (restriction\ (OcPref\ a)\ (UsedOutcomes\ (s2g\ s)))\ (OcPref'\ a)) \rightarrow$
 $Eq\ OcPref'\ s \rightarrow Eq\ OcPref\ s.$

Proof Invoke $Used_Induced_Outcomes$ and $Conv_s2g$. □

Lemma $SPE_order_inclusion : \forall OcPref\ OcPref'\ s,$
 $(\forall a : Agent,$
 $sub_rel\ (restriction\ (OcPref\ a)\ (UsedOutcomes\ (s2g\ s)))\ (OcPref'\ a)) \rightarrow$
 $SPE\ OcPref'\ s \rightarrow SPE\ OcPref\ s.$

Proof Assume two families of preferences and rewrite the claim as $\forall s, P\ s$ for some P . Then apply $Strat_ind2$ where $Q\ sl\ sc\ sr$ is $(\forall a, sub_rel\ (restriction\ (OcPref\ a)\ (UsedOutcomes\ (s2g\ (sN\ a\ sl\ sc\ sr))))\ (OcPref'\ a)) \rightarrow listforall\ (SPE\ OcPref')\ sl \rightarrow SPE\ OcPref'\ sc \rightarrow listforall\ (SPE\ OcPref')\ sr \rightarrow listforall\ (SPE\ OcPref)\ sl \wedge SPE\ OcPref\ sc \wedge listforall\ (SPE\ OcPref)\ sr$. For the first and fifth induction steps, apply $Eq_order_inclusion$. For the third and fourth induction step, invoke lemmas $transitive_sub_rel$, $sub_rel_restriction_incl$, $incl_appr$, $incl_appl$, and $incl_refl$. □

The following lemma generalises Kuhn's result to acyclic preferences (instead of totally ordered). It invokes a result related to topological sorting proved in [28] and chapter 3.

Theorem $acyclic_SPE : \forall OcPref,$
 $(\forall a, rel_dec\ (OcPref\ a)) \rightarrow$
 $(\forall a, irreflexive\ (clos_trans\ Outcome\ (OcPref\ a))) \rightarrow$
 $\forall g, \{s : Strat \mid s2g\ s = g \wedge SPE\ OcPref\ s\}.$

Proof Assume a family of decidable acyclic preferences. Let g be a game. According to [28] and chapter 3, by topological sorting there exists a family of decidable acyclic preferences that are strict total orders including the original preferences on the outcomes used by g . A subgame perfect equilibrium can be computed by BI_SPE . Conclude by $SPE_order_inclusion$. □

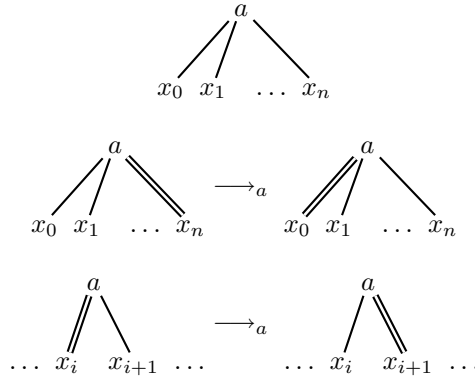
The following property relates to SPE_is_Eq .

Theorem SPE_Eq : $\forall OcPref,$
 $(\forall a, rel_dec (OcPref a)) \rightarrow$
 $(\forall g : Game, \{s : Strat \mid s2g s=g \wedge SPE OcPref s\}) \rightarrow$
 $\forall g : Game, \{s : Strat \mid s2g s=g \wedge Eq OcPref s\}.$

The next result says that if all games have Nash equilibria with respect to a given family of preferences, then these preferences are acyclic.

The Traditional Way

Informally, let an agent a prefer x_1 to x_0 , x_2 to x_1 , and so on, and x_0 to x_n . The game displayed below has no Nash equilibrium, as suggested graphically. The symbol $s \rightarrow_a s'$ means that agent a both prefers s' to s and can convert s to s' . So, the formula $s \rightarrow_a s'$ witnesses the agent's non-happiness.



The Formal Way

Now, the corresponding formal statement and its proof.

Theorem $Eq_acyclic$: $\forall OcPref,$
 $(\forall a, rel_dec (OcPref a)) \rightarrow$
 $(\forall g : Game, \{s : Strat \mid s2g s=g \wedge Eq OcPref s\}) \rightarrow$
 $\forall a : Agent, irreflexive (clos_trans Outcome (OcPref a)).$

Proof Assume all premises. In particular, let a be an agent and oc be an outcome related to itself by the transitive closure of the agent's preference. Prove a contradiction by building a game such that every strategy profile for the game can be improved upon, as follows. By lemma $clos_trans_path$, get an actual path loc from oc to itself with respect to the preference. If loc is empty then invoke lemma $Conv_refl$. If loc is not empty then, thanks to the assumption, compute a Nash equilibrium for the game with root owned by agent a and the children being leaves enclosing oc for the first and the elements of loc for the others. Case split on the right-hand substrategy profile list of the Nash equilibrium being empty. If it is empty, and the left-hand substrategy profile list of the Nash equilibrium as well, then the root of the game has two or more children and the strategy profile has one only, hence a contradiction. If the left-hand substrategy profile list is not empty then $ListConv_refl, map_s2g_gL_InducedOutcome,$

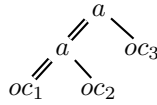
path_app_elim_right, and *map_app* may be needed. If the right-hand substrategy profile list of the Nash equilibrium is not empty then invoke associativity of list appending as well as the lemmas mentioned just above, and a case splitting on the left-hand substrategy profile list again. \square

4.9.5 Examples

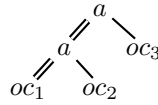
Partial Order and Subgame Perfect Equilibrium

As in section 4.9.3, consider three outcomes oc_1 , oc_2 and oc_3 , and an agent a that only prefers oc_2 to oc_3 . Each of the three possible linear extensions of a 's preference, prior to "backward induction", leads to a subgame perfect equilibrium as shown below, where the symbol $<$ represents the different linear extensions. Compare with the "backward induction" without prior linear extension of subsection 4.9.3.

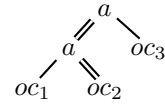
$$oc_2 < oc_3 < oc_1$$



$$oc_3 < oc_2 < oc_1$$



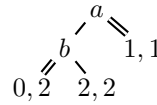
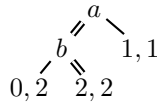
$$oc_3 < oc_1 < oc_2$$



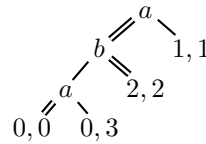
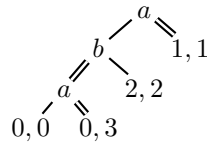
In the same way, the results obtained in this section show that subgame perfect equilibrium exists for games with preferences discussed in section 4.3.

Benevolent and Malevolent Selfishness

In subsection 4.3.1 were defined benevolent and malevolent selfishnesses. It is not surprising that, sometimes, benevolent selfishness yields better payoffs for all agents than malevolent selfishness. For instance compare the induced payoff functions below. The lefthand "backward induction" corresponds to benevolent selfishness, and the righthand one to malevolent selfishness.



On the contrary, malevolent selfishness may yield better payoffs for all agents than benevolent selfishness, as shown below. The lefthand "backward induction" corresponds to benevolent selfishness, and the righthand one to malevolent selfishness.



4.10 Conclusion

This chapter introduces a new inductive formalism for sequential games. It also replaces real-valued payoff functions with atomic objects called outcomes, and the usual total order over the reals with arbitrary preferences. This way, it also defines an abstract version of sequential games, with similar tree structure and notion of convertibility as for traditional sequential games. The notions of Nash equilibrium, subgame perfect equilibrium, and “backward induction” are translated into the new formalism. When preferences are totally ordered, “backward induction” guarantees existence of subgame perfect equilibrium for all sequential games, thus translating Kuhn’s result in the new formalism. However, an example shows that “backward induction” fails to provide equilibrium for non-totally ordered preferences, *e.g.*, partial orders. But when preferences are acyclic, it is still possible to perform “backward induction” on a game whose preferences have been linearly extended. This yields a subgame perfect equilibrium of the game with respect to the acyclic preferences, because removing arcs from a preference relation amounts to removing reasons for being unhappy. So, given a collection of outcomes, the following three propositions are equivalent:

- The agents’ preferences over the given outcomes are acyclic.
- Every game built over the given outcomes has a Nash equilibrium.
- Every game built over the given outcomes has a subgame perfect equilibrium.

The formalism introduced in this chapter is suitable for proofs in Coq, which is a (highly reliable) constructive proof assistant. This way, the above-mentioned equivalence is fully computer-certified using Coq. Beside the additional guarantee of correctness provided by the Coq proof, the activity of formalisation also provides the opportunity to clearly identify the useful definitions and the main articulations of the proof.

Informally, a result due to Aumann states that, when dealing with traditional sequential games, “common knowledge of rationality among agents” is equivalent to “backward induction” (where “rationality” means playing in order to maximise one’s payoff). This is arguable in abstract sequential games if one expects “common knowledge of rationality among agents” to imply Nash equilibrium. Indeed, “backward induction” may not imply Nash equilibrium, as seen in this chapter. Therefore, “backward induction” may not imply “common knowledge of rationality among agents”. Instead, one may wonder whether “common knowledge of rationality among agents” is equivalent to subgame perfect equilibrium, whatever it may mean, when preferences are acyclic, *i.e.* rational in some sense. In this case, the difference between “backward induction” and subgame perfect equilibrium seems critical again.

Chapter 5

Acyclic Preferences and Nash Equilibrium Existence: Another Proof of the Equivalence

Abstract sequential games are a class of games that generalises traditional sequential games. Basically, abstract atomic objects, called outcomes, replace the traditional real-valued payoff functions, and arbitrary binary relations over outcomes, called preferences, replace the usual total order over the reals. In 1953, Kuhn proved that every traditional sequential game has a Nash equilibrium. In 1965, Selten introduced refined Nash equilibria named subgame perfect equilibria. Lately, Le Roux proved (using the proof assistant Coq) that the following three propositions are equivalent: 1) Preferences over the outcomes are acyclic. 2) Every abstract sequential game has a Nash equilibrium. 3) Every abstract sequential game has a subgame perfect equilibrium. Part of the proof is akin to Kuhn's proof. This chapter proves that propositions 1) and 2) are equivalent, which is a corollary of the above-mentioned result. However, the proof presented in this chapter adopts a significantly different technique compared to the previous ones.

5.1 Introduction

5.1.1 Contribution

This chapter proves that, when dealing with abstract sequential games, the following two propositions are equivalent: 1) Preferences over the outcomes are acyclic. 2) Every sequential game has a Nash equilibrium. This is a corollary of the triple equivalence proved in [29] and chapter 4, however the new proof invokes neither structural induction on games, nor “backward induction”, nor topological sorting. Therefore, this alternative argument is of proof theoretical interest. The proof of the implication $1) \Rightarrow 2)$ invokes three main arguments. First, if preferences are strict partial orders, then every game has a Nash equi-

librium, by induction on the size of the game and a few cut-and-paste tricks on smaller games. Second, if a binary relation is acyclic, then its transitive closure is a strict partial order. Third, smaller preferences generate more equilibria, as seen in [29] and chapter 4. The converse $2) \Rightarrow 1)$ is proved as in [29] and chapter 4. It is worth noting that in [29] and chapter 4, the implication $1) \Rightarrow 2)$ follows $1) \Rightarrow 3)$ and $3) \Rightarrow 2)$. However, the proof of this chapter is direct: it does not involve subgame perfect equilibria.

5.1.2 Contents

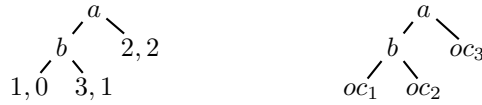
Section 5.2 defines the required concepts and section 5.3 proves the claim.

5.2 Preliminaries

First, this section briefly revisits some concepts that were formally defined in [29] and chapter 4. Then, it defines plays and induced plays.

Games

Consider a collection of outcomes that are the possible end-of-the-game situations, and a collection of agents that are the possible stake-holders and decision makers. Roughly speaking, an abstract sequential game is a traditional sequential game where each real-valued payoff function (enclosed in a leaf) has been replaced by an outcome. Below, the left-hand picture represents a traditional sequential game, and the right-hand picture represents an abstract sequential game on the same tree structure.



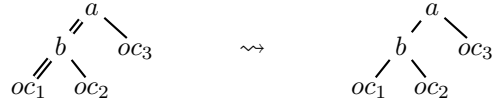
Strategy Profile

Roughly speaking, an abstract strategy profile is a traditional strategy profile where each real-valued payoff function (enclosed in a leaf) has been replaced by an outcome. Below, the left-hand picture represents a traditional strategy profile, and the right-hand picture represents an abstract strategy profile on the same tree structure.



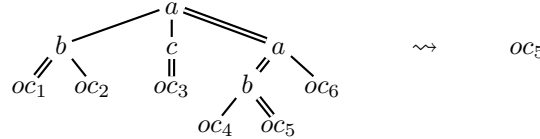
Underlying Game of a Strategy Profile

The underlying game of a strategy profile is obtained by forgetting all the choices made in the strategy profile, as suggested by the example below.



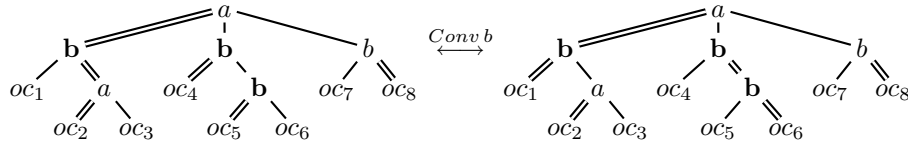
Induced Outcome

Starting at the root of a strategy profile and following the agents' consecutive choices leads to a leaf, and hereby to an outcome. This outcome is the induced outcome of the strategy profile.



Convertibility

An agent is granted the ability to change his choices at all nodes he owns, as shown in the following example. Agent b can change his choices at the nodes where b is displayed in bold font, and only at those nodes.



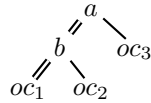
Nash Equilibrium

Definition 24 (Preferences) The preferences (of an agent) is a binary relation over the outcomes.

Definition 25 (Happiness) If an agent cannot convert a given strategy profile into another strategy profile that he prefers, he is said to be happy with the given strategy profile (and with respect to his preferences).

Definition 26 (Nash equilibrium) A Nash equilibrium (with respect to given preferences) is a strategy profile that make all agents happy (with respect to the same preferences).

Consider the following strategy profile whose induced outcome is oc_1 .



Assume that the preferences are partially defined by $oc_1 \prec_b oc_2$, then the strategy profile above is not a Nash equilibrium since b can convert it in a strategy profile that he prefers.

Assume that preferences are fully defined by $oc_1 \prec_a oc_2$, $oc_1 \prec_b oc_3$, and $oc_2 \prec_b oc_1$, then the strategy profile above is a Nash equilibrium.

Plays

The following concept was not defined in [29] or chapter 4, but it is needed in this chapter.

Definition 27 (Play, induced play) *A play of a game is a path from the root of the game to one of its leaves. Given a strategy profile s , the play induced by s is the path defined as follows: start from the root of s and follow the consecutive choices made by the agents owning the encountered nodes, until a leaf is encountered.*

5.3 The Proof

This section proves the following theorem.

Theorem 28 *The following two propositions are equivalent.*

1. *Preferences over the outcomes are acyclic.*
2. *Every abstract sequential game has a Nash equilibrium.*

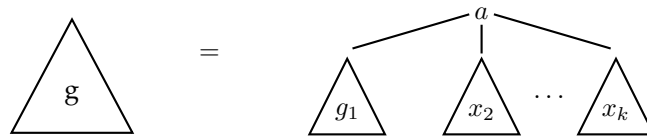
The more “interesting” implication is $1) \Rightarrow 2)$. The following lemma copes with the core of it. Then, the lemma is combined with other results in order to show the theorem.

Lemma 29 *If the agents’ preferences over a collection of outcomes are strict partial orders, then any sequential game built over these outcomes has a Nash equilibrium with respect to those preferences.*

Proof Assume that preferences are strict partial orders, *i.e.* transitive and irreflexive binary relations. For a given natural number n , consider the following property: “Any sequential game with at most n nodes has a Nash equilibrium”. The following proves by induction on n that the property holds for any n , which thereby proves the theorem.

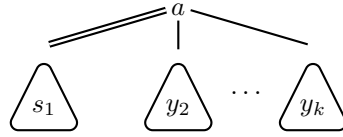
If $n = 1$, then any game with at most n nodes has one node exactly. Such a game is a leaf and the only strategy profile over that leaf is a Nash equilibrium. Indeed, by irreflexivity of the preferences, no agent prefers the outcome (enclosed in the leaf) to itself.

Now assume that the property holds for a given n greater than or equal to 1, and let g be a game with $n + 1$ nodes. Since g has 2 or more nodes, it is not a leaf. Let a be the agent owning the root of g and consider g_1 a subgame of g , say the leftmost one. The game g can therefore be decomposed as follows, where g_1 may actually be the only child of g , *i.e.* where $(x_i)_{2 \leq i \leq k}$ may actually be an empty family.



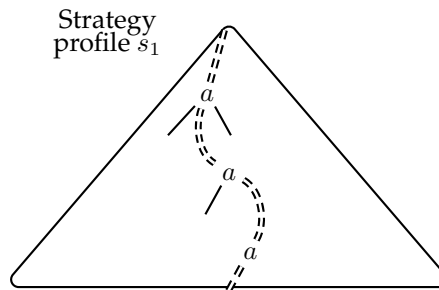
The game g_1 has at most n nodes, so by induction hypothesis, there exists s_1 a Nash equilibrium for g_1 . Now a case splitting is required. First case, for

any subgame x_i , there exists a corresponding strategy profile y_i that agent a cannot convert to a strategy profile that he prefers to s_1 . The case-splitting first assumption is a sufficient condition for the strategy profile displayed below to be a Nash equilibrium.



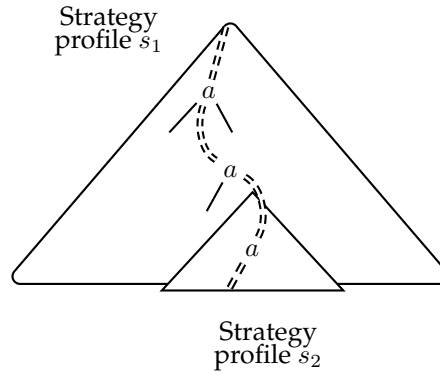
Second case, for any collection of strategy profiles whose underlying games are $x_2 \dots x_k$, agent a can convert one of these strategy profiles to a strategy profile that he prefers to s_1 . This implies two things. First, g has two or more subgames. Second, there exists a x_i among the $x_2 \dots x_k$ such that agent a can convert any strategy profile whose underlying game is x_i to a strategy profile he prefers to s_1 .

Now, a specific node in g is defined by case-splitting and named “pivot”. On the play induced by s_1 , some nodes may both be owned by agent a and have two or more children. If there is no such node, the pivot is defined as the root of g . If there are such nodes, consider the one that is farthest from the root. This node in s_1 corresponds to a node in g : the pivot. Below, an example shows three nodes owned by agent a on the play induced by s_1 . The third node has only one child, and the second node has two children, so the second node owned by a defines the pivot.



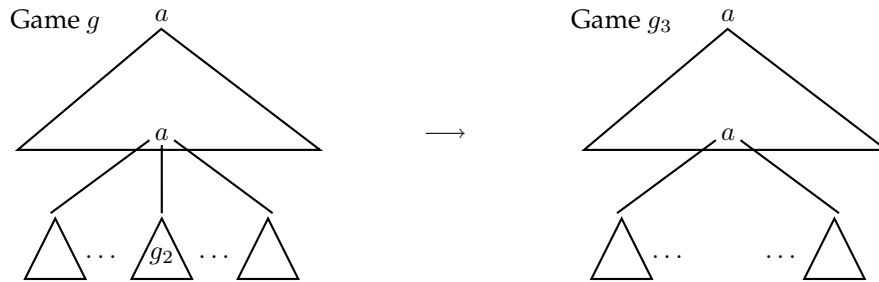
In both cases, the pivot is not a leaf since it has two or more children, and it is owned by agent a .

Now, define s_2 as a substrategy profile of s_1 (in a broad sense, child of a child of, and so on). Let s_2 be the strategy profile chosen by a at the pivot position, as shown below.

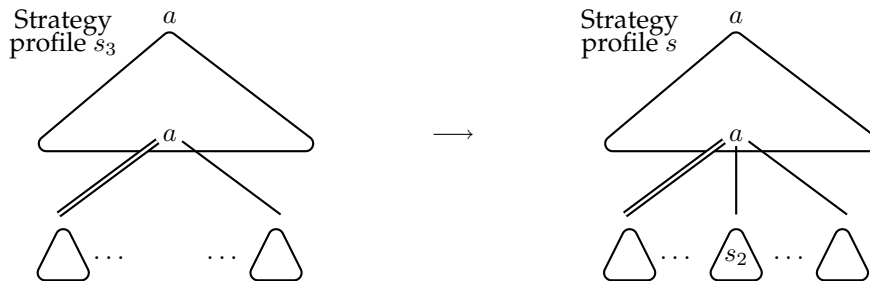


The strategy profile s_2 is a Nash equilibrium because it lies on the play induced by the Nash equilibrium s_1 . Bear in mind that if agent a converts s_2 into s'_2 , then both strategy profiles induce the same play, and therefore the same outcome. Indeed, by construction, agent a owns no node with two or more children on the play induced by s_2 .

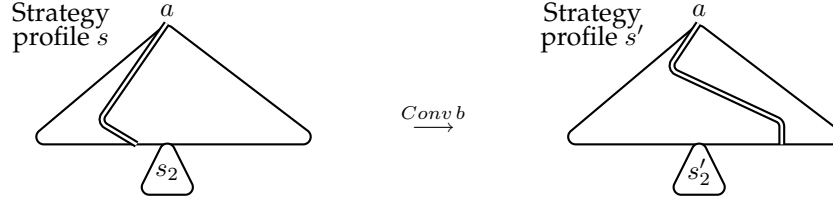
Let g_2 be the underlying game of s_2 . Build g_3 by removing g_2 from g at the pivot position, as shown below. Since the pivot in g has two or more children, g_3 has one or more child. Therefore the object g_3 is a well-defined game.



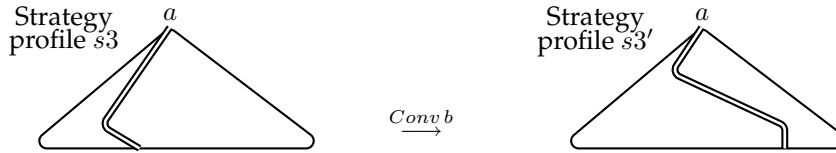
The game g_3 has at most n nodes so, by induction hypothesis, it has a Nash equilibrium s_3 . Build a strategy profile s by inserting s_2 into s_3 , as a non-chosen strategy profile at the pivot position (from where g_2 was removed), as shown below.



By construction, g is the underlying game of the strategy profile s . The remainder of the proof shows that s is a Nash equilibrium. Note that the play induced by s corresponds to the play induced by s_3 , since it does not involve s_2 . Let an agent b convert s into a new strategy profile s' , and hereby s_2 to s'_2 . Case split on the play induced by s' involving s'_2 or not. If the play induced by s' does not involve s'_2 , then the conversion is as follows.

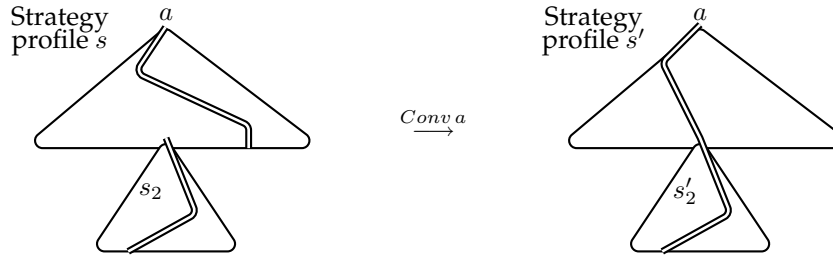


The same agent b can convert s_3 to s'_3 inducing the same play as s' , as shown below.

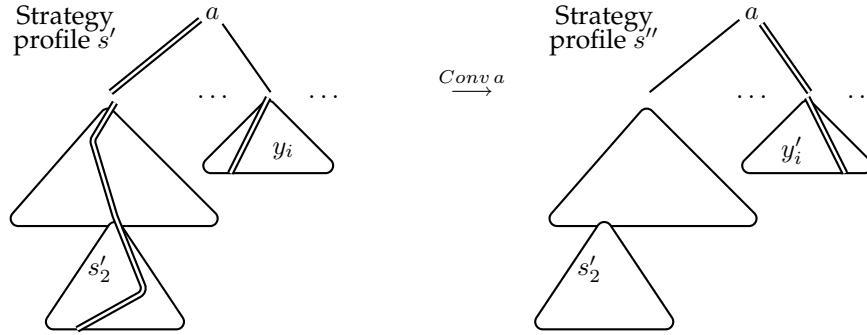


Since s_3 is a Nash equilibrium, the agent b does not prefer the outcome induced by s'_3 to the outcome induced by s_3 . In the same way, agent b does not prefer the outcome induced by s' to the outcome induced by s .

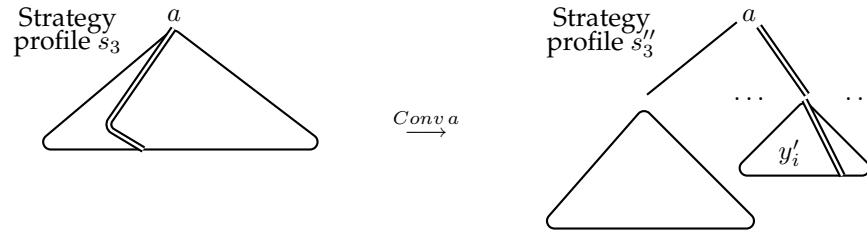
If the play induced by s' involves s'_2 , then the converting agent must be a . Indeed, at the pivot owned by a , the chosen substrategy profile must actually change towards s'_2 , while s_2 was inserted in s_3 as a non-chosen strategy profile. In addition, as argued before, the outcomes induced by s'_2 and s_2 are the same since a is the converting agent.



Recall that, by case splitting assumption, there exists x_i a subgame of g different from g_1 and with the following property: agent a can convert any strategy profile whose underlying game is x_i into a new strategy profile he prefers to s_1 . Agent a would also prefer the new strategy profile to s'_2 since both s_1 and s'_2 induce the same outcome. Let y_i be the substrategy profile of s' that corresponds to the subgame x_i mentioned above. By assumption, agent a can convert y_i into some y'_i preferred over s'_2 . Below, let us detail further the structure of s' and its conversion by agent a into a preferred s'' .



By transitivity of convertibility, agent a can convert s into s'' , so it can convert s_3 into a corresponding s''_3 as shown below. (Looking again at the conversions of s into s' and s' into s'' may help.)



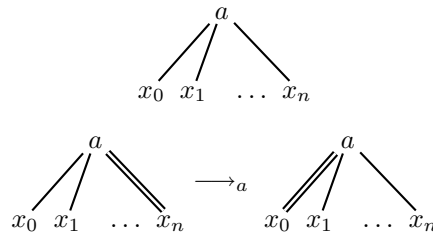
If agent a prefers s' to s , then it prefers s'_2 to s_3 , since s' and s'_2 (s and s_3) induce the same outcome. Therefore, by preference transitivity, agent a prefers y'_i to s_3 , and thus s''_3 to s_3 , which contradicts s_3 being a Nash equilibrium. Therefore, a does not prefer s' to s .

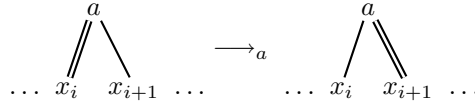
The long case splitting above has shown that all agents are happy with s , which means that s is a Nash equilibrium for g . \square

The following lemma was formally proved in [29] and chapter 4. By contraposition, it amounts to the implication $2) \Rightarrow 1)$.

Lemma 30 *If there exists a cycle in the preferences of one agent, then there exists a game without Nash equilibrium.*

Proof Informally, let an agent a prefer x_1 to x_0 , x_2 to x_1 , and so on, and x_0 to x_n . The game displayed below has no Nash equilibrium, as suggested graphically.





The symbol $s \rightarrow_a s'$ means that agent a both prefers s' to s and can convert s to s' . So, the formula $s \rightarrow_a s'$ witnesses the agent's non-happiness. \square

Now it is possible to state and prove the theorem.

Theorem 31 *The following two propositions are equivalent.*

1. *Preferences over the outcomes are acyclic.*
2. *Every abstract sequential game has a Nash equilibrium.*

Proof For $1) \Rightarrow 2)$, assume acyclicity of the preferences for all agents. Consider the bigger preferences defined by the transitive closures of the original preferences. These are strict partial orders, so for each game there exists a Nash equilibrium with respect to them, by lemma 29. It was proved in [29] and chapter 4 that smaller preferences generate more equilibria (less demanding agents are happier). This gives a Nash equilibrium with respect to the original preferences. For $2) \Rightarrow 1)$, invoke lemma 30. \square

5.4 Conclusion

This chapter generalises Kuhn's result in a way that is completely different from [29] and chapter 4. First, it performs an induction on the size of games instead of a structural induction on games. Second, it invokes the induction hypothesis exactly twice in two different contexts, while the proof in [29] and chapter 4 uses the induction hypothesis once for any subgame, *i.e.* arbitrarily many times altogether. Third, it performs a "simple" transitive closure instead of a "complicated" topological sorting. Fourth, while [29] and chapter 4 present an extension of Kuhn's proof, this chapter does not use "backward induction". Fifth, while [29] and chapter 4 prove a triple equivalence, this chapter proves only a double equivalence that may be difficult to extend to the triple one through a direct proof. Sixth, the hypotheses required for this chapter are slightly stronger than for [29] and chapter 4. Indeed, if one wants to compute equilibria with this proof, equality between agents needs to be decidable (when going down the path induced by s_1). Seventh, in addition the proof in [29] and chapter 4 seems to be more intuitive and informative. To sum up, although the proof in [29] and chapter 4 seems to have many advantages, substantial proof-theoretical differences make this new proof interesting as alternative spirit and technique.

Chapter 6

Graphs and Path Equilibria

The quest for optimal/stable paths in graphs has gained attention in a few practical or theoretical areas. To take part in this quest this chapter adopts an equilibrium-oriented approach that is abstract and general: it works with (quasi-arbitrary) arc-labelled digraphs, and it assumes very little about the structure of the sought paths and the definition of equilibrium, *i.e.* optimality/stability. In this setting, this chapter presents a sufficient condition for equilibrium existence for every graph; it also presents a necessary condition for equilibrium existence for every graph. The necessary condition does not imply the sufficient condition a priori. However, the chapter pinpoints their logical difference and thus identifies what work remains to be done. Moreover, the necessary and the sufficient conditions coincide when the definition of optimality relates to a total order, which provides a full-equivalence property. These results are applied to network routing.

6.1 Introduction

This chapter provides an abstract formalism that enables generic proofs, yet accurate results, about path equilibria in graphs. Beyond this, the purpose of this chapter is to provide a tool for a generalisation of sequential (tree-) games within graphs. However, these game-theoretic facets are not discussed in this chapter. In addition to the game-theoretic application, the results presented in this chapter may help solve problems of optimisation/stability of paths in graphs: a short example is presented for the problem of network routing .

6.1.1 Contribution

This chapter introduces the terminology of *dalographs* which refers to finite, arc-labelled, directed graphs with non-zero outdegree, *i.e.* each of whose node has an outgoing arc. An embedding of arc-labelled digraphs into dalographs shows that the non-zero-outdegree constraint may not yield a serious loss of generality. The paths that are considered in this chapter are infinite. Indeed, finite paths and infinite paths are of slightly different "types". Considering both may hinder an algebraic approach of the system. However, another embedding allows representing finite paths in a dalograph as infinite paths in another

dalograph. This shows that the infiniteness constraint may not yield a serious loss of generality either. Note that the non-zero-outdegree constraint ensures existence of infinite paths, starting from any node. This uniformity facilitates an algebraic approach of the system. The paths considered in this chapter are non-self-crossing, which somehow suggests consistency. This sounds desirable in many areas, but it may be an actual restriction in some others.

In this formalism, a path induces an ultimately periodic sequence of labels (of arcs that are involved in the path). An arbitrary binary relation over ultimately periodic sequences of labels is assumed and named *preference*. This induces a binary relation over paths, which is also named *preference*. It is defined as follows. Given two paths starting from the same node, one is preferred over the other if the sequence of labels that is induced by the former is preferred over the sequence of labels that is induced by the latter. Maximality of a given path in a graph means that no path is preferred over the given path. A *strategy* is an object built over a dalograph. It amounts to every node choosing an outgoing arc. This way, a strategy induces paths starting from any given node. An *equilibrium* is a strategy inducing optimal paths for any node.

The proof of equilibrium existence is structured as follows. First, a *seeking-forward* function is defined so that given a node it returns a path. Given a node, the function chooses a path that is maximal (according to the definition of the previous paragraph), and the function "follows" the path until the remaining path is not maximal (among the paths starting from the current node). In this case, the function chooses a maximal path starting from the current node and proceeds as before. All of this is done under the constraint that a path is non-self-crossing. Under some conditions, this procedure yields a path that is maximal not only at its starting node, but also at all nodes along the path. Such a path is called a *hereditary maximal path*. Equipped with this lemma, the existence of an equilibrium for every dalograph is proved as follows by induction on the number of arcs in the dalograph.

- Compute a hereditary maximal path in the dalograph.
- Remove the arcs of the dalograph that the path ignored while visiting adjacent nodes and get a smaller dalograph.
- Compute an equilibrium on this smaller dalograph and add the ignored arcs back. This yields an equilibrium for the original dalograph.

The sufficient condition for equilibrium existence involves a notion lying between strict partial order and strict total order, namely *strict weak order*. This chapter requires a few preliminary results about strict weak orders. Moreover, the definition of the seeking-forward function requires the design of a recursion principle that is also used as a proof principle in this chapter. To show the usefulness of this sufficient condition, this chapter provides a few examples of non-trivial relations that meet the requirements of this condition: lexicographic extension of a strict weak order, Pareto-like order, and two limit-set-oriented orders. Then, as an application to network routing, one derives a sufficient condition for a routing policy to guarantee existence of stable routing solutions.

The proof of the necessary condition for equilibrium existence involves various closures of binary relations. Most of the closures defined here are related to properties that are part of the sufficient condition. For instance, the

sufficient condition involves transitivity of some preference (binary relation), and the necessary condition involves the transitive closure of the preference. Each of those various closures is proved to preserve existence of equilibrium. More specifically, if a preference ensures existence of equilibrium for every dalograph, then the closure of the preference also ensures existence of equilibrium for every dalograph. A new closure is defined subsequently as the combination of all the above-mentioned closures. Thanks to a preliminary development introducing the notion of *simple closure*, this combination closure also preserves existence of equilibrium. Since the combination closure has a few properties, this gives a non-trivial necessary condition for equilibrium existence. All the closures mentioned above are defined inductively through inference rules, which allows using rule induction as a proof principle. To show the usefulness of this necessary condition, this chapter provides an example of a non-trivial relation that does not meet the requirements of this condition.

However, not all the properties that are part of the sufficient condition can be easily turned into a simple closure preserving equilibrium existence (and hereby be part of the necessary condition). Therefore, the necessary condition and sufficient condition do not look exactly the same. Some examples show that the necessary condition is too weak. Some more work, *i.e.* the design of more general simple closures, is likely to take care of those examples, and hereby provide a stronger necessary condition. However, there is also likely to be more complex examples that would require the design of more complex simple closures. As to the sufficient condition, it is still unclear whether or not it is too strong.

In the case where the preference is a total order, the sufficient and necessary conditions coincide. This gives a necessary and sufficient condition (on a total order preference) for existence of equilibrium in every dalograph. This leads to a necessary and sufficient condition (on a total order preference) for equilibrium existence in the network routing application mentioned above.

6.1.2 Contents

Section 6.2 defines a graph-like structure named dalograph and the notion of equilibrium in dalographs. Section 6.3 defines a refinement of partial order that is named strict weak order in the literature. This section also connects the notion of strict weak order to other properties of binary relations. Section 6.4 proves a sufficient condition that guarantees existence of equilibrium for all dalographs. It also gives a few examples of non-trivial relations meeting the requirements of the sufficient condition. Finally, it gives an application in network routing. Section 6.5 defines the notion of simple closure and proves a property about the *union* of simple closures. Section 6.6 discusses preservation of equilibrium existence by closures, and thus provides a necessary condition for existence of equilibrium for all dalographs. Section 6.7 compares the sufficient condition and the necessary condition, and shows that they coincide in the total order case but not in general. It also describes a relation that does not meet the requirements of the necessary condition. Finally, it gives a necessary and sufficient condition, in the total order case, for equilibrium existence in the network routing application mentioned above.

6.1.3 Conventions

Unless otherwise stated, universal quantifiers are usually omitted in front of (semi-) formal statements. For instance, a claim of the form $P(x, y)$ should be read $\forall x, y, P(x, y)$.

Usually, when proving a claim of type $\tau_0 \Rightarrow \dots \Rightarrow \tau_{n-1} \Rightarrow \tau_n \Rightarrow \tau$, the beginning of the proof implicitly assumes $\tau_0, \dots, \tau_{n-1}$, and τ_n , and then starts proving τ .

Given u a non-empty finite sequence, u^f (resp. u^l) represents the first (resp. last) element of u .

The notation $P(x) \triangleq Q(x)$ means that P is defined as coinciding with Q .

The negation of a relation \prec is written as follows.

$$\alpha \not\prec \beta \triangleq \neg(\alpha \prec \beta)$$

The corresponding incomparability relation is defined below.

$$\alpha \# \beta \triangleq \alpha \not\prec \beta \wedge \beta \not\prec \alpha$$

The inverse of a relation \prec is defined as follows.

$$\alpha \succ \beta \triangleq \beta \prec \alpha$$

Restrictions of binary relations are written as follows.

$$\alpha \prec|_S \beta \triangleq \alpha \prec \beta \wedge (\alpha, \beta) \in S^2$$

Function composition is defined as (almost) usual. For any functions f_1 of type $A \rightarrow B$ and f_2 of type $B \rightarrow C$, the composition of f_1 and f_2 is written $f_2 \circ f_1$. It is of type $A \rightarrow C$ and it is defined as follows.

$$f_2 \circ f_1(x) \triangleq f_2(f_1(x))$$

In this chapter, $f_1 \circ f_2$ may be written $f_1 f_2$, in the same order.

6.2 Dalographs and Equilibria

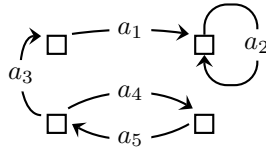
Subsection 6.2.1 defines a class of arc-labelled digraphs, subsection 6.2.2 defines a class of paths in such arc-labelled digraphs, and subsection 7.2.3 derives a notion of equilibrium from a notion of maximality for paths. The more general will be those graphs and paths, the more general will be the derived notion of equilibrium.

6.2.1 Dalographs

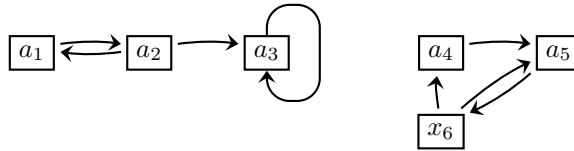
This subsection gives a name to the class of arc-labelled directed graphs each of whose node has at least one outgoing arc. Then it briefly justifies why arc labelling is "more general" than node labelling, and why the non-zero-outdegree constraint may not yield a serious loss of generality.

Definition 32 (Dalograph) *Let L be a collection of labels. A dalograph is a finite directed graph whose arcs are labelled with elements of L , and such that every node has a non-zero outdegree.*

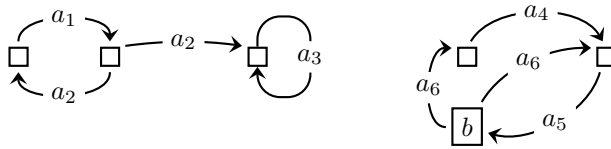
The picture below is an example of dalograph, with 5 labels from a_1 to a_5 . Squares represent nodes, but they may not be displayed in every further example.



One may argue that labelling the arcs leave out the node-labelled digraphs, as the example below.

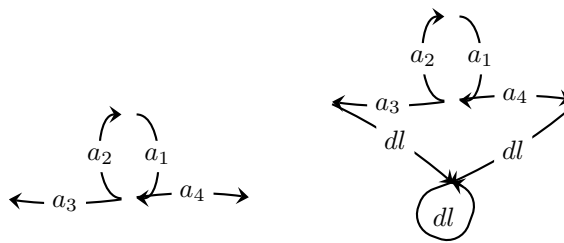


However, there is a natural way of embedding node-labelled digraphs into arc-labelled digraphs: for each node, remove the label from the node and put it on all the outgoing arcs of the node. The picture below shows the above example being translated into dalographs.



Due to the embedding, arc labelling appears to be more general than node labelling.

Demanding that all nodes have non-zero outdegree may not yield a serious loss of generality either. Indeed there is also an embedding from arc-labelled digraphs into dalographs, using a dummy node and a dummy label dl . For example, this embedding maps the left-hand graph below to the right-hand dalograph.



6.2.2 Walks and Paths

This subsection defines a walk in a dalograph as a finite sequence of nodes that goes continuously along the arcs of the dalograph. In addition, a walk must stop when intersecting itself. A looping walk is defined as a walk intersecting

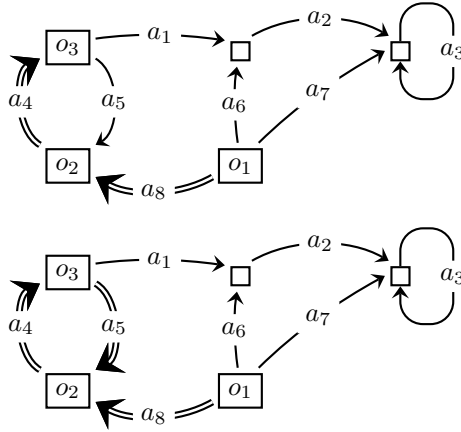
itself. This enables the definition of an alternative induction principle along walks. Then, paths are defined as infinite sequences that are consistent with looping walks, and this definition is briefly justified.

Definition 33 (Walks as sequence of nodes) *Walks in a digraph are defined by induction as follows.*

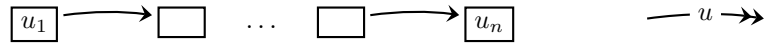
- ϵ is the empty walk.
- o is a walk for any node o of the digraph.
- If $o_0 \dots o_n$ is a walk, if o does not occur in o_0, \dots, o_{n-1} , and if oo_0 is an arc of the digraph, $oo_0 \dots o_n$ is also a walk.

If a node occurs twice in a walk, then it occurs at the end of the walk. In this case, the walk is said to be looping.

The first picture below shows the walk $o_1 o_2 o_3$ using double lines. The second picture shows the looping walk $o_1 o_2 o_3 o_2$.



It will often be convenient to represent walks partially. A (possibly empty) non-looping walk is represented by a double-headed arrow that is labelled with a sequence of nodes. For instance, the left-hand walk $u = u_1 \dots u_n$ below is represented by the right-hand picture.



A looping walk is represented by a squared-bracket-headed arrow that is labelled with a sequence of nodes. For instance below, the left-hand looping walk $uovo$ is represented by the right-hand picture.



A usual induction principal for walks would go from the empty walk to bigger walks. Here, walks take place in finite dalographs so they are bounded. This allows an alternative induction principle for walks.

Lemma 34 (Nibbling induction principle for walks) *Let g be a dalograph. Let P be a predicate on walks in g . Assume that given any walk x , $P(xo)$ for all walks xo implies $P(x)$. Then the predicate holds for all walks.*

Proof First, P holds for all looping walks since, by definition, x being a looping walk implies xo is not a walk. Second, assume that there exist walks that do not satisfy P . Let x be one such walk. By finiteness of g , it makes sense to take x as long as possible. So all walks xo satisfy P by definition of x , therefore x satisfies P by assumption. This is a contradiction. \square

This lemma can be used as a proof principle or, quite similarly, as a programming principle. Indeed, let f be a function on walks. If the definition of f on any input x invokes only results of the computation of f with the walks xo , then f is well-defined a priori.

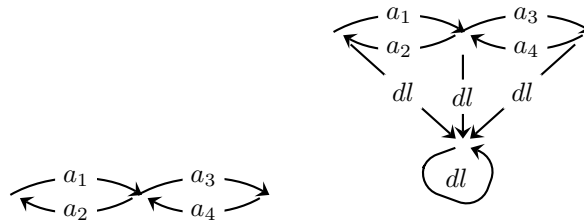
Paths are defined as consistent infinite continuations of looping walks.

Definition 35 (Paths as looping walks) *Given a looping walk $uovo$, the corresponding path is the infinite sequence $u(ov)^\omega$. Given a walk x and a path Γ such that $x\Gamma$ is also a path, Γ is called a continuation of the walk x .*

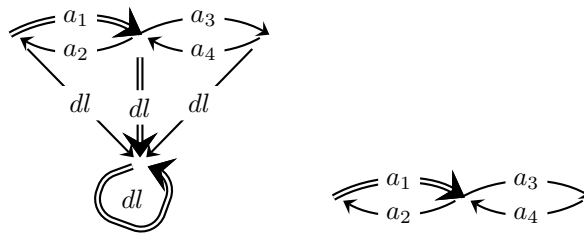
Informally, a path has a memory, and when visiting a node for the second time and more, it chooses the same next node as for the first time. The formalism ensures the following properties.

Lemma 36 *Every walk has a continuation and a looping walk $uovo$ has a unique continuation $(vo)^\omega$.*

If one wishes to deal with "finite paths", i.e. non-looping walks, within the dalograph formalism, it is possible to add a dummy node and dummy arcs from all original nodes to the dummy one, as shown below.



Through the embedding above, the left-hand infinite path below may be interpreted as the right-hand finite path.



6.2.3 Equilibria

This subsection compares paths by comparing infinite sequences of labels. It defines strategies as dalographs having chosen an outgoing arc for each of their nodes, and it defines equilibria as strategies inducing maximal paths everywhere.

Below, finite sequences of labels induced by walks are defined by induction (the usual induction principle, not the nibbling one). Subsequently, paths induce infinite sequences of labels.

Definition 37 (Induced sequence) *Induced sequences are inductively defined as follows.*

- *A walk o induces the empty sequence.*
- *A walk $o_1 o_2 x$ induces the sequence $a.seq(o_2 x)$ where a is the label on the arc $o_1 o_2$.*

A path corresponding to a looping walk $uovo$ induces a sequence $seq(uo)seq(ovo)^\omega$. A sequence that is induced by a path starting from a given node is said eligible at this node.

The ultimately periodic sequences over labels are comparable through an arbitrary binary relation.

Definition 38 (Preference) *A relation over the ultimately periodic sequences over labels is called a preference. Preferences are written \prec in this chapter.*

Comparing the induced sequences of two paths starting from the same node is a natural way of comparing paths.

Definition 39 (Paths comparison) *If paths Γ_1 and Γ_2 start from the same node and induce sequences γ_1 and γ_2 with $\gamma_1 \prec \gamma_2$ (resp. $\gamma_1 \not\prec \gamma_2$), one writes $\Gamma_1 \prec \Gamma_2$ (resp. $\Gamma_1 \not\prec \Gamma_2$) by abuse of notation.*

The following definition captures the notion of maximality (with respect to a preference) of a path among the continuations of a given walk.

Definition 40 (Maximal continuation) *The notation $m_{g,\prec}(xo, \Gamma)$ accounts for the property: $xo\Gamma$ is a path in g and $o\Gamma \not\prec o\Gamma'$ for all paths $xo\Gamma'$ of the dalograph g .*

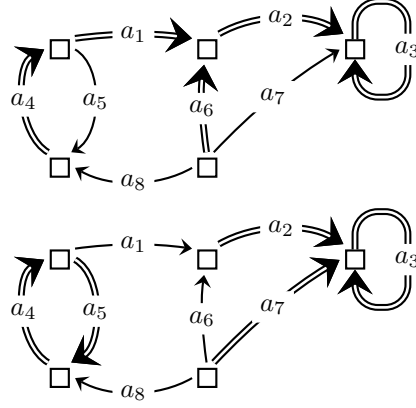
One may write $m(xo, \Gamma)$ instead of $m_{g,\prec}(xo, \Gamma)$ when there is no ambiguity. In addition to a path being maximal from the point of view of its starting node, this chapter needs to discuss paths all of whose subpaths are maximal from their starting points. The notion of hereditary maximality captures this idea below.

Definition 41 (Hereditary maximal path) *Let Γ be a path. If $m_{g,\prec}(o, \Gamma')$ for any decomposition $\Gamma = xo\Gamma'$ where xo is a non-looping walk, one writes $hm_{g,\prec}(\Gamma)$.*

A strategy is an object built on a dalograph by choosing an outgoing arc at each node.

Definition 42 (Strategy) Given a dalograph g , a strategy s on g is a pair (g, c) , where c is a function from the nodes of g to themselves, and such that for all nodes o , the pair $(o, c(o))$ is an arc of g .

The two examples below show two strategies with the same underlying dalograph. The choices are represented by double lines.



As seen in the pictures above, given a strategy and a node, the strategy induces exactly one path starting from this node.

Definition 43 (Induced continuation) Let s be a strategy. Define $p(s, o)$ such that $o \cdot p(s, o)$ is the path induced by s starting from o .

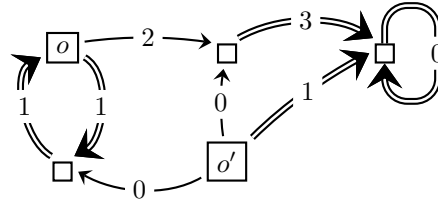
Given a dalograph and a preference, a strategy on the dalograph is a local equilibrium at a given node if it induces a hereditary maximal path at this node.

Definition 44 (Local equilibrium) $LEq_{\prec}(s, o) \triangleq m_{g, \prec}(o, p(s, o))$

A global equilibrium for a dalograph is intended to be a strategy inducing a maximal path at every node of the dalograph. It follows that a global equilibrium can be defined as a strategy that is a local equilibrium for every node of the dalograph.

Definition 45 (Global equilibrium) $GEq_{\prec}(s) \triangleq \forall o \in g, LEq_{\prec}(s, o)$

In the rest of this chapter, the terminology of equilibrium refers to global equilibria, unless otherwise stated, or \prec -equilibrium to avoid ambiguity. In the example below, arcs are labelled with natural numbers and \prec is the lexicographic extension of the usual order to infinite sequences of natural numbers. The following strategy is a local equilibrium for node o' but not for node o .



If a preference is a subrelation of another preference, and if a given strategy is a local/global equilibrium with respect to the bigger preference, then the strategy is also a local/global equilibrium with respect to the smaller preference. This is formally stated below.

Lemma 46 (Equilibrium for subpreference) *Preservation by subrelation is stated as follows.*

$$\prec \subseteq \prec' \Rightarrow LEq_{\prec'}(s, o) \Rightarrow LEq_{\prec}(s, o)$$

$$\prec \subseteq \prec' \Rightarrow GEq_{\prec'}(s, o) \Rightarrow GEq_{\prec}(s, o)$$

Proof Note that the following formula holds.

$$\prec \subseteq \prec' \Rightarrow m_{g, \prec'}(xo, \Gamma) \Rightarrow m_{g, \prec}(xo, \Gamma). \quad \square$$

Assume that two preferences coincide on a subdomain, *i.e.* a subset of the ultimately periodic sequences over labels. Assume that a given dalograph involves only sequences from this subdomain, *i.e.* all paths in the dalograph induce only sequences in the subdomain. In this case, a local/global equilibrium for this dalograph with respect to one preference is also a local/global equilibrium for this dalograph with respect to other preference. As for the lemma above, this can be proved by simple unfolding of the definitions. This result is stated below.

Lemma 47 *Let g be a dalograph involving sequences in S only. Assume that $\prec|_S = \prec'|_S$. In this case,*

$$LEq_{\prec}(s, o) \Leftrightarrow LEq_{\prec'}(s, o)$$

$$GEq_{\prec}(s) \Leftrightarrow GEq_{\prec'}(s)$$

The following lemma relates to the fact that when there is only one choice, this choice is the best possible one.

Lemma 48 *If \prec is an irreflexive preference, then a dalograph each of whose node has outdegree 1 has a \prec -equilibrium.*

Proof In such a case, only one strategy corresponds to that dalograph and there is only one possible path starting from any node. Therefore, each path is maximal, by irreflexivity, and this strategy is an equilibrium. \square

6.3 Binary Relations

This section defines a few predicates on binary relations. Some properties connecting these predicates are also presented. Subsection 6.3.1 deals with binary relations in general, while subsection 6.3.2 focuses on binary relations over sequences.

6.3.1 General Binary Relations

This subsection slightly rephrases the notion of strict weak order, which already exists in the literature. This subsection also defines the notion of imitation. It turns out that strict weak orders can be equivalently characterised by a few other simple formulae. The structure of such relations is studied in detail.

The notions of transitive, asymmetric, and irreflexive binary relation that are used in this chapter are the usual ones.

Definition 49 (Transitivity, asymmetry and irreflexivity) *A binary relation \prec is transitive if it complies with the first formula below. Asymmetry amounts to the second formula, and irreflexivity to the third one.*

$$\begin{array}{ll} \alpha \prec \beta \Rightarrow \beta \prec \gamma \Rightarrow \alpha \prec \gamma & \text{transitivity} \\ \alpha \prec \beta \Rightarrow \beta \not\prec \alpha & \text{asymmetry} \\ \alpha \not\prec \alpha & \text{irreflexivity} \end{array}$$

Asymmetry implies irreflexivity, as being formally stated below.

Lemma 50 $(\forall \alpha, \beta, \alpha \prec \beta \Rightarrow \beta \not\prec \alpha) \Rightarrow \forall \alpha, \alpha \not\prec \alpha$

Proof Instantiate β with α . □

Transitivity of the negation of a relation does not imply transitivity of the relation. For instance, let $\alpha \neq \beta$ and define \prec on $\{\alpha, \beta\}$ by $\alpha \prec \beta$ and $\beta \prec \alpha$, and nothing more. Due to the symmetry, \prec is not transitive, while $\not\prec$ is transitive. The next definition and two lemmas shows that transitivity of the negation almost implies transitivity.

Definition 51 (Strict weak order) *A strict weak order is an asymmetric relation whose negation is transitive.*

Equivalent definitions of strict weak order can be found in the literature. The rest of this subsection explains some properties of strict weak orders and it gives an intuition of the underlying structure. The following lemma shows that transitivity of a relation can be derived from asymmetry of the relation and transitivity of its negation.

Lemma 52 *A strict weak order is transitive.*

Proof Let α, β and γ be such that $\alpha \prec \beta$ and $\beta \prec \gamma$. Therefore $\beta \not\prec \alpha$ and $\gamma \not\prec \beta$ by asymmetry, and $\gamma \not\prec \alpha$ by transitivity of the negation. If $\alpha \not\prec \gamma$ then $\beta \not\prec \alpha$ and transitivity of the negation yields $\beta \not\prec \gamma$, which is absurd. Therefore $\alpha \prec \gamma$. □

Strict weak orders have a second property that makes non-comparability an equivalence relation.

Lemma 53 *If \prec is a strict weak order, then $\#$ is an equivalence relation.*

Proof If $\neg(\alpha \# \alpha)$ then $\alpha \prec \alpha$, which contradicts asymmetry by lemma 50. So $\#$ is reflexive. If $\alpha \# \beta$ then $\beta \# \alpha$ by definition, so $\#$ is symmetric. If $\alpha \# \beta$ and $\beta \# \gamma$, then $\beta \not\prec \alpha$ and $\gamma \not\prec \beta$ by definition. So $\gamma \not\prec \alpha$ by transitivity of the negation. In the same way, $\alpha \not\prec \gamma$ also holds, so $\alpha \# \gamma$. Therefore $\#$ is transitive. The incomparability relation $\#$ is symmetric by definition, so $\#$ is an equivalence relation. □

A binary relation is lower (resp. upper) imitating if any two non-comparable elements have the same predecessors (resp. successors).

Definition 54 (Lower/upper imitation) A binary relation \prec complying with the following formula is called a lower-imitating relation.

$$(\alpha \# \beta \wedge \gamma \prec \alpha) \Rightarrow \gamma \prec \beta$$

A binary relation \prec complying with the following formula is called an upper-imitating relation.

$$(\alpha \# \beta \wedge \alpha \prec \gamma) \Rightarrow \beta \prec \gamma$$

A relation that is both lower and upper imitating is called an imitating relation.

Lower and upper imitations do not only look "symmetric" definitions, they also are.

Lemma 55 If a relation is lower (resp. upper) imitating, then its inverse is upper (resp. lower) imitating.

Proof Let \prec be a binary relation. Assume that \prec is lower imitating and assume that $\alpha \#^{-1} \beta$ and $\beta \succ \gamma$. So $\alpha \# \beta$ and $\gamma \prec \beta$, which implies $\gamma \prec \alpha$, and therefore $\alpha \succ \gamma$. \square

Guaranteeing asymmetry makes the predicates of lower and upper imitations coincide.

Lemma 56 Let \prec be an asymmetric relation. In this case, \prec is upper-imitating iff \prec is lower-imitating.

Proof left-to-right. Assume $\alpha \# \beta$ and $\gamma \prec \beta$. If $\alpha \prec \gamma$ then $\beta \prec \gamma$ by upper-imitation, which contradicts asymmetry. If $\alpha \# \gamma$ then $\alpha \prec \beta$ by upper-imitation, which is absurd. So $\gamma \prec \alpha$. The converse follows lemma 55, knowing that the inverse of an asymmetric relation is asymmetric. \square

Because of the lemma above, only the concept of imitation will be referred to when dealing with asymmetric relations. The next lemma connects the notion of imitation to the notion of transitivity.

Lemma 57 If \prec is transitive and $\#$ is an equivalence relation, then \prec is irreflexive and imitating.

Proof If $\alpha \prec \alpha$ then $\#$ is not reflexive, which contradicts the assumption. So \prec is irreflexive. Assume that $\alpha \# \beta$ and $\beta \prec \gamma$. If $\gamma \prec \alpha$ then $\alpha \prec \beta$ by transitivity, which is absurd. If $\gamma \# \alpha$, then $\beta \# \gamma$ by transitivity of $\#$, which is absurd. Therefore $\alpha \prec \gamma$. \square

Imitation implies transitivity provided that "small" cycles are forbidden.

Lemma 58 Let \prec be without any cycle involving 2 or 3 elements. If \prec is imitating then \prec is transitive.

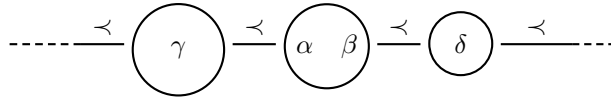
Proof Assume $\alpha \prec \beta$ and $\beta \prec \gamma$. If $\gamma \prec \alpha$ then there is a cycle involving 3 elements, which is absurd. Now assume that $\gamma \# \alpha$. If \prec is imitating then $\beta \prec \alpha$, which contradicts asymmetry. So $\alpha \prec \gamma$. \square

Actually, an imitating relation is a strict weak order provided that "small" cycles are forbidden.

Lemma 59 *Let \prec be without any cycle of length 2 or 3. If \prec is imitating then \prec is a strict weak order.*

Proof Let α, β and γ be such that $\beta \not\prec \alpha$ and $\gamma \not\prec \beta$. First case, assume that $\alpha \# \beta$ and $\beta \# \gamma$. If $\gamma \prec \alpha$ then $\beta \prec \alpha$ and $\gamma \prec \beta$ by double imitation. Since transitivity is guaranteed by lemma 58, this yields $\gamma \prec \alpha$ which is absurd by assumption. Second case, either $\alpha \# \beta$ and $\beta \prec \gamma$ or $\alpha \prec \beta$ and $\beta \# \gamma$. By imitation, $\alpha \prec \gamma$, so $\gamma \not\prec \alpha$ by asymmetry. Third case, $\alpha \prec \beta$ and $\beta \prec \gamma$, so $\alpha \prec \gamma$ by transitivity, so $\gamma \not\prec \alpha$. \square

The picture below is meant to give an intuitive understanding of what is a strict weak order. The circles represent equivalence classes of $\#$. Here we have $\gamma \prec \alpha, \beta, \delta$ and $\gamma, \alpha, \beta \prec \delta$ and $\alpha \# \beta$. Informally, a strict weak order looks like a knotted rope.



The following lemma sums up the results of this subsection.

Lemma 60 *Let \prec be a binary relation, and let $\#$ be the corresponding non-comparability relation. The following three propositions are equivalent.*

1. \prec is a strict weak order.
2. \prec is transitive and $\#$ is an equivalence relation.
3. \prec is imitating and has no cycle of length 2 or 3.

Proof Implication 1 \Rightarrow 2 by lemmas 52 and 53, implication 2 \Rightarrow 3 by lemma 57, and implication 3 \Rightarrow 1 by lemma 59. \square

6.3.2 Binary Relations over Sequences

This subsection deals with binary relations over (finite or infinite) sequences built over (finite or infinite) collections. It introduces a few notions such as E-prefix, A-transitivity and subcontinuity.

For binary relations over sequences, the following captures the notions of preservation by prefix elimination and preservation by prefix addition.

Definition 61 (E-prefix and A-prefix) *A binary relation \prec over sequences is said E-prefix when complying with the following formula.*

$$u\alpha \prec u\beta \Rightarrow \alpha \prec \beta$$

It is said A-prefix when complying with the following formula.

$$\alpha \prec \beta \Rightarrow u\alpha \prec u\beta$$

It is possible to define a mix between transitivity and A-prefix.

Definition 62 (A-transitivity) A relation over sequences is said A-transitive when complying with the following formula.

$$\alpha \prec \beta \Rightarrow u\beta \prec \gamma \Rightarrow u\alpha \prec \gamma$$

The following lemma shows the connections between transitivity, A-prefix, and A-transitivity. Note that the converse implications do not hold a priori.

Lemma 63 Transitivity plus A-prefix imply A-transitivity, and A-transitivity implies transitivity.

Proof Consider the formula defining A-transitivity. For the first claim, $u\alpha \prec u\beta$ by A-prefix, and conclude by transitivity. For the second claim, instantiate u with the empty sequence. \square

The following lemma shows that a strict weak order that is preserved by prefix elimination is A-transitive.

Lemma 64 An E-prefix strict weak order is A-transitive.

Proof Assume that $\alpha \prec \beta$ and $u\beta \prec \gamma$. Therefore $\beta \not\prec \alpha$ by asymmetry, so $u\beta \not\prec u\alpha$ by contraposition of E-prefix. If $u\alpha \not\prec \gamma$ then $u\beta \not\prec \gamma$ by transitivity of the negation, which contradicts the assumption. Therefore $u\alpha \prec \gamma$. \square

The A-prefix predicate seems to be a bit too restrictive for what is intended in this chapter, but a somewhat related notion will be useful. Informally, consider a relation \prec that is A-prefix and transitive. If $\alpha \prec u\alpha$ then $u\alpha \prec u^2\alpha$ and we have an infinite ascending chain $\alpha \prec u\alpha \prec \dots \prec u^n\alpha \prec \dots$. In this case, a natural thought might be to topologically close the chain with u^ω as an upper bound, i.e. $\alpha \prec u\alpha \prec \dots \prec u^n\alpha \prec \dots \prec u^\omega$. The following definition captures this informal thought.

Definition 65 (Subcontinuity) A relation over sequences is said subcontinuous when complying with the following formula, where u is any non-empty finite sequence.

$$\alpha \prec u\alpha \Rightarrow u^\omega \not\prec \alpha$$

The next definition proceeds in the vein of the previous one and gives an alternative, slightly more complex definition of subcontinuity.

Definition 66 (Alt-subcontinuity) A relation over sequences is alt-subcontinuous when complying with the following formula, where v and t are any non-empty finite sequences.

$$\alpha \prec t\beta \Rightarrow (v\alpha \prec \beta \vee \alpha \prec (tv)^\omega)$$

The next lemma shows that alt-subcontinuity is "stronger" than subcontinuity.

Lemma 67 An alt-subcontinuous asymmetric relation is subcontinuous.

Proof Let \prec be an alt-subcontinuous asymmetric relation. Assume that $\alpha \prec u\alpha$ for some u and α . By alt-subcontinuity, $u\alpha \prec \alpha \vee \alpha \prec u^\omega$. By asymmetry, $u\alpha \not\prec \alpha$, so $\alpha \prec u^\omega$. Therefore $u^\omega \not\prec \alpha$ by asymmetry. \square

The following lemma states that, under some demanding conditions, subcontinuity implies alt-subcontinuity.

Lemma 68 *A E-prefix subcontinuous strict weak order is alt-subcontinuous.*

Proof Assume that $v\alpha \not\prec \beta$, $\alpha \not\prec (tv)^\omega$, and $\alpha \prec t\beta$. If $(tv)^\omega \not\prec t\beta$ then $\alpha \not\prec t\beta$ by transitivity of \prec , which is a contradiction, so $(tv)^\omega \prec t\beta$. So $(vt)^\omega \prec \beta$ by E-prefix. If $(vt)^\omega \not\prec v\alpha$ then $(vt)^\omega \not\prec \beta$ by transitivity of \prec , which is a contradiction, so $(vt)^\omega \prec v\alpha$, and $(tv)^\omega \prec \alpha$. By subcontinuity, this implies $\alpha \not\prec tv\alpha$. By assumption, $v\alpha \not\prec \beta$, so $tv\alpha \not\prec t\beta$ by contraposition of E-prefix. Therefore $\alpha \not\prec t\beta$ by transitivity of the negation, which contradicts the assumption, so $\alpha \prec t\beta$. \square

The following two definitions generalise the notion of E-prefix.

Definition 69 *Let W be a non-empty finite set of finite sequences such that the empty sequence is not in W and such that at most one sequence of length 1 is in W . If for all walks u in W there exists a walk v in W such that $u\alpha \prec v\beta$, one writes $W\alpha \prec W\beta$.*

Definition 70 (Gen-E-prefix) *A relation over sequences is said gen-E-prefix when complying with the following formula.*

$$W\alpha \prec W\beta \Rightarrow \alpha \prec \beta$$

6.4 Equilibrium Existence

Subsection 6.4.1 proves a sufficient condition that guarantees existence of equilibrium in any dalograph; subsection 6.4.2 gives a few examples of non-trivial relations meeting the requirements of the sufficient condition; subsection 6.4.3 applies the result to network routing.

6.4.1 The Proof

The two main stages of the proof are, first, building hereditary maximal paths, which are the only paths involved in equilibria, and second, proceeding by induction on the number of arcs in the dalograph.

Hereditary maximal paths seem difficult to be built a priori. A weaker notion is that of semi-hereditary maximal path. On the one hand, a subpath of a hereditary maximal path is a maximal continuation of the preceding node along the hereditary maximal path. On the other hand, a subpath of a semi-hereditary maximal path is a maximal continuation of the beginning of the semi-hereditary maximal path, as defined below.

Definition 71 (Semi-hereditary maximal path) *Let x be a non-empty walk of continuation Γ . If $m(xy, \Gamma')$ for all decompositions $\Gamma = y\Gamma'$ where xy is a walk, one writes $shm(x, \Gamma)$.*

Unsurprisingly, semi-hereditary maximality implies maximality.

Lemma 72 $shm(x, \Gamma) \Rightarrow m(x, \Gamma)$

Proof Instantiate y with the empty walk in the definition of shm . \square

The next result states that semi-hereditary maximality is implied by maximality plus semi-hereditary maximality of the subpath starting one node further along the path.

Lemma 73 $m(x, o\Gamma) \Rightarrow shm(xo, \Gamma) \Rightarrow shm(x, o\Gamma)$

Proof Consider a decomposition $o\Gamma = y\Gamma'$ where xy is a walk. If y is empty then $\Gamma' = o\Gamma$ so $m(xy, \Gamma')$ by assumption. If y is not empty then $y = oz$. Since $\Gamma = z\Gamma'$ and $shm(xo, \Gamma)$, it follows that $m(xoz, \Gamma')$. Hence $shm(x, o\Gamma)$. \square

The notion of semi-hereditary maximality can also be defined along another induction principle for walks. In such a case, the definition would look like lemma 73.

As to the nibbling induction principle for walks in subsection 6.2.2, it will be exploited as a recursive programming principle instead. It is used below to define a function that expects a non-empty walk and returns a path. It starts from one given node, finds a direction that promises maximality, follows the direction until there is a better direction to be followed, and so on, but without ever going back. It stops when the walk is looping because a looping walk defines a path. It is called the seeking-forward function.

Definition 74 (Seeking-forward function) Let \prec be an acyclic preference and let g be a dalograph. Define a function that expects a non-empty walk in g and a continuation of this walk. More specifically, $F(x, \Gamma)$ is recursively defined along the nibbling induction principle for walks.

- If x is a looping walk of continuation Γ , let $F(x, \Gamma) \triangleq \Gamma$.
- If x is not a looping walk then case split as follows.
 1. If $m(x, \Gamma)$ then $F(x, \Gamma) \triangleq oF(xo, \Gamma')$, where $\Gamma = o\Gamma'$.
 2. If $\neg m(x, \Gamma)$ then $F(x, \Gamma) \triangleq oF(xo, \Gamma')$ for some $o\Gamma'$ such that $m(x, o\Gamma')$ and $x^l\Gamma \prec x^l o\Gamma'$.

The following lemma states that whatever the point of view, a path processed by the seeking-forward function is somehow not worse than the original path. Before reading the lemma recall that given u a non-empty finite sequence, u^f (resp. u^l) represents the first (resp. last) element of u .

Lemma 75 Let \prec be an irreflexive and A-transitive preference. Let u be a non-empty suffix of x . The following formula holds.

$$uF(x, \Gamma) \prec u^f \Delta \Rightarrow u\Gamma \prec u^f \Delta$$

Proof By nibbling induction on walks. Base step, x is a looping walk of unique continuation Γ . By definition of F we have $F(x, \Gamma) = \Gamma$, so the claim holds. Inductive step, case split on Γ being or not a maximal continuation of x . First case, $m(x, \Gamma)$, so $F(x, \Gamma) = oF(xo, \Gamma')$ with $\Gamma = o\Gamma'$. Assume that $uF(x, \Gamma) \prec u^f \Delta$, so $oF(xo, \Gamma') \prec u^f \Delta$. By induction hypothesis, $uo\Gamma' \prec u^f \Delta$, so $u\Gamma \prec u^f \Delta$. Second case, $\neg m(x, \Gamma)$, so $F(x, \Gamma) = oF(xo, \Gamma')$ and $x^l\Gamma \prec x^l o\Gamma'$ for some $o\Gamma'$ maximal continuation of x . Assume that $uF(x, \Gamma) \prec u^f \Delta$, so $uoF(xo, \Gamma') \prec u^f \Delta$. By induction hypothesis, $uo\Gamma' \prec u^f \Delta$. Since u is a non-empty suffix of x , we have $u^l = x^l$. It follows that $u^l\Gamma \prec u^l o\Gamma'$. By A-transitivity, $u\Gamma \prec u^f \Delta$. \square

The next lemma describes an involution property of the seeking-forward function, which suggests that seeking-forward once is enough.

Lemma 76 *Let \prec be an irreflexive and A-transitive preference. The following formula holds.*

$$F(x, F(x, \Gamma)) = F(x, \Gamma)$$

Proof By nibbling induction on walks. Base step: x is a looping walk of unique continuation Γ . By definition of F we have $F(x, \Gamma) = \Gamma$, so $F(x, F(x, \Gamma)) = F(x, \Gamma)$. Inductive step: by definition of F , $F(x, \Gamma) = oF(xo, \Gamma')$ for some $o\Gamma'$ such that $m(x, o\Gamma')$. So $F(x, F(x, \Gamma)) = F(x, oF(xo, \Gamma'))$. Since $m(xo, F(xo, \Gamma'))$ by contraposition of lemma 75, $F(x, oF(xo, \Gamma')) = oF(xo, F(xo, \Gamma'))$. By induction hypothesis, $F(xo, F(xo, \Gamma')) = F(xo, \Gamma')$, so $F(x, F(x, \Gamma)) = oF(xo, \Gamma') = F(x, \Gamma)$. \square

The next analytical property about the seeking-forward function shows that a subpath of a fixed point is also a fixed point.

Lemma 77 *Let \prec be an irreflexive and A-transitive preference, and let xo be a walk. The following formula holds.*

$$F(x, o\Gamma) = o\Gamma \Rightarrow F(xo, \Gamma) = \Gamma$$

Proof By definition of F , $F(x, o\Gamma) = o'F(xo', \Gamma')$ for some $o'\Gamma'$. It follows that $o' = o$ and $F(xo, \Gamma') = \Gamma$. Therefore $F(xo, F(xo, \Gamma')) = F(xo, \Gamma)$ by term substitution. By lemma 76, $F(xo, F(xo, \Gamma')) = F(xo, \Gamma')$, therefore $F(xo, \Gamma) = \Gamma$ by transitivity of equality. \square

The following lemma states that any fixed point of the seeking-forward function is a semi-hereditary maximal path. The converse also holds and the proof is straightforward, but this converse result is not relevant in this chapter.

Lemma 78 *Let \prec be an irreflexive, E-prefix, and A-transitive preference. The following formula holds.*

$$F(x, \Gamma) = \Gamma \Rightarrow shm(x, \Gamma)$$

Proof By nibbling induction on the walk x . Base step, x is a looping walk, so $m(x, \Gamma)$ by definition of m . For any decomposition $\Gamma = y\Gamma'$ where xy is a walk, y is empty and $\Gamma' = \Gamma$ because x is a looping walk, so $m(xy, \Gamma')$. This shows $shm(x, \Gamma)$. Inductive step, assume that $F(x, o\Gamma) = o\Gamma$. So $F(xo, \Gamma) = \Gamma$ according to lemma 77, and $shm(xo, \Gamma)$ by induction hypothesis, so $m(xo, \Gamma)$ by lemma 72. If $\neg m(x, o\Gamma)$, there exists a path $xo'\Gamma'$ such that $x'o\Gamma \prec x'o'\Gamma'$ and $F(x, o\Gamma) = o'F(xo', \Gamma')$. So $o' = o$, and $o\Gamma \prec o\Gamma'$ since \prec is E-prefix, which contradicts $m(xo, \Gamma)$. Therefore $m(x, o\Gamma)$, and lemma 73 allows concluding. \square

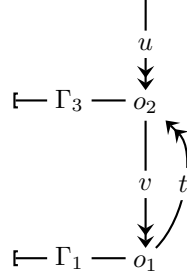
The next lemma gives a sufficient condition so that semi-hereditary maximality implies hereditary maximality.

Lemma 79 *Let \prec be an irreflexive, E-prefix, and A-transitive preference whose inverse of negation is alt-subcontinuous, and let g be a dalograph.*

$$shm_{g, \prec}(o, \Gamma) \Rightarrow hm_{g, \prec}(o\Gamma)$$

Proof Assume that $shm(o, \Gamma)$. It suffices to prove by induction on the walk x that $o\Gamma = xo_1\Gamma_1$ implies $m(o_1, \Gamma_1)$. First case, x is empty, so $o = o_1$ and $\Gamma_1 = \Gamma$. Since $shm(o, \Gamma)$ by assumption, $m(o_1, \Gamma_1)$ by definition of shm . Second

case, assume that $o\Gamma = xo'o_1\Gamma_1$ and let $o_1\Gamma_2$ be a path. If $xo_1\Gamma_2$ is a path, $o\Gamma_1 \not\prec o\Gamma_2$ by semi-hereditary maximality of Γ . If $xo_1\Gamma_2$ is not a path, then x and Γ_2 intersect. Let o_2 be the first intersection node along Γ_2 . So $x = uo_2v$ and $\Gamma_2 = to_2\Gamma_3$, with xo_1to_2 being a looping walk. The situation is displayed below.



$o_1\Gamma_2$ is a path so $o_2\Gamma_3$ is also a path. Since $o\Gamma = uo_2vo_1\Gamma_1$ and u is smaller than x , the induction hypothesis says that $m(o_2, vo_1\Gamma_1)$. Therefore $o_2vo_1\Gamma_1 \not\prec o_2\Gamma_3$. Because xo_1to_2 is a looping walk, $u(o_2vo_1t)^\omega$ is a path, by definition, and so is $(o_2vo_1t)^\omega$. Since $m(o_2, vo_1\Gamma_1)$, we have $(o_2vo_1t)^\omega \prec o_2vo_1\Gamma_1$. Let α be the sequence induced by $o_1\Gamma_1$, β by $o_2\Gamma_3$, y by o_2vo_1 , and z by o_1to_2 . We have $y\alpha \not\prec \beta$ and $\alpha \not\prec (zy)^\omega$, so $\alpha \not\prec z\beta$ by alt-subcontinuity of the inverse of the negation. Therefore $o_1\Gamma_1 \not\prec o_1to_2\Gamma_3$, which shows that $m(o_1, \Gamma_1)$. \square

Now that hereditary maximal paths are available/computable, it is possible to prove the existence of equilibrium for every dalograph, by induction on the number of arcs in the dalograph. Compute one hereditary maximal path and remove the arcs that the path ignored while visiting adjacent nodes. Compute an equilibrium on this smaller dalograph and add the ignored arcs back. This yields an equilibrium for the bigger dalograph. This procedure is detailed below.

Theorem 80 *If a preference is included in a subcontinuous E-prefix strict weak order, then any dalograph has a global equilibrium with respect to the preference.*

Proof According to lemma 46, it suffices to show the claim for preferences that are actually subcontinuous E-prefix strict weak orders. Proceed by induction on the number of arcs in the dalograph. If the dalograph has one arc only, we are done since there is only one possible strategy and since preference is irreflexive. Now assume that the claim is proved for any dalograph with n or less arcs, and consider a dalograph g with $n + 1$ arcs. If each node has only one outgoing arc, we are done by lemma 48. Now assume that there exists a node o with at least two outgoing arcs as shown below.



Since the preference is a subcontinuous E-prefix strict weak order, it is irreflexive, A-transitive, E-prefix, and alt-subcontinuous by lemmas 50, 64 and 68. According to lemmas 76, 78 and 79 there exists a hereditary maximal path Γ

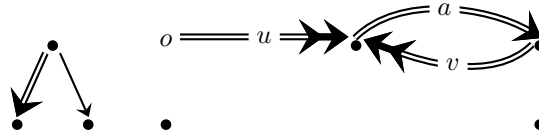
starting from the node o . Let $u(av)^\omega$ be the corresponding sequences of labels, where u and v are finite (possibly empty) sequences of labels and a is a label.



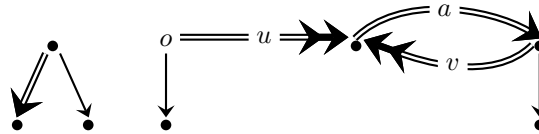
From the dalograph g and the path Γ , build a new dalograph g' as follows. Remove all the arcs that are dismissed by the choices along Γ . There is at least one such arc since Γ starts at node o , which has two or more outgoing arcs. Below, the dalograph g is to the left and the dalograph g' to the right.



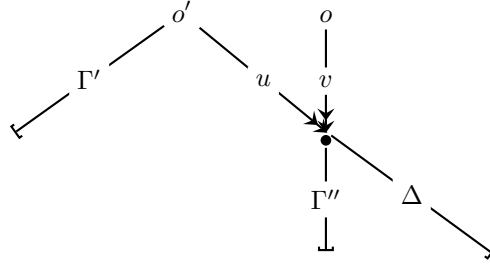
The new dalograph g' has n or less arcs, so it has an equilibrium by induction hypothesis. Such an equilibrium is represented below and named s' . The double lines represent the choices of the strategy, one choice per node.



The path induced by s' from the node o is the same as Γ because there is only one possible path from this node in the dalograph g' , after removal of the arcs. From the equilibrium s' , build a new strategy named s by adding back the arcs that were removed when defining g' , as shown below.



The remainder of the proof shows that s is an equilibrium for g . Since Γ is hereditary maximal, s is a local equilibrium for any node involved in Γ . Now let o' be a node outside Γ , and let Γ' be the path induced by s (and s') starting from o' . Consider another path starting from o' . If the new path does not involve any arc dismissed by Γ , then the new path is also valid in g' , so it is not greater than Γ' since s' is an equilibrium. If the new path involves such an arc, then the situation looks like the picture below, where $\Gamma = v\Gamma''$ and the new path is $u\Delta$.



By hereditary maximality, $\Gamma'' \not\prec \Delta$, so $u\Gamma'' \not\prec u\Delta$ by contraposition of E-prefix. The path $u\Gamma''$ is also valid in g' , so $\Gamma' \not\prec u\Gamma''$ because s' is an equilibrium. Therefore $\Gamma' \not\prec u\Delta$ by transitivity of the negation. Hence, Γ' is also maximal in g , and s is an equilibrium. \square

6.4.2 Examples

This subsection gives a few examples of non-trivial relations included in some subcontinuous E-prefix strict weak order. First, it discusses the lexicographic extension of a strict weak order, and a component-wise order in Pareto style, which happens to be included in the lexicographic extension. Second, it defines two limit-set orders involving maxima and minima of a set. The second order happens to be included in the first one.

Lexicographic extension

The lexicographic extension is widely studied in the literature. It is the abstraction of the way entries are ordered in a dictionary, hence the name. The lexicographic extension usually involves total orders, but it can be extended to strict weak orders.

Definition 81 (Lexicographic extension) Let (A, \prec) be a set equipped with a strict weak order. The lexicographic extension of the strict weak order is defined over infinite sequences of elements of A .

$$\frac{a \prec b}{a\alpha \prec^{lex} b\beta} \qquad \frac{\alpha \prec^{lex} \beta \quad a \sharp b}{a\alpha \prec^{lex} b\beta}$$

For instance $01^\omega \prec^{lex} 10^\omega$ and $(03)^\omega \prec^{lex} 1^\omega \prec^{lex} (30)^\omega$, with the usual order on figures 0, 1 and 3.

The following defines when two sequences of the same length are "equivalent" with respect to a strict weak order.

Definition 82 Let (A, \prec) be a set equipped with a strict weak order. Let u and v be two sequences of length n of elements of A . If for all i between 1 and n , $u_i \sharp v_i$, then one writes $u \sharp v$.

Next lemma characterises the lexicographic extension of a strict weak order through equivalent prefixes followed by comparable letters.

Lemma 83 *Let (A, \prec) be a set equipped with a strict weak order, and let \prec^{lex} be its lexicographic extension.*

$$\begin{aligned} \alpha \prec^{lex} \beta \\ \iff \\ \exists u, v \in A^*, \exists a, b \in A, \exists \alpha', \beta' \in A^\omega, \alpha = ua\alpha' \wedge \beta = vb\beta' \wedge u \# v \wedge a \prec b \end{aligned}$$

Proof left-to-right: by rule induction. First rule, $\alpha \prec^{lex} \beta$ comes from $a \prec b$, so $\alpha = a\alpha'$ and $\beta = b\beta'$ for some α' and β' . Second rule, $\alpha \prec^{lex} \beta$ comes from $\alpha' \prec^{lex} \beta'$ and $a \# b$. The induction hypothesis provides some u, v ; au and bv are witnesses of for the claim. Right-to-left. By induction on the length of u . If u is empty then the claim corresponds to the first inference rule. For the inductive step, invoke the second inference rule. \square

The following lemma states the transitivity of the lexicographic extension of a strict weak order.

Lemma 84 *Let (A, \prec) be a set equipped with a strict weak order. Then \prec^{lex} is transitive.*

Proof Assume that $\alpha \prec^{lex} \beta$ and $\beta \prec^{lex} \gamma$. By lemma 83, this gives one decomposition $\alpha = ua\alpha'$ and $\beta = ub\beta'$ with $a \prec b$, and one decomposition $\beta = vb'\beta''$ and $\gamma = vc\gamma'$ with $b' \prec c$. Therefore $\beta = ub\beta' = vb'\beta''$. Case split along the following three mutually exclusive cases: first $u = v$, second u is a proper prefix of v , and third v is a proper prefix of u . If $u = v$ then $b = b'$ so $a \prec c$ by transitivity of \prec , so $\alpha \prec^{lex} \gamma$ by lemma 83. If u is a proper prefix of v then $u = vb'v'$, so $\alpha = vb'v'\alpha'$ and $\gamma = vc\gamma'$ with $b' \prec c$, therefore $\alpha \prec^{lex} \gamma$ by lemma 83. If v is a proper prefix of u then $v = ubu'$, so $\alpha = ua\alpha'$ and $\gamma = ubu'\gamma'$ with $a \prec b$, therefore $\alpha \prec^{lex} \gamma$ by lemma 83. \square

By contraposition, lemma 83 yields the following characterisation of $\not\prec^{lex}$, the negation of \prec^{lex} .

Lemma 85 *Let (A, \prec) be a set equipped with a strict weak order, and let \prec^{lex} be its lexicographic extension.*

$$\begin{aligned} \alpha \not\prec^{lex} \beta \\ \iff \\ \forall u, v \in A^*, \forall a, b \in A, \forall \alpha', \beta' \in A^\omega, \alpha = ua\alpha' \wedge \beta = vb\beta' \wedge u \# v \Rightarrow a \not\prec b \end{aligned}$$

The construction of the lexicographic extension preserves strict weak ordering, as stated below.

Lemma 86 *Let (A, \prec) be a set equipped with a strict weak order. The derived \prec^{lex} is also a strict weak order.*

Proof Since \prec^{lex} is transitive by lemma 84, it suffices to show that $\#^{lex}$ is an equivalence relation, by lemma 60. The relation \prec^{lex} is irreflexive, which can be proved by rule induction on its definition. So $\#^{lex}$ is reflexive. It is also symmetric by definition. By lemma 85, $\alpha \#^{lex} \beta$ is equivalent to $u \# v$ for all decompositions $\alpha = u\alpha'$ and $\beta = v\beta'$ with u and v of the same length. This property is transitive since $\#$ is transitive by lemma 60. \square

The lexicographic extension of a strict weak order is E-prefix.

Lemma 87 *Let (A, \prec) be a set equipped with a strict weak order. The derived \prec^{lex} is E-prefix.*

Proof Prove by induction on u that $u\alpha \prec^{lex} u\beta$ implies $\alpha \prec^{lex} \beta$. If u is empty, that is trivial. If $u = au'$, then $au'\alpha \prec^{lex} au'\beta$ must come from the second inference rule of the definition of \prec^{lex} , which means that $u'\alpha \prec^{lex} u'\beta$. Therefore $\alpha \prec^{lex} \beta$ by induction hypothesis. \square

The lexicographic extension of a strict weak order is also subcontinuous.

Lemma 88 *Let (A, \prec) be a set equipped with a strict weak order. The derived \prec^{lex} is subcontinuous.*

Proof Assume that $u^\omega \prec^{lex} \alpha$, so $u^n u' \# v$, $u = u' au''$, $\alpha = vb\alpha'$, and $a \prec b$ for some n, u', u'', a, b and α' . Therefore $u^{n+1} u' \# uv$, which can be written $u^n u' au'' u' \# uv$. Decompose $uv = v' cv''$ with v'' and $u'' u'$ of the same length. So $u^n u' \# v'$ and $a \# c$. So $c \prec b$ by strict weak ordering and $v \# v'$ since $\#$ is an equivalence relation. Since $u\alpha = v'c(v''b\alpha')$ and $\alpha = vb\alpha'$, $u\alpha \prec^{lex} \alpha$. Therefore \prec^{lex} is subcontinuous. \square

Theorem 80 together with lemmas 86, 87, and 88 allows stating the following.

Theorem 89 *A dalograph labelled with elements of a strict weak order has a global equilibrium with respect to the lexicographic extension of the strict weak order.*

Pareto Extension

A Pareto extension allows comparing vectors with comparable components. A first vector is "greater" than a second one if it is not smaller component-wise and if it is greater for some component. This can be extended to infinite sequences.

Definition 90 (Pareto extension) *Let (A, \prec) be a set equipped with a strict weak order. The Pareto extension of the strict weak order is defined over infinite sequences of elements of A .*

$$\alpha \prec^P \beta \quad \triangleq \quad \forall n \in \mathbb{N}, \beta(n) \not\prec \alpha(n) \quad \wedge \quad \exists n \in \mathbb{N}, \alpha(n) \prec \beta(n)$$

For instance $01^\omega \not\prec^P 10^\omega$ but $(01)^\omega \prec^P 1^\omega \prec^P (13)^\omega$ with the usual order on figures 0, 1 and 3.

The following lemma states that a Pareto extension of a strict weak order is included in the lexicographic extension of the same strict weak order.

Lemma 91 *Let (A, \prec) be a set equipped with a strict weak order.*

$$\alpha \prec^P \beta \quad \Rightarrow \quad \alpha \prec^{lex} \beta$$

Proof Assume that $\alpha \prec^P \beta$, so by definition $\beta(n) \not\prec \alpha(n)$ for all naturals n , and $\alpha(n) \prec \beta(n)$ for some natural n . Let n_0 be the smallest natural n such that $\alpha(n) \prec \beta(n)$. So $\alpha = u\alpha'$ and $\beta = vb\alpha'$ for some u and v of length n_0 and $a \prec b$. For i between 0 and $n_0 - 1$, $v(i) \not\prec u(i)$ by assumption, and $u(i) \not\prec v(i)$ by definition of n_0 . Therefore $u \# v$, so $\alpha \prec^{lex} \beta$ by lemma 60. \square

Pareto extension also guarantees existence of equilibrium in dalographs.

Theorem 92 *A dalograph labelled with elements of a strict weak order has a global equilibrium with respect to the derived Pareto extension.*

Proof Invoke lemmas 91 and 46, and theorem 89. \square

Max-Min Limit-Set Order

As discussed in subsection 6.3.1, two non-comparable elements of a strict weak order compare the same way against any third element. Therefore, comparison of two elements amounts to comparison of their non-comparability equivalence classes.

Definition 93 *Let (E, \prec) be a set equipped with a strict weak order. This induces a total order defined as follows on the \sharp -equivalence classes A^\sharp and B^\sharp .*

$$A^\sharp \prec B^\sharp \triangleq \exists x \in A^\sharp, \exists y \in B^\sharp, x \prec y$$

Through total ordering, it is easy to define a notion of extrema of finite sets.

Definition 94 (Class maximum and minimum) *Let (E, \prec) be a set equipped with a strict weak order. The maximum (resp. minimum) of a finite subset A of E is the maximal (resp. minimal) \sharp -class intersecting A .*

An order over sets is defined below. It involves extrema of sets.

Definition 95 (Max-min order over sets) *Let (E, \prec) be a set equipped with a strict weak order. The max-min order is defined on finite subsets of E .*

$$A \prec^{Mm} B \triangleq \max(A) \prec \max(B) \vee (\max(A) = \max(B) \wedge \min(A) \prec \min(B))$$

For instance $\{1, 2, 3\} \prec^{Mm} \{0, 4\}$ and $\{0, 2, 3\} \prec^{Mm} \{1, 3\}$ with the usual total order over the naturals.

The negation of the above order is characterised below.

Lemma 96 *Let (E, \prec) be a set equipped with a strict weak order.*

$$A \not\prec^{Mm} B \Leftrightarrow \max(A) \not\prec \max(B) \wedge (\max(A) = \max(B) \Rightarrow \min(A) \not\prec \min(B))$$

The max-min construction preserves strict weak ordering, as stated below.

Lemma 97 *Let (E, \prec) be a set equipped with a strict weak order. The Max-min order on E is also a strict weak order.*

Proof First, prove transitivity of its negation. Assume that $A \not\prec^{Mm} B$ and $B \not\prec^{Mm} C$. By assumption, $\max(A) \not\prec \max(B)$ and $\max(B) \not\prec \max(C)$, so $\max(A) \not\prec \max(C)$ since \prec is a total order for \sharp -classes. Assume that $\max(A) = \max(C)$, so $\max(A) = \max(C) = \max(B)$. Therefore $\min(A) \not\prec \min(B)$ and $\min(B) \not\prec \min(C)$ follows from the assumptions, and $\min(A) \not\prec \min(C)$ by total ordering. This shows that $A \not\prec^{Mm} C$. Second, prove that \sharp^{Mm} is an equivalence relation: just note that $A \sharp^{Mm} B$ is equivalent to $\max(A) = \max(B)$ and $\min(A) = \min(B)$. \square

The elements that appear infinitely many times in an infinite sequence constitute the limit set of the sequence. Sequences with non-empty limit sets can be compared through max-min comparisons of their limit sets.

Definition 98 (Max-min limit-set order) *Let (E, \prec) be a set equipped with a strict weak order. For α infinite sequence over E , let L_α be its limit set, i.e. the set of the elements that occur infinitely often in α . Two infinite sequences whose limit sets are non-empty are compared as follows.*

$$\alpha \prec^{Mmls} \beta \triangleq L_\alpha \prec^{Mm} \beta$$

For instance $3^n 4^\omega \prec^{Mmls} 0^p (50^q)^\omega$ because $4 \prec 5$ according to the usual order over the naturals.

Next lemma states preservation of strict weak ordering by the max-min construction.

Lemma 99 *Let (E, \prec) be a set equipped with a strict weak order. The max-min limit-set order is a strict weak order over the sequences with non-empty limit set.*

Since the limit set of a sequence is preserved by prefix elimination and addition, the following holds.

Lemma 100 *Let (E, \prec) be a set equipped with a strict weak order. The max-min limit-set order is E -prefix and subcontinuous over the sequences of non-empty limit set.*

Theorem 80 together with lemmas 99 and 100 allows stating the following.

Theorem 101 *A dalograph labelled with elements of a strict weak order has a global equilibrium with respect to the derived max-min limit-set order.*

Max-Min-Light Limit-Set Order

Roughly speaking, the max-min-light order relates sets such that the elements of one are bigger than the elements of the other.

Definition 102 (Max-min-light order) *Let (E, \prec) be a set equipped with a strict weak order. The max-min-light order is defined on finite subsets of E .*

$$A \prec^{Mml} B \triangleq \forall x \in A, \forall y \in B, y \not\prec x \quad \wedge \quad \exists x \in A, \exists y \in B, x \prec y$$

For instance $\{1, 2\} \not\prec^{Mml} \{0, 3\}$ but $\{0, 1\} \prec^{Mml} \{1\} \prec^{Mml} \{1, 2\}$

Definition 103 (Max-min-light limit-set order) *Let (E, \prec) be a set equipped with a strict weak order. For α infinite sequence over E , let L_α be its limit set, i.e. the set of the elements that occur infinitely often in α . Two infinite sequences whose limit sets are non-empty are compared as follows.*

$$\alpha \prec^{Mmlls} \beta \triangleq L_\alpha \prec^{Mml} \beta$$

The following theorem states that the max-min-light limit-set order guarantees equilibrium existence.

Theorem 104 *A dalograph labelled with elements of a strict weak order has a global equilibrium with respect to the derived max-min-light limit-set order.*

Proof Note that the max-min-light limit-set order is included in the max-min limit-set order. Conclude by lemma 46 and theorem 101. \square

6.4.3 Application to Network Routing

The following issue is related to existing literature such as [17].

Definition 105 *A routing policy is a binary relation over finite words over a collection of labels. A routing problem is a finite digraph whose arcs are labelled with the above-mentioned labels. In addition, one node is called the target. It has outdegree zero and it is reachable from any node through some walk in the digraph. A routing strategy for the routing problem is a function mapping every node different from the target to one of the arcs going out that node. A routing strategy is said to be a routing equilibrium if for each node, the path that is induced by the strategy starting from that node leads to the target, and if for each node, no strategy induces a better (according to the routing policy) such path.*

The following lemma gives a sufficient condition for every routing problem to have a routing equilibrium. The condition may not be decidable in general, but it is decidable on the domain of every finite routing problem.

Lemma 106 *If a routing policy is (included in) an E-prefix strict weak order \prec^r such that $v \not\prec^r uv$ for all v and u , then every routing problem has a routing equilibrium.*

Proof A routing problem can be transformed into a dalograph as follows. Add a dummy node to the routing problem, which is a digraph. Add an arc from the target to the dummy node and from the dummy node to itself. Add dummy labels dl on both arcs. From the routing policy \prec^r over finite words, build a preference \prec over the union of two sets. The first set is made of the infinite words over the original labels L (without the dummy label). The second set is made of the concatenations of finite words over the original labels and the infinite word dl^ω built only with the dummy label.

$$\frac{u \prec^r v}{udl^\omega \prec vdl^\omega} \qquad \frac{\alpha \in L^\omega \quad u \in L^*}{\alpha \prec udl^\omega}$$

Since \prec^r is E-prefix by assumption, \prec is also E-prefix. Since $v \not\prec^r uv$ for all v and u by assumption, \prec is subcontinuous. Therefore the built dalograph has a global equilibrium, by theorem 80. This global equilibrium corresponds to a routing equilibrium. \square

6.5 Simple closures

This section introduces the notion of simple closure, which characterises some operators on relations, and the notion of union of simple closures. This yields two monoids whose combination is similar to a semiring (distributivity is in the opposite direction though). This development intends to show that if given

simple closures preserve a predicate, then any finite restriction of their union also preserves this predicate. This result will be useful in section 6.6.

The section starts with the following general lemma. It states that if a predicate is preserved by given functions, then it is preserved by any composition of these functions.

Lemma 107 *Let f_0 to f_n be functions of type $A \rightarrow A$. Let Q be a predicate on A . Assume that each f_k preserves Q . Then for all x in A and all w words on the f_k the following formula holds.*

$$Q(x) \Rightarrow Q(w(x))$$

Proof By induction on w . First case, w is the empty word, i.e. the identity function. So x equals $w(x)$, and $Q(x)$ implies $Q(w(x))$. Second case, w equals $f_k w'$ and the claim holds for w' . Assume $Q(x)$, so $Q(w'(x))$ by induction hypothesis, and $Q(f_k \circ w'(x))$ since f_k preserves Q . \square

In the remainder of this section, the function domain A that is mentioned in lemma 107 above will be a set of relations. More specifically, A will be the relations of arity r , for an arbitrary r that is fixed throughout the section. In addition, the symbol X represents a vector (X_1, \dots, X_r) of dimension r .

Usually in mathematics, the closure of an object is the smallest bigger (or equal) object of same type that complies with some given predicates. What follows describes a certain kind of closures for relations of arity r , namely simple closures. Simple closures are operators (on relations of arity r) inductively defined through inference rules. The first rule ensures that the simple closures are bigger or equal than the original relation. The other rules pertain to the intended properties of the simple closure.

Definition 108 *Let f be an operator on relations of arity r . The operator f is said to be a simple closure if it is inductively defined with the first inference rule below and some rules having the same form as the second inference rule below.*

$$\frac{R(X)}{f(R)(X)}$$

$$\frac{K(X, \{X^i\}_{i \in C}) \quad \bigwedge_{i \in C} f(R)(X^i)}{f(R)(X)}$$

The next lemma states a few basic properties involving simple closures and inclusion.

Lemma 109 *Let f be a simple closure. The following formulae hold.*

- $R \subseteq f(R)$
- $R \subseteq R' \Rightarrow f(R) \subseteq f(R')$
- $f \circ f(R) \subseteq f(R)$

Proof The first claim follows the first rule $R(X) \Rightarrow f(R)(X)$. The second claim is proved by rule induction on the definition of f . First case, $f(R)(X)$ comes from $R(X)$. By inclusion, $R'(X)$, so $f(R')(X)$. Second case $f(R)(X)$

comes from $\bigwedge_{i \in C} f(R)(X^i)$. By induction hypothesis, $\bigwedge_{i \in C} f(R')(X^i)$, therefore $f(R')(X)$. The third claim is also proved by rule induction on the definition of f . First case, $f \circ f(R)(X)$ comes from $f(R)(X)$ we are done. Second case $f \circ f(R)(X)$ comes from $\bigwedge_{i \in C} f \circ f(R)(X^i)$. By induction hypothesis, $\bigwedge_{i \in C} f(R)(X^i)$, so $f(R)(X)$. \square

The following lemma generalises the first property of lemma 109.

Lemma 110 *Let f_0 to f_n be simple closures on relations of arity r , and Let w, u , and v be words on the f_k . For all R relation of arity r and for all X vector of dimension r , the following formula holds.*

$$w(R)(X) \Rightarrow uwv(R)(X)$$

Proof First, prove the claim for empty v by induction on u . If u is empty then it is trivial. If $u = f_i u'$ then $u'w(R)(X)$ by induction hypothesis, so $f_i u'w(R)(X)$ by the first part of lemma 109. Second, prove the claim for empty u by induction on v . If v is empty then it is trivial. If $v = v' f_i$ then $wv'(R)(X)$ by induction hypothesis. Since $R \subseteq f_i(R)$ by the first part of lemma 109, we also have $wv' f_i(R)(X)$ by the second part of lemma 109. Third, assume $w(R)(X)$. So $uw(R)(X)$ by the first part of this proof, and $uwv(R)(X)$ by the second part of this proof. \square

Definition 111 (Rule union) *Let f and g be two simple closure on relations of arity r . Assume that f is defined by the induction rules F_1 to F_n , and that g is defined by the induction rules G_1 to G_m . Then, the operator $f + g$ defined by the induction rules F_1 to F_n and G_1 to G_m is also a simple closure on relations of arity r .*

The law $+$ defines an abelian monoid on simple closures on relation of the same arity, the neutral element being the identity operator. Moreover the law $+$ is distributive over the law \circ , but it should be the opposite for $(A \rightarrow A, +, \circ)$ to be a semiring.

The union of two simple closures yields bigger relations than simple closures alone, as stated below. It is provable by rule induction on the definition of f .

Lemma 112 *Let R be a relation and let f and g be simple closures. The following formula holds.*

$$f(R) \subseteq (f + g)(R)$$

The following lemma generalises the previous result. It shows that composition is somehow "bounded" by union.

Lemma 113 *Let f_0 to f_n be simple closures on relations of arity r , and let f equal $\Sigma_{0 \leq k \leq n} f_k$. Let w be a word on the f_k . For all R relation of arity r , we have $w(R) \subseteq f(R)$.*

Proof By induction on w . If w is empty then $w(R) = R$ and the first part of lemma 109 allows concluding. If $w = f_i w'$ then $w'(R) \subseteq f(R)$ by induction hypothesis. So $f_i w'(R) \subseteq f \circ f(R)$ by lemma 112, and $f_i w'(R) \subseteq f(R)$ by the third part of lemma 109. \square

Although composition is "bounded" by union, union is approximable by composition, as developed in the next two lemmas. For any relation R of arity r , the union of given simple closures (applied to R) can be simulated at a given point X by some composition of the same simple closures (applied to R), as stated below.

Lemma 114 *Let f_0 to f_n be simple closures on relations of arity r , and let f equal $\Sigma_{0 \leq k \leq n} f_k$. Let R be a relation of arity r . If $f(R)(X)$ then $w(R)(X)$ for some word w on the f_k .*

Proof By rule induction on the definition of f . First case, assume that $f(R)(X)$ comes from the following rule.

$$\frac{R(X)}{f(R)(X)}$$

Since $R(X)$ holds, $id(R)(X)$ also holds, so the empty word is a witness for the claim. Second case, assume that $f(R)(X)$ is induced by the following rule.

$$\frac{K(X, \{X^i\}_{i \in C}) \quad \bigwedge_{i \in C} f(R)(X^i)}{f(R)(X)}$$

By induction hypothesis, for all i in C , $f(R)(X^i)$ implies that $w_i(R)(X^i)$ for some word w_i . Let w be a concatenation of the w_i . By lemma 110, we have $w(R)(X^i)$ for all i . Assume that the inference rule above comes from f_k . By lemma 110, $f_{k+1} \dots f_n w(R)(X^i)$ holds for all i . So, $f_k f_{k+1} \dots f_n w(R)(X)$ by applying the inference rule. So $f_0 \dots f_n w(R)(X)$ by lemma 110 again. Therefore the word $f_0 \dots f_n w$ is a witness, whatever f_k the inference rule may come from. \square

For any relation R of subdomain S , the union of given simple closures (applied to R) can be approximated on S by a composition of the same simple closures (applied to R), as stated below.

Lemma 115 *Let f_0 to f_n be simple closures on relations of arity r , and let f equal $\Sigma_k f_k$. Let R be a relation of arity r , and let S be a finite subdomain of R . There exists a word w on the f_k such that $f(R) \upharpoonright_S$ is included in $w(R)$.*

Proof Since S is finite, there exist finitely many X in S such that $f(R)(X)$. For each such X there exists a word u such that $u(R)(X)$, by lemma 114. Let w be a concatenation of all these u . By lemma 110, $w(R)(X)$ for each such X . Therefore $f(R) \upharpoonright_S$ is included in $w(R)$. \square

The following lemma shows that if given simple closures preserve a predicate that is also preserved by subrelation, then any finite restriction of the union of the closures also preserves the predicate.

Lemma 116 *Let Q be a predicate on relations that is preserved by the simple closures f_0 to f_n and subrelation, i.e. $R \subset R' \Rightarrow Q(R') \Rightarrow Q(R)$. Then for all relations R of finite subdomain S , $Q(R)$ implies $Q(\Sigma_k f_k(R) \upharpoonright_S)$.*

Proof Lemma 115 provides a w such that $f(R) \upharpoonright_S$ is included in $w(R)$. Then lemma 107 shows that $Q(w(R))$, and preservation by subrelation allows concluding. \square

6.6 Preservation of Equilibrium Existence

This section defines (gen-) E-prefix and A-transitive closure. These are closely related to the (gen-) E-prefix and A-transitivity predicates that are defined in subsection 6.3.2. It is shown that these closures preserve equilibrium existence, *i.e.* if every dalograph has an equilibrium with respect to a preference, then every dalograph also has an equilibrium with respect to the closure of the preference. A combination of these closures is defined, and it also preserves equilibrium existence.

The E-prefix closure of a binary relation is its smallest E-prefix superrelation. It is inductively defined below.

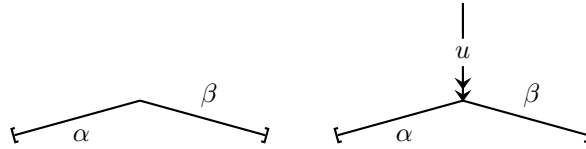
Definition 117 (E-prefix closure)

$$\frac{\alpha \prec \beta}{\alpha \prec^{ep} \beta} \quad \frac{u\alpha \prec^{ep} u\beta}{\alpha \prec^{ep} \beta}$$

The following lemma states that if a preference guarantees existence of equilibrium for all dalographs, then the E-prefix closure of this preference also guarantees existence of equilibrium for all dalographs.

Lemma 118 *E-prefix closure preserves existence of equilibrium. Put otherwise, if all dalographs have \prec -equilibria, then all dalographs have \prec^{ep} -equilibria.*

Proof Let g be a dalograph. First note that, if $\alpha \prec^{ep} \beta$, then there exists u such that $u\alpha \prec u\beta$. (Provable by rule induction). At any node with eligible α and β such that $\alpha \prec^{ep} \beta$, add an incoming path inducing u , as shown below.



This new dalograph g' has a \prec -equilibrium. For any α and β such that $\alpha \prec^{ep} \beta$, the equilibrium does not induce $u\alpha$ when $u\beta$ is possible, so it does not induce α when β is possible. Removing the newly added walks u yields a \prec^{ep} -equilibrium. \square

The transitive closure of a binary relation is its smallest transitive superrelation. It is inductively defined below according to the usual formal definition of transitive closure.

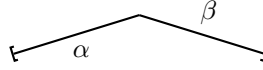
Definition 119 (Transitive closure)

$$\frac{\alpha \prec \beta}{\alpha \prec^t \beta} \quad \frac{\alpha \prec^t \beta \quad \beta \prec^t \gamma}{\alpha \prec^t \gamma}$$

For the transitive closure and other closures that are dealt with in this chapter, it may not be as simple as for E-prefix closure to prove preservation of equilibrium existence. It is done in two steps in this chapter.

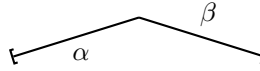
Lemma 120 Let \prec be a preference, and let α and β be such that $\alpha \prec^t \beta$. There exists a dalograph g with the following properties.

- Only one "top" node has several outgoing arcs.
- The dalograph below is a subgraph of the dalograph g .

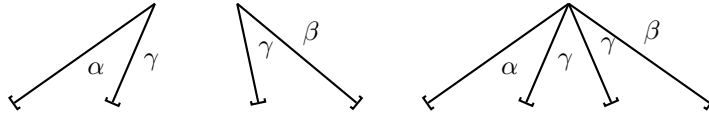


- Only β may be \prec -maximal among the eligible sequences at the top node.

Proof Proceed by rule induction on the definition of the transitive closure. First rule, $\alpha \prec^t \beta$ comes from $\alpha \prec \beta$. The dalograph below complies with the requirements.



Second rule, $\alpha \prec^t \beta$ comes from $\alpha \prec^t \gamma$ and $\gamma \prec^t \beta$. By induction hypothesis there exists one dalograph for $\alpha \prec^t \gamma$ and one for $\gamma \prec^t \beta$, as shown below to the left and the centre. In both dalographs, there is a node with several outgoing arcs. Fuse these nodes as shown below on the right-hand side.



By construction any path in either of the two dalographs is still a path in the new dalograph, so a non-maximal path is still a non-maximal path. By induction hypothesis, if there is a path that is \prec -maximal starting from the top node, then it induces β , but not α . Hence, the new dalograph complies with the requirements. \square

The A-transitive closure of a binary relation is its smallest A-transitive superrelation. It is inductively defined below.

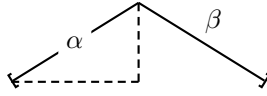
Definition 121 (A-transitive closure)

$$\frac{\alpha \prec \beta}{\alpha \prec^{st} \beta} \quad \frac{\alpha \prec^{st} \beta \quad u\beta \prec^{st} \gamma}{u\alpha \prec^{st} \gamma}$$

The following result about A-transitivity is a generalisation of the previous result about transitivity.

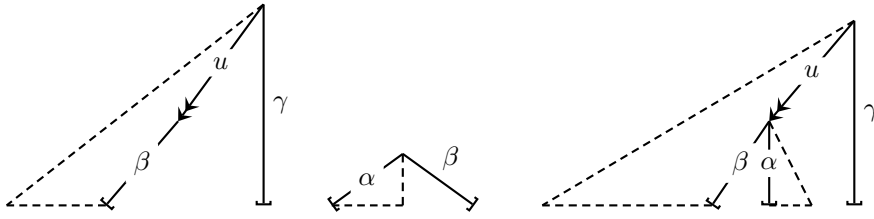
Lemma 122 Let \prec be a preference, and let α and β be such that $\alpha \prec^{st} \beta$. There exists a dalograph g with the following properties.

- The dalograph g has the following shape (dashed lines represent and delimit the rest of the dalograph).



- The path inducing β is not branching after the top node.
- Any equilibrium for g involves the path inducing β .

Proof Proceed by rule induction on the definition of the A-transitive closure. First rule, the subproof is straightforward. Second rule, $u\alpha \prec^{st} \gamma$ comes from $u\beta \prec^{st} \gamma$ and $\alpha \prec^{st} \beta$. By induction hypothesis there exist a dalograph g_1 for $u\beta \prec^{st} \gamma$ and a dalograph g_2 for $\alpha \prec^{st} \beta$. Cut the path inducing β away from g_2 (but the top node), and fuse two nodes as shown below.



By induction hypothesis, the node inducing γ is not branching in g_1 , so it is still not branching in the new dalograph g . In an equilibrium, the node just below u must choose β , by induction hypothesis. So the top node must involve the path inducing γ , also by induction hypothesis. \square

The gen-E-prefix closure is a generalisation of the E-prefix closure.

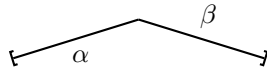
Definition 123 (gen-E-prefix closure)

$$\frac{\alpha \prec \beta}{\alpha \prec^{sep} \beta} \quad \frac{W\alpha \prec^{sep} W\beta}{\alpha \prec^{sep} \beta}$$

The following lemma is a step towards a generalisation of lemma 118 about E-prefix closure.

Lemma 124 Let \prec be a preference. For any $\alpha \prec^{sep} \beta$ there exists a dalograph g with the following properties.

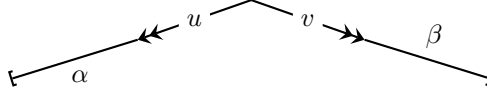
- The dalograph below is a subgraph of the dalograph g .



- Aside from the top node, the paths inducing α and β are not branching.
- Any equilibrium for g involves the path inducing β .

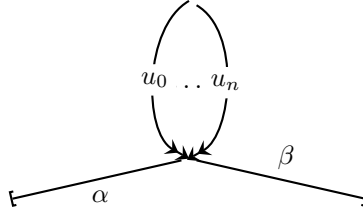
Proof By rule induction on the definition of \prec^{sep} . First case, $\alpha \prec^{sep} \beta$ comes from $\alpha \prec \beta$. Straightforward. Second case, $\alpha \prec^{sep} \beta$ comes from $W\alpha \prec^{sep} W\beta$. By definition of $W\alpha \prec^{sep} W\beta$, for all u in W there exists v in W such that $u\alpha \prec v\beta$. So by induction hypothesis there exists a dalograph $g_{u,v}$ with the following properties.

- The dalograph below is a subgraph of the dalograph $g_{u,v}$.



- Aside from the top node, the paths inducing $u\alpha$ and $v\beta$ are not branching.
- Any equilibrium for $g_{u,v}$ involves the path inducing $v\beta$.

Consider all these dalographs $g_{u,v}$ for u in W . Fuse their top nodes into one node. Also fuse their nodes from where either α or β starts into one single node. This yields a dalograph g' with the following as a subgraph.

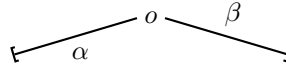


Aside from the central node, the paths inducing α and β are not branching. Each u and v are represented in the u_i , so any equilibrium for g' involves the path inducing β . \square

The following lemma states that under some conditions (similar to the conclusions of the lemmas above) equilibrium existence is preserved by superrelation.

Lemma 125 Let \prec and \prec' be two preferences. Assume that \prec is included in \prec' and that for any $\alpha \prec' \beta$, there exists a dalograph g with the following properties.

- The dalograph below is a subgraph of the dalograph g .

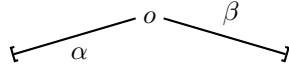


- The path inducing β is not branching after the top node o .
- Any \prec -equilibrium for g involves the path inducing β .

In this case, if all dalographs have \prec -equilibria, then all dalographs have \prec' -equilibria.

Proof Let g be a dalograph. For each 3-uple (o', α, β) such that $\alpha \prec' \beta$ and α and β are eligible at node o' in g , do the following. By assumption, there is a dalograph $g_{o,\alpha,\beta}$ complying with the requirements below.

- The dalograph below is a subgraph of the dalograph $g_{o,\alpha,\beta}$.



- The path inducing β is not branching after the top node o .
- Any equilibrium for $g_{o,\alpha,\beta}$ involves the path inducing β .

Define $g'_{o,\alpha,\beta}$ by cutting away from $g_{o,\alpha,\beta}$ the path inducing β , but leaving the top node o . Fuse the node o' from g and the node o from $g'_{o,\alpha,\beta}$. Let g' be the dalograph built from g after addition of all $g'_{o,\alpha,\beta}$. Let s' be a \prec -equilibrium for g' . Let us consider again any 3-uple (o', α, β) such that $\alpha \prec' \beta$ and α and β are eligible at node o' in g . By construction at node o' , s' does not induce any sequence that is eligible at the top node o of $g'_{o,\alpha,\beta}$. More specifically, s' does not induce α at node o' . Since this holds for any of the considered 3-uple, it means that s' is also a \prec' -equilibrium. Removing the parts of s' that corresponds to all the $g'_{o,\alpha,\beta}$ yields a \prec' -equilibrium for g . \square

Thanks to the result above it is now possible to show that, like E-prefix closure, (A-) transitive closure, alt-subcontinuous closure, and gen-E-prefix closure preserve equilibrium existence.

Lemma 126 *If all dalographs have \prec -equilibria, then all dalographs have \prec^{st} -equilibria, \prec^{sc} -equilibria, \prec^{sep} -equilibria.*

Proof By lemma 125 together with 122 and 124. \square

The combination closure of a binary relation is its smallest superrelation that is A-transitive and gen-E-prefix.

Definition 127 (Combination closure)

$$\frac{\alpha \prec \beta}{\alpha \prec^c \beta} \quad \frac{\alpha \prec^c \beta \quad u\beta \prec^c \gamma}{u\alpha \prec^c \gamma} \quad \frac{W\alpha \prec^c W\beta}{\alpha \prec^c \beta}$$

The combination closure preserves equilibrium existence, as stated below.

Theorem 128 *If all dalographs have \prec -equilibria, all dalographs have \prec^c -equilibria.*

Proof Let \prec be a preference that guarantees existence of equilibrium. Let g be a dalograph and let S be the finite set of all pairs of sequences that are eligible in g . The A-transitive closure and gen-E-prefix closure are simple closures, and they preserve equilibrium existence by lemmas 126. Moreover, the combination closure is their union, so by lemma 116, the restriction to S of the full closure of \prec also guarantees existence of equilibrium. Since $\prec^c|_S$ -equilibrium is also a \prec^c -equilibrium by lemma 160, this allows concluding. \square

6.7 Sufficient Condition and Necessary Condition

After a synthesis of and a discussion about the results obtained so far, this section gives a non-trivial example of a relation that does not meet the requirement of the necessary condition for equilibrium existence. Finally, a further result on network routing application is given.

6.7.1 Synthesis

Firstly, this subsection gathers the main results of this chapter that concern existence of equilibrium. Secondly, It points out that if the preference is a total order, then the sufficient condition and the necessary conditions coincide. Finally, it shows that the necessary condition is not sufficient in general.

The following theorem presents the sufficient condition and the necessary condition for every dalograph to have an equilibrium. The sufficient condition involving the notion of strict weak order is written with few words. However it is difficult to compare it with the necessary condition. Therefore, the sufficient condition is rewritten in a way that enables comparison.

Theorem 129

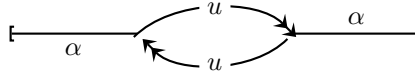
The preference \prec is included in some \prec' .
The preference \prec' is an E-prefix and subcontinuous strict weak order.
 \Downarrow
The preference \prec is included in some \prec' .
The preference \prec' is E-prefix, subcontinuous, transitive, and irreflexive.
The non comparability relation \sharp' is transitive.
 \Downarrow
Every dalograph has a \prec -equilibrium.
 \Downarrow
The preference \prec is included in some \prec' .
The preference \prec' is (gen-) E-prefix, (A-) transitive, and irreflexive.

Proof The topmost two propositions are equivalent by lemma 60, and they imply the third proposition by lemma 80. For the last implication, assume that all dalographs have \prec -equilibria. By theorem 128, all dalographs have \prec^c -equilibria. So \prec is irreflexive, otherwise the reflexive witness alone allows building a game without \prec^c -equilibrium. In addition, \prec^c is E-prefix and A-transitive, and \prec is included in \prec^c by construction. \square

When the preference is a strict total order, the following corollary proves a necessary and sufficient condition for all dalographs to have equilibria.

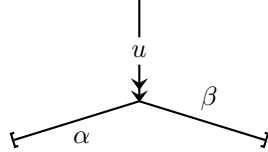
Corollary 130 *Let a preference be a strict total order. All dalographs have equilibria iff the preference is E-prefix and subcontinuous.*

Proof Left-to-right implication: by theorem 167, the strict total order is gen-E-prefix and A-transitive, so it is E-prefix and transitive. If \prec is not subcontinuous, $u^\omega \prec \alpha \prec u\alpha$ for some u and α . So u is non-empty and $\alpha \prec uu\alpha$, and the following dalograph has no equilibrium.



The right-to-left implication follows directly theorem 167 because a strict total order is a strict weak order. \square

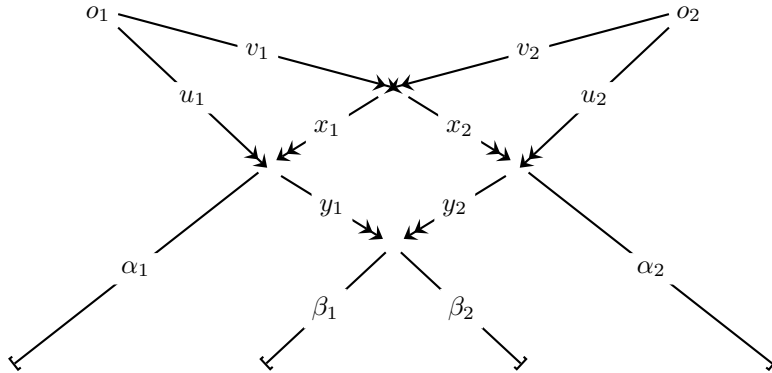
There is a direct proof of this corollary, some parts of whose are much simpler than the proof of theorem 167. For instance, if a total order is E-prefix, then its negation is also E-prefix. This ensures that any maximal path is also semi-hereditary maximal. Therefore the definition of the seeking-forward function is not needed. The necessary condition is much simpler too. Indeed, if \prec is not E-prefix, we have $u\alpha \prec u\beta$ and $\alpha \not\prec \beta$ for some u, α and β . By the first assumption we have $\alpha \neq \beta$, so $\beta \prec \alpha$ by total ordering. Therefore the following dalgograph has no equilibrium.



In general, the necessary condition is not a sufficient condition, as shown by the following two examples. First, let \prec be defined as followed.

$$\begin{array}{ll}
 u_1y_1\beta_2 \prec v_1x_1\alpha_1 & u_2y_2\beta_1 \prec v_2x_2\alpha_2 \\
 v_1x_2\alpha_2 \prec u_1\alpha_1 & v_2x_1\alpha_1 \prec u_2\alpha_2 \\
 v_1x_1y_1\beta_2 \prec u_1\alpha_1 & v_2x_2y_2\beta_1 \prec u_2\alpha_2 \\
 v_1x_2y_2\beta_2 \prec u_1\alpha_1 & v_2x_1y_1\beta_1 \prec u_2\alpha_2
 \end{array}$$

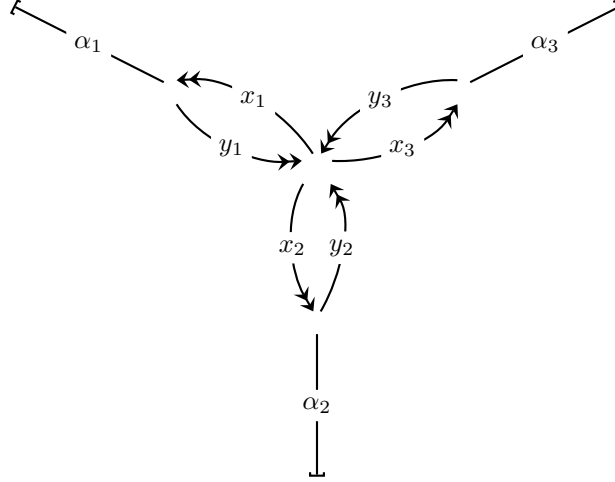
The preference \prec complies with the necessary condition but the dalgograph below has no \prec -equilibrium. Indeed, the node o_1 "wants" to follow a path leading to α_1 or β_1 , while the node o_2 "wants" to follow a path leading to α_2 or β_2 .



Second example, let \prec be defined as followed.

$$\begin{array}{lll}
 \alpha_1 \prec y_1x_2\alpha_2 & \alpha_2 \prec y_2x_3\alpha_3 & \alpha_3 \prec y_3x_1\alpha_1 \\
 (x_1y_1)^\omega \prec x_3\alpha_3 & (x_2y_2)^\omega \prec x_1\alpha_1 & (x_3y_3)^\omega \prec x_2\alpha_2
 \end{array}$$

The preference \prec complies with the necessary condition but the dalograph below has no \prec -equilibrium. Indeed, the situation looks like it is in the jurisdiction of alt-subcontinuity, but it is not.



It is possible to design closures that rule out the above "annoying" situations. For instance, the closure suggested below (by mutual induction) may take care of the triskele example (and of any related example with n branches). However, this kind of incremental procedure is very likely to leave out some more complex examples.

$$\begin{array}{c}
 \frac{\alpha \prec \beta}{\alpha \prec^l \beta} \quad \frac{\alpha \prec^l v\beta \quad (uv)^\omega \prec^l \gamma}{u\alpha \prec^{ll} \beta, \gamma} \quad \frac{\alpha \prec^{ll} \beta}{\alpha \prec^l \beta} \\
 \\
 \frac{A \prec^{ll} B \quad B \prec^{ll} C}{A \prec^{ll} C} \quad \frac{A \prec^l B}{A, C \prec^l B, C} \quad \frac{A, C \prec^{ll} B, C}{A \prec^l B}
 \end{array}$$

6.7.2 Example

The max-min limit-set order is defined in subsection 6.4.2, whereas the max-min set order is defined below. It does not only consider the limit set of the sequence, but every element occurring in the sequence.

Definition 131 (Max-min set order) Let (E, \prec) be a set equipped with a strict weak order. For α an infinite sequence over E , let S_α be the set of all elements occurring in α .

$$\alpha \prec^{Mms} \beta \triangleq S_\alpha \prec^{Mm} S_\beta$$

This order cannot guarantee existence of global equilibrium, as stated below.

Lemma 132 *There exists (E, \prec) a set equipped with a strict weak order, such that there exists a dalograph that is labelled with elements in E , and that has no global equilibrium with respect to the max-min set order.*

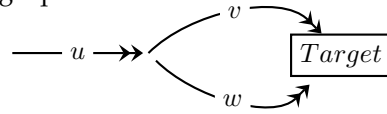
Proof Along the usual order over the figures 0, 1 and 2, we have $2(02)^\omega \prec^{Mms} 21^\omega$, since $\{0, 2\} \prec^{Mm} \{1, 2\}$. However, when removing the first 2 of these two sequences, we have $1^\omega \prec^{Mms} (02)^\omega$ since $\{1\} \prec^{Mm} \{0, 2\}$. Therefore any E-prefix and transitive relation including \prec^{Mms} is not irreflexive. Conclude by theorem 167. \square

6.7.3 Application to Network Routing

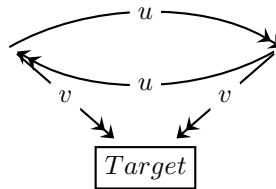
In the total order case, the necessary and sufficient condition for equilibrium in dalographs yields a necessary and sufficient condition for routing equilibrium in routing problems. The necessary condition implication invokes constructive arguments that are similar to the ones used for the necessary condition in dalographs. However, the proof is simple enough so that just doing it is more efficient than applying a previous result.

Theorem 133 *Assume a routing policy \prec^r that is a total order. Then every routing problem has a routing equilibrium iff the policy is E-prefix and $uv \prec^r v$ for all v and non-empty u .*

Proof Left-to-right: by contraposition, assume that either \prec^r is not E-prefix or there exists u and v such that $v \prec^r uv$. First case, \prec^r is not E-prefix. So there exists u, v and w such that $uv \prec^r uw$ and $w \prec^r v$. So the following routing problem has no routing equilibrium.



Second case, there exists u and v such that $v \prec^r uv$. The following routing problem has no routing equilibrium.



The right-to-left implication follows lemma 106. \square

6.8 Conclusion

Consider a collection of labels and a binary relation, called preference, over ultimately periodic sequences over these labels. This chapter shows that if the preference is an E-prefix and subcontinuous strict weak order, then all dalographs labelled with those labels have equilibria with respect to this preference.

This sufficient condition is proved by a recursively-defined seeking-forward function followed by a proof by induction on the number of arcs in a dalograph. A necessary condition is also proved thanks to the notion of simple closure and the design of some simple closures, the union of which preserves equilibrium existence. Some examples show that the necessary condition is not sufficient in general. However, a few examples show the usefulness of both the sufficient and the necessary conditions. A detailed study shows that the necessary condition plus the subcontinuity plus the transitivity of the incomparability relation implies the sufficient condition. Because of this, the two conditions coincide when the preference is a strict total order, which could also be found by a direct proof. However for now, there is no obvious hint saying whether or not the sufficient condition is also necessary.

This chapter applies its theoretical results to a network routing problem: first, the above-mentioned sufficient condition yields a sufficient condition on routing policy for routing equilibrium existence in a simple routing problem. Second, the above-mentioned necessary and sufficient condition of the total order case also yields a necessary and sufficient condition on a total order routing policy for routing equilibrium existence in a simple routing problem.

This chapter is also useful to one other respect: many systems that are different from dalographs also require a notion of preference. In a few of these systems, preferences may be thought as total orders without a serious loss of generality: in these systems, any preference that guarantees equilibrium existence is included in some total order also guaranteeing equilibrium existence, and equilibrium existence is preserved by subrelation. In such a setting, considering only total orders somehow accounts for all binary relations. However in the case of dalographs, there might exist a preference guaranteeing equilibrium existence, such that any linear extension of the preference does not guarantee equilibrium existence. In this case, assuming total ordering of the preference would yield a (non-recoverable) loss of generality. The following example is a candidate for such a preference. Consider the ultimately periodic sequences over $\{a, b, c, d\}$. An A-transitive preference \prec over these sequences is defined below.

$$\begin{aligned} a^\omega &\prec cb^\omega \prec da^\omega \\ db^\omega &\prec ca^\omega \prec b^\omega \end{aligned}$$

The preference \prec defined above is (A-) transitive and (gen-) E-prefix. In addition, it is not included in any transitive and E-prefix total order. Indeed, let $<$ be such a total order. If $a^\omega < b^\omega$ then $da^\omega < db^\omega$ by E-prefix and total ordering, so $cb^\omega < ca^\omega$ by transitivity, so $b^\omega < a^\omega$ by E-prefix, contradiction. If $b^\omega < a^\omega$ then $ca^\omega < cb^\omega$ by transitivity, so $a^\omega < b^\omega$ by E-prefix, contradiction. So, the key question is whether or not the preference \prec guarantees equilibrium existence for all dalographs (this is not proved in this chapter).

Chapter 7

Sequential Graph Games

Abstract sequential tree games generalise traditional sequential tree games, by abstraction over the real-valued payoff functions, while keeping the tree structure. This permits generalising Kuhn's result, which states that every traditional sequential game has a Nash equilibrium. This generalisation also uses the concept of "backward induction" already used by Kuhn. The main purpose of this chapter is to define games whose structures are more general than trees, but where "backward induction" still allows building an equilibrium. For this, the chapter generalises further the concept of abstract sequential game by using a graph structure. In such a setting, local and global equilibria are defined. They generalise Nash and subgame perfect equilibria respectively. For a subclass of sequential graph games, the chapter presents both a necessary condition and a sufficient condition for global equilibrium existence. These conditions relate to the necessary condition and sufficient condition for equilibrium existence in dalographs. Moreover, the sufficient condition adapts and uses "backward induction", which is the aim of this chapter. In addition, the concept of global equilibrium makes clear that subgame perfect equilibria of a sequential tree game are the Nash equilibria of some derived abstract game. More concretely, an embedding identifies the subgame perfect equilibria of a traditional sequential game with the Nash equilibria of a derived strategic game.

7.1 Introduction

In the first place, the notion of Nash equilibrium relates to strategic games with real-valued payoff functions. However, there exists a natural embedding of sequential games with real-valued payoff functions into strategic games with real-valued payoff functions. A strategy profile (of a sequential game) whose image by the embedding is a Nash equilibrium is also called a Nash equilibrium. Kuhn [25] proved in 1953 that every traditional sequential game has a Nash equilibrium. For this he used a procedure named "backward induction". Lately in [29], the notion of sequential game was generalised by replacing real-valued payoff functions with abstract atomic objects, called outcomes, and by replacing the usual total order over the reals with arbitrary binary relations over outcomes, called preferences. This introduces a general abstract

formalism where Nash equilibrium, subgame perfect equilibrium, and “backward induction” can still be defined. Using a lemma on topological sorting, the above-mentioned paper proves that the following three propositions are equivalent: 1) Preferences over the outcomes are acyclic. 2) Every sequential game has a Nash equilibrium. 3) Every sequential game has a subgame perfect equilibrium. The result was fully computer-certified using Coq.

7.1.1 Graphs and Games

Traditional game theory seems to work mainly with strategic games and sequential games, *i.e.* games whose underlying structure is either an array or a rooted tree. These game can involve many players. On the contrary, combinatorial game theory studies games with various structures, for instance games in graphs. It seems that most of these combinatorial games involve two players only. Moreover, the possible outcomes at the end of most of these games are “win-lose”, “lose-win”, and “draw” only. The book [7] presents many aspects of combinatorial game theory.

Chess is usually thought as a sequential tree game. However, plays in chess can be arbitrarily long (in terms of number of moves) even with the fifty moves rules which says that a player can claim a draw if no capture has been made and no pawn has been moved in the last fifty consecutive moves. So, the game of chess is actually defined through a directed graph rather than a tree, since a play can enter a cycle. Every node of the graph is made of both the location of the pieces on the chessboard and an information about who has to move next. The arcs between the nodes correspond to the valid moves. So the game of chess is a bipartite digraph (white and black play in turn) with two agents. This may sound like a detail since the game of chess is well approximated by a tree. However, this is not a detail in games that may not end for intrinsic reasons instead of technical rules: for instance poker game or companies sharing a market. In both cases, the game can continue as long as there are at least two players willing to play. In the process, a sequence of actions can lead to a situation similar to a previous situation (in terms of poker chips or market share), hence a cycle.

A country with several political parties is a complex system. Unlike chess and poker, rules may not be definable by a digraph, but at least the system may be modelled by a digraph: The nodes represent the political situations of the country, *i.e.* which party is in power at which level, etc. Each node is also labelled with the party that has to take the next decision, *i.e.* vote a law, start a campaign, etc. The possible decision are represented by arcs from the node to other nodes where other parties have to take other decisions. The arcs are labelled with the short-term outcomes of the decision. This process may enter a cycle when a sequence of actions and elections leads to a political situation that is similar to a previous situation, *i.e.* the same parties are in power at the same levels as before.

Internet can be seen as a directed graph whose nodes represent routers and whose arcs represent links between routers. When receiving a packet, a router has to decide where to forward it to: either to a related local network or to another router. Each router chooses according to “it’s owner interest”. Therefore Internet can be seen as a digraph with nodes labelled with owners of routers. This digraph is usually symmetric since a link from router A to router B can

be easily transformed to a link between router b and router A . Moreover, the interests of two different owners, *i.e.* Internet operators, may be contradictory since they are supposed to be competitors. Therefore the owners' trying to maximise their benefits can be considered a game. Local benefits (or costs) of a routing choice may be displayed on the arcs of the graph.

7.1.2 Contribution

This chapter introduces the notion of sequential graph game. Such a game involves agents and outcomes. A sequential graph game is a directed graph whose nodes are labelled with agents, whose arcs are labelled with outcomes, and each of whose node has an outgoing arc. The design choices are quickly justified, and an interpretation of these games is proposed through an informal notion of play. A strategy profile for a sequential graph game amounts to choosing an outgoing arc at every node of the game. By changing these choices only at some nodes that he owns, an agent can convert a strategy profile into another one; this defines convertibility. Starting from a given node, one can follow the arcs that are prescribed by a given strategy profile. This induces an infinite sequence of outcomes. For each agent, a binary relation accounts for the agent's preferences among infinite sequences of outcomes. Given a node of a sequential graph game, an agent can compare two strategy profiles for the game by comparing the induced infinite sequences at the given node; this defines preference. Having a notion of convertibility and preference for each agent, the local equilibria at given nodes are defined like the Nash equilibria of some derived C/P game: they are strategy profiles that no agent can convert into a preferred profile. A global equilibrium is defined as a strategy profile that is a local equilibrium at every node of the underlying game.

It turns out that the global equilibria of a sequential graph game are exactly the Nash equilibria of some derived C/P game that is different from the C/P game mentioned above. In addition, the chapter defines an embedding of sequential tree games into sequential graph games. This embedding sends Nash equilibria to local equilibria and vice versa, and sends subgame perfect equilibria to global equilibria and vice versa. Therefore, local equilibrium is a generalisation of Nash equilibrium and global equilibrium is a generalisation of subgame perfect equilibrium.

In sequential tree games, subgame perfect equilibria can be built through "backward induction" following topological sorting. This chapter generalises the procedure of "backward induction" for a subclass of sequential graph games. This leads to a sufficient condition on the agents' preferences for global equilibrium existence in every game in the subclass mentioned above. It thus generalises the generalisation [29] of Kuhn's result [25], which states that every sequential game has a Nash (and subgame perfect) equilibrium. In addition, the chapter gives a necessary condition on the agents' preferences for global equilibrium existence in every game in the subclass mentioned above. For the necessary condition and the sufficient condition, which do not coincide in general, the chapter invokes some results about dalographs proved in chapter 6. However, the two conditions coincide when the preferences are total orders, which gives an equivalence property. In the same way, a sufficient condition is given for equilibrium existence in every sequential graph game.

As mentioned above, the subgame perfect equilibria of a game are the global

equilibria of another game, and the global equilibria of a game are the abstract Nash equilibria of yet another game. Therefore the subgame perfect equilibria of a game are the abstract Nash equilibria of a derived game. Working in an abstract setting helps think of the previous remark, but there is a direct (and less obvious) argument. Indeed, there exists a second embedding of traditional sequential games into traditional strategic games, with the following property: the subgame perfect equilibria of the sequential game are exactly the Nash equilibria of the strategic game. Compared to the traditional embedding, this second embedding just changes the payoffs by giving weights to payoffs induced by all the subprofiles. This has the following consequence: on the one hand, the Nash equilibria of a sequential game were named so because they are the Nash equilibria of some strategic game obtained by the traditional embedding. On the other hand, the subgame perfect equilibria of a sequential game are the Nash equilibria of some other strategic game obtained by the second embedding. Therefore, subgame perfect equilibria could have been named Nash equilibria!

7.1.3 Contents

Section 7.2 defines the notion of sequential graph game, play (informally), strategy profile, and other concepts needed to define local and global equilibrium. Section 7.3 embeds sequential tree games into sequential graph games and also show that subgame perfect equilibria are the Nash equilibria of some derived strategic game. Section 7.4 concerns equilibrium existence in sequential graph games. It presents the generalisation of "backward induction" in graphs.

7.2 Sequential Graph Games and Equilibria

Subsection 7.2.1 introduces the notion of sequential graph games; subsection 7.2.2 interprets informally what would be a play in such games; and subsection 7.2.3 defines the notions of local and global equilibria.

7.2.1 Sequential Graph Games

This subsection introduces the notion of sequential graph games. It is meant to capture a wide range of games where some agents play in turn within an underlying structure that resembles a finite graph. The design choices are quickly justified: sequential graph games seem to be general enough, but not too much, and thus seem to suit the purpose of this chapter.

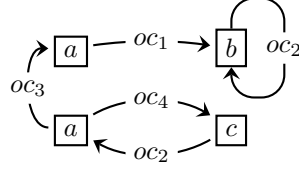
Consider a collection of agents and a collection of outcomes. A sequential graph game is a finite directed graph whose nodes are labelled with agents, whose arcs are labelled with outcomes, and such that every node has a non-zero outdegree, *i.e.* every node has an outgoing arc.

Definition 134 (Sequential graph game) *A sequential graph game is defined by a triple $\langle E, vl, el \rangle$ complying with the following.*

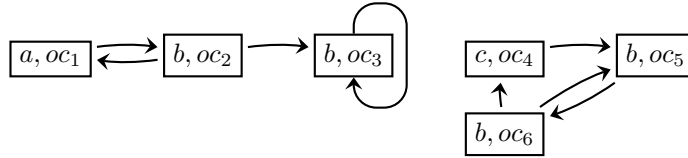
- $E \subseteq V \times V$ is a set of arcs, where V is a non-empty finite set of nodes. Moreover, E is sink-free: $\forall n \in V, \exists n' \in V, (n, n') \in E$.

- vl is a function of type $V \rightarrow \mathcal{A}$, where \mathcal{A} is a non-empty set of agents.
- el is a function of type $E \rightarrow Oc$, where Oc is a non-empty set of outcomes.

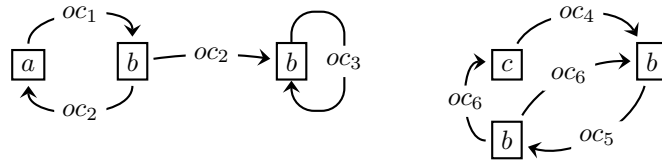
The picture below is an example of sequential graph game involving three agents a , b and c , and four outcomes from oc_1 to oc_4 .



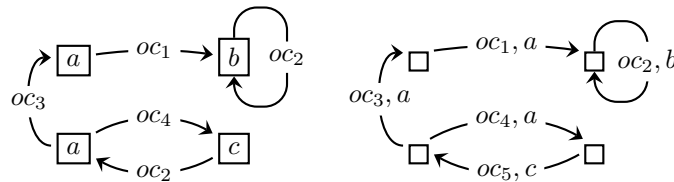
One may argue that labelling the arcs with the outcomes loses generality by leaving out the digraphs where outcomes label the nodes, like the example below.



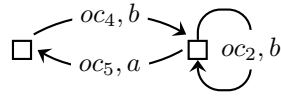
However, there is a natural way of embedding node-only-labelled digraphs into sequential graph games: for each node, remove the outcome and put it on all the outgoing arcs of the node. The picture below shows the example above being translated into sequential graph games.



Due to the above-mentioned embedding reason, arc labelling appears to be more general than node labelling. One may want to apply this remark to agents, and thus label arcs with both outcomes and agents. This procedure would transform the left-hand sequential graph game below into the right-hand graph.

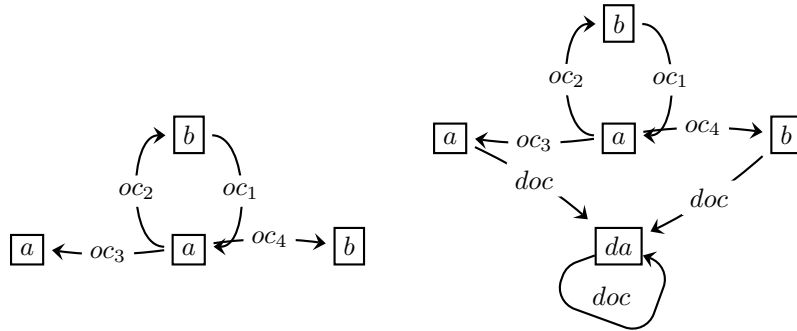


Considering all such digraphs whose arcs are labelled with both outcomes and agents is too general. Indeed, it also requires to consider the following example. In this example, a node has two outgoing arcs that are labelled with different agents.

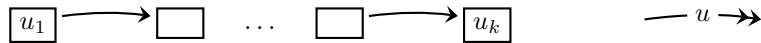


However, this chapter studies games where agents explicitly play in turn. This means that at a given node the decision to use a specific arc must be made by one and only one agent. Therefore the definition of sequential graph games must keep agents inside the nodes.

Demanding that all nodes have a non-zero outdegree may not yield a serious loss of generality either. Indeed, there is also an embedding from graphs without the non-zero-outdegree constraint into sequential graph games. The embedding uses a dummy node controlled by a dummy agent da , and a dummy outcome doc labelling a few dummy arcs. For instance, the left-hand graph below is transformed into the right-hand sequential graph game.



It will sometimes be convenient to represent walks in a sequential graph game partially, without even displaying the agents. A (possibly empty) non-looping walk is represented by a double-headed arrow that is labelled with a sequence of nodes. For instance, the left-hand walk $u = u_1 \dots u_k$ below is represented by the right-hand picture.



A looping walk is represented by a squared-bracket-headed arrow that is labelled with a sequence of nodes. For instance below, the left-hand looping walk $unvn$ is represented by the right-hand picture.

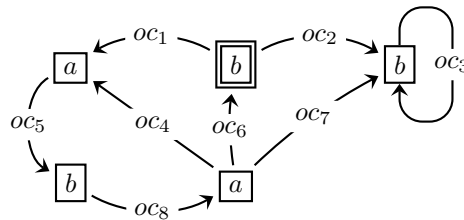


7.2.2 Plays

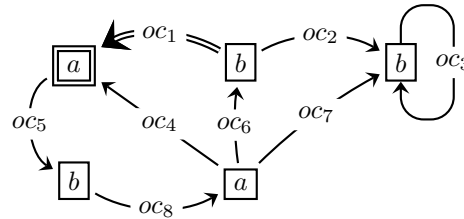
This subsection describes the informal notion of play in sequential graph games, and thus gives an informal interpretation of sequential graph games. The definition of play is quickly justified: plays in sequential graph games seem to be

general enough, but not too much, and thus seem to suit the purpose of this chapter.

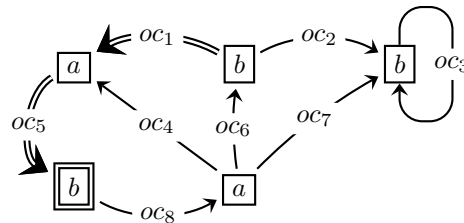
The interpretation of sequential graph games is as follows. A play starts at one node of the game. Any node will do. The agent owning the current node chooses the next move among the non-empty collection of outgoing arcs at that node. This way, the play reaches another node, and so on. There is one constraint to such a play: when reaching a node that has been already visited during the play, the owner of the node must be consistent with his previous choice and follow the same arc as at his first visit to that node. The example below describes such a play. At any stage of the play, the sole current node is double-squared and choices/moves are represented by double lines instead of simple lines. The history of choices is remembered during the play. One possible play starts at the double-squared node below.



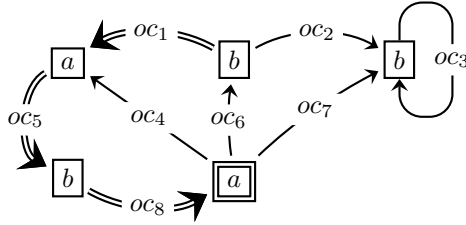
Above, agent b has to choose between left and right, *i.e.* between oc_1 and oc_2 . In this example, he chooses left and the play reaches a node owned by agent a , as displayed below.



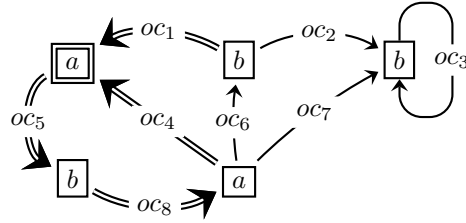
In turn, agent a has to choose the next move in the situation above. Only one option is available to agent a so he "chooses" it, as shown below.



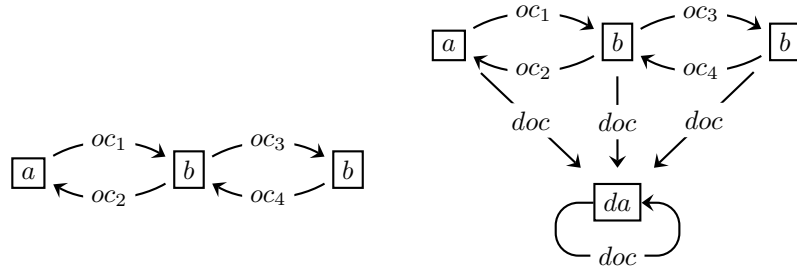
Same scenario for agent b , and the play reaches a node where agent a has to choose between three options.



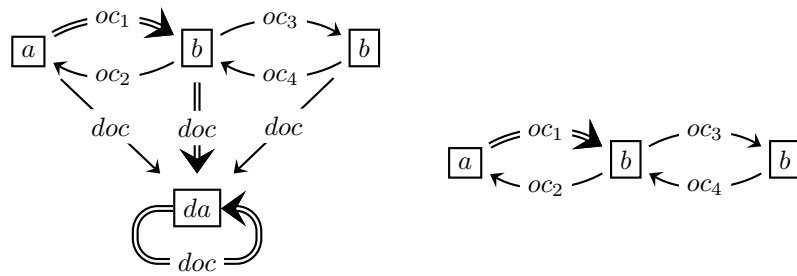
Agent a chooses left, *i.e.* oc_4 , and the play reaches a node already visited before. So the play continues in a loop as prescribed by the history of choices.



It is possible to deal with finite "play", *i.e.* non-self-intersecting paths, within the sequential graph game formalism: given a sequential graph game, first add to it a dummy node that is owned by a dummy agent. Then add dummy arcs, labelled with a dummy outcome, from all original nodes to the dummy one. According to this procedure, the left-hand sequential graph game below is transformed into the right-hand sequential graph game.



Through the above translation, the left-hand infinite play below (pertaining to the right-hand sequential graph game above) may be interpreted as the right-hand "finite play" below (pertaining to the left-hand sequential graph game above).



Thanks to the embedding mentioned above, the non-zero-outdegree constraint may not yield a serious loss of generality.

7.2.3 Equilibria

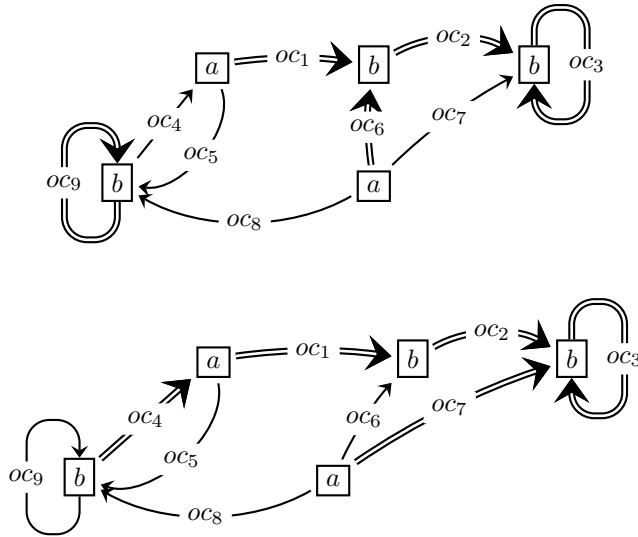
This subsection first extends the notion of strategy profile to the sequential graph game formalism. In the first place, the concept of strategy profiles pertains to strategic games. This concept was already extended to sequential tree games in traditional game theory. The extension of this chapter actually follows the idea of the traditional extension. This subsection also extends the notion of conversion ability of agents. Then, the subsection defines the notions of induced sequence of outcomes and preference over these, and the notions of local happiness and local/global equilibrium. Finally, the local (resp. global) equilibria of a sequential graph game are shown to be exactly the abstract Nash equilibria of some derived C/P game that is equipped with a Cartesian product structure.

A strategy profile is an object built on a sequential graph game by choosing an outgoing arc at each node.

Definition 135 (Strategy profile) *A strategy profile for a sequential graph game $g = \langle E, vl, el \rangle$ is a pair (g, c) , where c is a function of type $V \rightarrow V$ complying with the following formula.*

$$\forall n \in V, (n, c(n)) \in E$$

The two examples below show two (strategy) profiles for the same underlying game. The choices are represented by double lines.



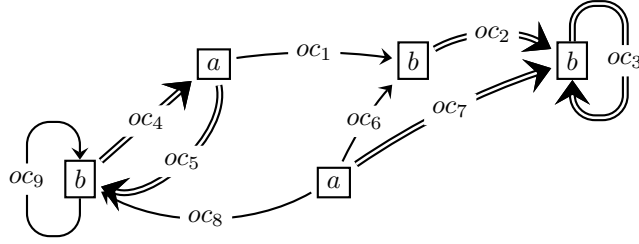
Informally, each agent is granted the ability to change his choices at any node that he owns. Such changes do not occur during a play in a dynamic way, but before a play in a static way, when choosing strategies. If two strategy profiles s and s' are different only by the choices of a given agent a , then the

two strategy profiles are said convertible (one to each other) by agent a . Convertibility is therefore an equivalence relation. This is formally defined below.

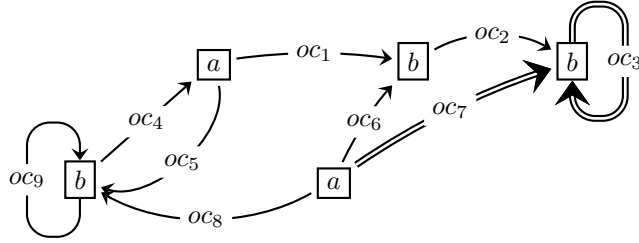
Definition 136 (Convertibility) Let $g = \langle E, vl, el \rangle$ be a sequential graph game. Let $s = (g, c)$ and $s' = (g, c')$ be two strategy profiles for g . Convertibility by agent a of s and s' is defined as follows.

$$s \xrightarrow{+a} s' \triangleq \forall n \in V, c(n) \neq c'(n) \Rightarrow vl(n) = a$$

For instance, the two profiles above are not convertible one to each other. Neither by a nor by b . However, agent a can convert the profile above into the profile below.



Informally, the play induced at one given node of a given strategy profile is the sequence of nodes obtained when starting from the given node and following what the strategy profile prescribes. For instance, the above strategy profile induces the play below, which starts from a node owned by agent a .

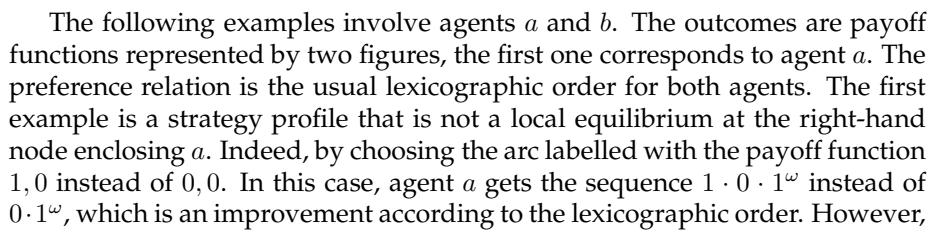


In turn, a given play induces an (ultimately periodic) infinite sequence of outcomes. These outcomes are the labels of the arcs that are visited during the play. This is defined below.

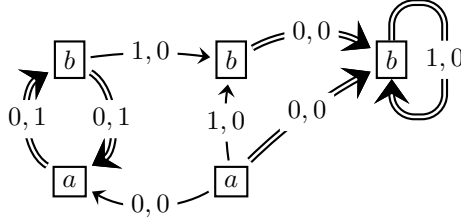
Definition 137 (Induced sequence of outcomes) Let $s = (g, c)$ be a strategy profile for a sequential graph game $g = \langle E, vl, el \rangle$. The infinite sequence of outcomes induced by s starting from n is defined below.

$$seq(s, n) \triangleq el(n, c(n)) \cdot seq(s, c(n))$$

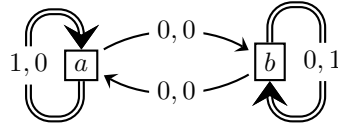
In the following example, the induced sequences at the nodes owned by agent a are $(oc_5 \cdot oc_4)^\omega$ and $oc_7 \cdot oc_3^\omega$.



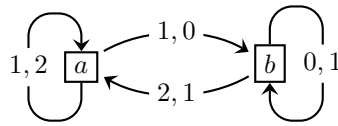
the first example below is a local equilibrium at the leftmost node enclosing b , as well as at the other nodes.



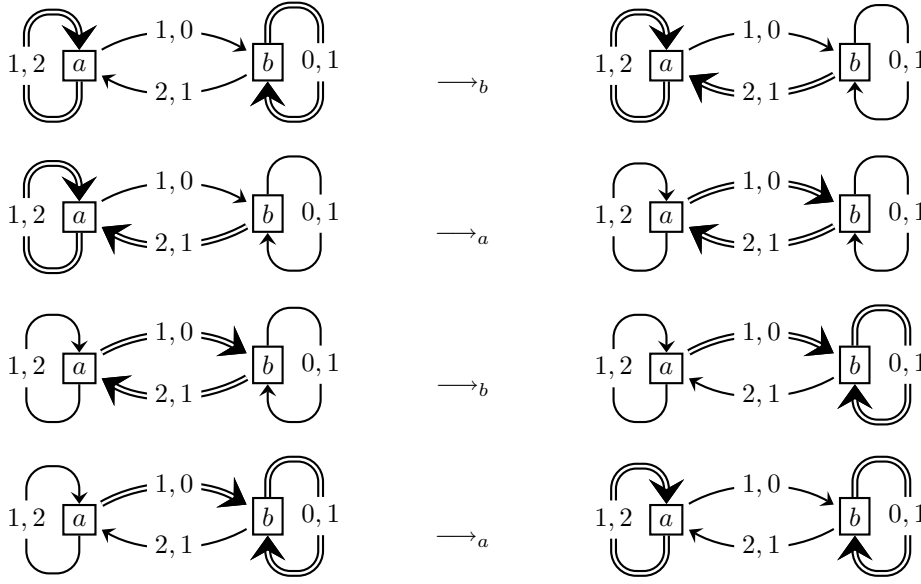
The following strategy profile is a global equilibrium. Indeed, if a changes his choice at the only node that he owns, then from the same node he gets the sequence 0^ω instead of 1^ω , which is a worsening. Same scenario for agent b .



The following sequential graph game has no global equilibrium, as explain thereafter.



Indeed, the game above has four corresponding strategy profiles since each agent has two possible choices. Each of the profiles, on the left-hand side below, makes one agent unhappy, *i.e.* able to improve upon the induced play at some node (actually the node that he owns). First profile: b gets 1^ω but can get $1 \cdot 2^\omega$ by changing his choice. Second profile: a gets 1^ω but can get $(1 \cdot 2)^\omega$ by changing his choice. Third profile: b gets $(1 \cdot 0)^\omega$ but can get 1^ω by changing his choice. Fourth profile: a gets $1 \cdot 0^\omega$ but can get 1^ω by changing his choice.



The notions of local equilibrium and global equilibrium are actually both instances of the concept of abstract Nash equilibrium. While it is clear for the local equilibrium, whose definition has a shape similar to the definition of abstract Nash equilibrium in C/P games, it is less clear for global equilibrium. The following presentation intends to clarify this.

An agent globally prefers a strategy profile over another one if he locally prefers the first one for some node.

Definition 142 (Global preference)

$$s \xrightarrow{a} s' \triangleq \exists n, s \xrightarrow{n} s'$$

An agent is globally happy with a current strategy profile if he cannot convert this current profile into another one that he globally prefers.

Definition 143 (Global happiness)

$$Happy_{\xrightarrow{\quad}}(a, s) \triangleq \forall s', s \xrightarrow{+a} s' \Rightarrow \neg(s \xrightarrow{a} s')$$

The following lemma states that global equilibrium amounts to global happiness of all agents.

Lemma 144

$$GEq_{\xrightarrow{\quad}}(s) \Leftrightarrow \forall a, Happy_{\xrightarrow{\quad}}(a, s)$$

The lemma above makes clear that the global equilibria of a sequential graph games are the Nash equilibria of a derived C/P game. Actually, the derived C/P game has a Cartesian product structure given by the strategy profiles, so it can be seen as an abstraction of a strategic game.

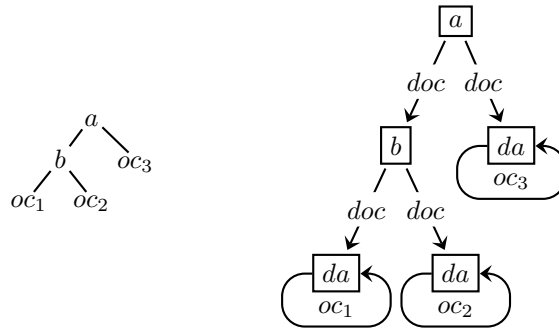
7.3 Sequential Tree/Graph Game and Nash/Subgame Perfect Equilibrium

Subsection 7.3.1 shows that sequential tree games can be considered as sequential graph games. More specifically, an embedding sends Nash equilibria to local equilibria and vice versa, and sends subgame perfect equilibria to global equilibria and vice versa. Subsequently, this subsection notes that the subgame perfect equilibria of a sequential tree game are exactly the Nash equilibria of a derived C/P game that is equipped with a Cartesian product structure. To make this last point more concrete, subsection 7.3.2 embeds sequential tree games with real-valued payoff functions into strategic games with real-valued payoff functions. This embedding sends subgame perfect equilibria to Nash equilibria and vice versa, while the traditional embedding of sequential tree games into strategic games sends game-tree sends (sequential) Nash equilibria to (strategic) Nash equilibria and vice versa. This shows that subgame perfect equilibria could be named (sequential) Nash equilibria instead of the current (sequential) Nash equilibria.

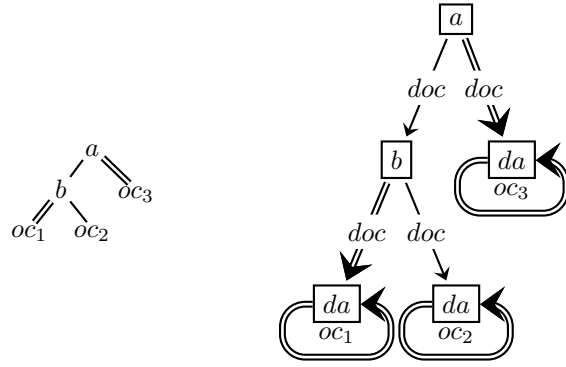
The formal definition of sequential tree games and their equilibria can be found in [29].

7.3.1 Sequential Tree Games are Sequential Graph Games

There is a natural embedding of sequential tree games into sequential graph games. Consider a sequential tree game and transform it as follows. Graphically, the idea is to replace all the edges between nodes by arcs directed away from the root, and to label all those arcs with a dummy outcome *doc*. For each leaf, replace the enclosed outcome with a dummy agent *da*, and put the outcome on an arc looping around the ex-leaf. This translation function is called *t2g*, which stands for tree to graph. The left-hand picture below represents a tree game, and the right-hand picture represents its translation into graph games.



In the same way, there is a natural embedding of strategy profiles of tree games into strategy profiles of graph games. It follows the same idea as for games. In addition, all the double lines representing choices are kept, and double lines are added on all arcs looping around ex-leaves. This translation function is also called *t2g*. The left-hand picture below represents a game-tree strategy profile, and the right-hand picture represents its translation into graph-game strategy profiles.



Conversion ability of an agent is preserved and reflected by the strategy profile translation, as formally stated below.

Lemma 145 (Conversion conservation) Let $\xrightarrow{+a}_t$ (resp. $\xrightarrow{+a}_g$) be the binary relation over game-tree (resp. graph-game) strategy profiles that accounts for conversion ability of agent a . The following formula holds.

$$s \xrightarrow{+a}_t s' \Leftrightarrow t2g(s) \xrightarrow{+a}_g t2g(s')$$

The preferences of the agents are translated as follows.

Definition 146 (Translation of preference) Let \xrightarrow{a} be the preference of agent a in the game-tree setting. The corresponding preference in the graph-game setting is defined below.

$$doc^i oc_1^\omega \xrightarrow{a}_{t2g} doc^j oc_2^\omega \triangleq oc_1 \xrightarrow{a} oc_2$$

The following two lemma compare equilibria in both tree and graph formalisms. They can be proved by induction on the strategy profile s . The notation $Eq(s)$ (resp. $SPE(s)$) means that s is a Nash (resp. subgame perfect) equilibrium. A game-tree strategy profile is a Nash equilibrium *iff* its translation is a local equilibrium for the root.

Lemma 147 (Nash equilibrium is local) When s is a game-tree strategy profile, let $r(s)$ be its root. With this notation, the following formula holds.

$$Eq_{\xrightarrow{a}}(s) \Leftrightarrow LEq_{\xrightarrow{a}_{t2g}}(t2g(s), r(s))$$

A game-tree strategy profile is a subgame perfect equilibrium *iff* its translation is a global equilibrium.

Lemma 148 (Subgame perfect equilibrium is global)

$$SPE_{\xrightarrow{a}}(s) \Leftrightarrow GEq_{\xrightarrow{a}_{t2g}}(t2g(s))$$

The above embedding leads to the following interpretation back to sequential tree games: a Nash equilibrium is a strategy profile that is a local Nash

equilibrium for plays starting at the root of the tree. On the other hand, a subgame perfect equilibrium is a strategy profile that is a local Nash equilibrium wherever the play starts in the tree. The benefit of this interpretation is that no further notion of rationality or epistemic logic is needed.

Furthermore, as seen in subsection 7.2.3, the global equilibria of a sequential graph game are the Nash equilibria of a derived C/P game with Cartesian product structure. Therefore, the subgame perfect equilibria of a sequential tree game are the Nash equilibria of a derived C/P game with Cartesian product structure.

7.3.2 Subgame Perfect Equilibrium and Nash Equilibrium Coincide

To make the last remark of subsection 7.3.1 more concrete, this subsection embeds sequential tree games into strategic games. Unlike the traditional embedding, this embedding allows seeing the subgame perfect equilibria of a sequential tree game as the Nash equilibria of a derived strategic game.

The following definition introduces the notion of global payoff induced by a strategy profile of a sequential tree game. This notion is different from the usual notion of induced payoff. On the one hand, the usual induced payoffs account for the result of a game being played according to the strategy profile. On the other hand, the induced global payoffs accounts for the usual induced payoffs of all "subprofiles" of a given profile. It is done by a linear combination of the usual induced payoffs. This combination gives more weight to the usual induced payoffs of "bigger" subprofiles.

Definition 149 *Given a sequential tree game with real-valued payoff functions, the induced payoff function I is defined inductively for a game-tree strategy profile s . The induced global payoff function GI is also introduced just after I .*

$$\begin{aligned}
 I(sL(pf), a) &\triangleq pf(a) \\
 I(s, a) &\triangleq I(s_k, a) \\
 GI(sL(pf), a) &\triangleq 2^{pf(a)} \\
 GI(s, a) &\triangleq \sum_{1 \leq i \leq n} GI(s_i, a) + \frac{n \times M(s, a)}{r(s, a)} \times 2^{I(s, a)}
 \end{aligned}$$

With the notation $s = sN(b, s_1 \dots s_{k-1}, s_k, s_{k+1} \dots, s_n)$ and $M(s, a) \triangleq \max\{GI(s'_i, a) \mid 1 \leq i \leq n \wedge s_i \xrightarrow{+a} s'_i\}$ and $r(s, a) \triangleq \min\{|2^{I(s', a)} - 2^{I(s'', a)}| \mid I(s', a) \neq I(s'', a) \wedge s \xrightarrow{+a} s' \wedge s \xrightarrow{+a} s''\}$, and $\min(\emptyset) \triangleq +\infty$.

Informally, $r(s, a)$ is the minimal "distance" between the induced payoffs that agent a can obtain by converting the profile s . And $M(s, a)$ is the maximal global payoff that agent a can obtain by considering and converting the substrategy profiles of s . In the definition above, the exponentiation function could be replaced with any strictly increasing function with positive images; at least for the purpose of this subsection.

The following lemma states that the functions M and r are invariant by profile convertibility.

Lemma 150

$$s' \xleftrightarrow{+a} s \Rightarrow M(s', a) = M(s, a) \quad \wedge \quad r(s', a) = r(s, a)$$

Next lemma says that GI is positive, which can be proved by induction on the strategy profile argument.

Lemma 151 $0 < GI(s, a)$.

Now consider two convertible profiles. If one of them induces greater payoff than the other, then it also induces greater global payoff, as shown below.

Lemma 152

$$s' \xleftrightarrow{+a} s \Rightarrow I(s', a) < I(s, a) \Rightarrow GI(s', a) < GI(s, a)$$

Proof Case split on s being or not a compound profile. First case, assume that s has only one node. So $s = s' = sLpf$ for some payoff function pf , so $I(s', a) = I(s, a)$. Second case, let $s = sN(b, s_1 \dots s_{k-1}, s_k, s_{k+1} \dots, s_n)$ be a compound profile and let $s' = sN(b, s'_1 \dots s'_{k'-1}, s'_{k'}, s'_{k'+1} \dots, s'_n)$ be such that $s' \xleftrightarrow{+a} s$. Assume that $I(s', a) < I(s, a)$. Since $2^{I(s', a)} < 2^{I(s, a)}$, we also have $\frac{1}{r(s', a)} \times 2^{I(s', a)} + 1 \leq \frac{1}{r(s, a)} \times 2^{I(s, a)}$ by definition of r and lemma 150. For all i between 1 and n , we have $s'_i \xleftrightarrow{+a} s_i$, so $GI(s'_i, a) \leq M(s, a)$. Therefore $\sum_{1 \leq i \leq n} GI(s'_i, a) \leq n \times M(s, a)$. So $GI(s', a) \leq n \times M(s, a) \times (1 + \frac{1}{r(s', a)}) \times 2^{I(s', a)} \leq n \times M(s, a) \times \frac{1}{r(s, a)} \times 2^{I(s, a)}$. But $GI(s, a) = \sum_{1 \leq i \leq n} GI(s_i, a) + \frac{n \times M(s, a)}{r(s, a)} \times 2^{I(s, a)}$ with $0 < \sum_{1 \leq i \leq n} GI(s_i, a)$ by lemma 151. Hence $GI(s', a) < GI(s, a)$. \square

The following defines an embedding of real-valued payoff function sequential tree game into real-valued payoff function strategic game.

Definition 153 Let g be a sequential tree game. Define $Gsg(g)$ the global strategic game of g as follows. The agents are the same in g and $Gsg(g)$, the strategy profiles are the same, the convertibility relations are the same, and the payoff functions of $Gsg(g)$ are the induced global payoff functions GI of the sequential tree game.

Note that the only difference between the traditional embedding and the embedding above is that the former uses the induced payoff functions I and the latter the global induced payoff functions GI .

Next lemma proves that the above embedding sends subgame perfect equilibria to Nash equilibria (of some strategic games).

Lemma 154 A subgame perfect equilibrium for a sequential tree game g is a Nash equilibrium for $Gsg(g)$.

Proof By structural induction on s , a subgame perfect equilibrium for g . Note that the claim holds for leaf games. Now let $s = sN(b, s_1 \dots s_{k-1}, s_k, s_{k+1} \dots, s_n)$ be a compound profile and assume that the claim holds for subprofiles of s . By definition of subgame perfect equilibrium, s_i is a subgame perfect equilibrium

for all i . So by induction hypothesis, $s_i \xrightarrow{+a} s'_i$ implies $GI(s'_i, a) \leq GI(s_i, a)$ for all i . Assume that $s \xrightarrow{+a} s'$ for some $s' = sN(b, s'_1 \dots s'_{k'-1}, s'_{k'}, s'_{k'+1} \dots, s'_n)$. So $s_i \xrightarrow{+a} s'_i$ for all i , and $GI(s'_i, a) \leq GI(s_i, a)$. As a subgame perfect equilibrium, s is also a Nash equilibrium, so $I(s', a) \leq I(s, a)$. Therefore $GI(s', a) \leq GI(s, a)$ by definition of GI and by invoking lemma 150. So s is a Nash equilibrium for $Gsg(g)$. \square

Nash equilibria are preserved by the embedding, as stated below.

Lemma 155 *Let g be a sequential tree game. A Nash equilibrium for $Gsg(g)$ is a Nash equilibrium for g .*

Proof Let s be a profile for g . Assume that s is a Nash equilibrium for $Gsg(g)$. So by definition, $s \xrightarrow{+a} s'$ implies $GI(s', a) \leq GI(s, a)$ for all profiles s' . Therefore, $s \xrightarrow{+a} s'$ implies $I(s', a) \leq I(s, a)$ by lemma 152. So s is also a Nash equilibrium for g . \square

Consider a strategy profile s for a sequential tree game g . If s is a Nash equilibrium for $Gsg(g)$, then the substrategy profiles of s are Nash equilibria for the images by the embedding of the corresponding subgames of g .

Lemma 156 *Let $s = sN(b, s_1 \dots s_{k-1}, s_k, s_{k+1} \dots, s_n)$ be a strategy profile for a sequential tree game $g = gN(b, g_1, \dots, g_n)$.*

$$Eq_{Gsg(g)}(s) \Rightarrow \forall i, Eq_{Gsg(g_i)}(s_i)$$

Proof Assume $s_i \xrightarrow{+a} s'_i$. Build s' from s by replacing s_i with s'_i . So $s \xrightarrow{+a} s'$, and $GI(s', a) \leq GI(s, a)$ follows. If $i \neq k$ then $0 \leq GI(s, a) - GI(s', a) = GI(s_i, a) - GI(s'_i, a)$. If $i = k$ then $0 \leq GI(s, a) - GI(s', a) = GI(s_k, a) - GI(s'_k, a) + c \times (I(s_k, a) - I(s'_k, a))$ for some $0 \leq c$. If $c = 0$ then $GI(s'_k, a) \leq GI(s_k, a)$ follows. Now assume that $0 < c$. If $GI(s_k, a) < GI(s'_k, a)$ then we have $I(s_k, a) \leq I(s'_k, a)$ by lemma 152, which leads to $0 < 0$. So $GI(s'_k, a) \leq GI(s_k, a)$. \square

The following lemma shows that, for g a sequential tree game, a Nash equilibrium for $Gsg(g)$ is a subgame perfect equilibrium for g .

Lemma 157 *Let s be a strategy profile for the sequential game g .*

$$Eq_{Gsg(g)}(s) \Rightarrow SPE(s)$$

Proof By induction on s the Nash equilibrium for $Gsg(g)$. Note that the claim holds for leaf profiles. Now let $s = sN(b, s_1 \dots s_{k-1}, s_k, s_{k+1} \dots, s_n)$ be a compound profile for the game $g = gN(b, g_1, \dots, g_n)$. Assume that the claim holds for subprofiles of s . By lemma 156, every s_i is a Nash equilibrium for $Gsg(g_i)$. By induction hypothesis, s_i is a subgame perfect equilibrium. Moreover, s is a Nash equilibrium by lemma 155, so s is a subgame perfect equilibrium. \square

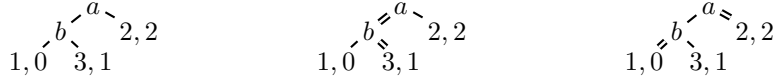
The theorem below states that the subgame perfect equilibria of a sequential tree game g are exactly the Nash equilibria of the strategic game $Gsg(g)$. The result follows lemmas 154 and 157.

Theorem 158 *Let s be a strategy profile for the sequential game g .*

$$Eq_{Gsg(g)}(s) \Leftrightarrow SPE(s)$$

This theorem suggests the following interpretation in sequential tree games. Compared to Nash equilibria, subgame perfect equilibria may not result from more rational agents with higher-level conscience, but only from agents playing the game in a different way: for instance when the starting point of the play may not be the root of the tree but an arbitrary node chosen non-deterministically.

The remainder of the subsection describes an instance of the embedding Gsg . Consider the following traditional sequential game with its two Nash equilibria, the first one being a subgame perfect equilibrium.



The traditional embedding into strategic games yields the following strategic game. Its two Nash equilibria are (a_{left}, b_{right}) and (a_{right}, b_{left}) , and they correspond to the two Nash equilibria above.

	b_{left}	b_{right}
a_{left}	1 0	3 1
a_{right}	2 2	2 2

Now compute GI for all the profiles of the above sequential game.

$$\begin{array}{ll}
 GI((1,0), a) &= 2^1 & GI((1,0), b) &= 2^0 \\
 GI((3,1), a) &= 8 & GI((3,1), b) &= 2 \\
 GI((2,2), a) &= 4 & GI((3,1), b) &= 4
 \end{array}$$

$$\begin{array}{ll}
 GI(\text{tree with } (a,b) \text{ dashed}, a) &= 2 + 8 & GI(\text{tree with } (a,b) \text{ dashed}, b) &= 1 + 2 + \frac{2 \times 2}{1} \times 1 \\
 GI(\text{tree with } (b,3,1) \text{ double}, a) &= 2 + 8 & GI(\text{tree with } (b,3,1) \text{ double}, b) &= 1 + 2 + \frac{2 \times 2}{1} \times 2
 \end{array}$$

$$\begin{aligned}
GI(\text{graph}_1, a) &= 10 + 4 + \frac{2 \times 4}{2} \times 2 \\
GI(\text{graph}_2, b) &= 7 + 4 + \frac{2 \times 2}{1} \times 1 \\
GI(\text{graph}_3, a) &= 10 + 4 + \frac{2 \times 8}{4} \times 8 \\
GI(\text{graph}_4, b) &= 11 + 4 + \frac{2 \times 2}{1} \times 2 \\
GI(\text{graph}_5, a) &= 10 + 4 + \frac{2 \times 4}{2} \times 4 \\
GI(\text{graph}_6, b) &= 7 + 4 \\
GI(\text{graph}_7, a) &= 10 + 4 + \frac{2 \times 8}{4} \times 4 \\
GI(\text{graph}_8, b) &= 11 + 4
\end{aligned}$$

The newly defined embedding into strategic games yields the following strategic game. Its only Nash equilibrium is (a_{left}, b_{right}) , which is also the only subgame perfect equilibrium of the corresponding sequential game.

	b_{left}	b_{right}
a_{left}	22 15	46 23
a_{right}	30 11	30 15

7.4 Global Equilibrium for a Subclass of Sequential Graph Games

This section presents results on equilibrium existence and non-existence for a few classes of sequential graph games. Subsection 7.4.1 lists preliminary definitions and lemmas. Subsection 7.4.2 concerns sequential graph games with the following property: along any cycle of the game at most one agent has to choose between two different arcs. The subsection gives a necessary condition and a sufficient condition on the preferences of the agents for equilibrium existence. These conditions coincide when the preferences are total orders. Subsection 7.4.3 quickly discusses the class of all sequential graph games in terms of equilibrium existence.

7.4.1 Preliminary Results

If a preference is a subrelation of a bigger preference, and if a given strategy is an local/global equilibrium with respect to the bigger preference, then the

strategy is also an local/global equilibrium with respect to the smaller preference. This is formally stated below.

Lemma 159 (Equilibrium for subpreference) *Let $\xrightarrow{\circ}$ and $\xrightarrow{\circ'}_$ be two preferences. Equilibrium preservation by subrelation is stated as follows.*

$$\begin{aligned}\xrightarrow{\circ} \subseteq \xrightarrow{\circ'}_ &\Rightarrow LEq_{\xrightarrow{\circ}_}(s, n) \Rightarrow LEq_{\xrightarrow{\circ'}_}(s, n) \\ \xrightarrow{\circ} \subseteq \xrightarrow{\circ'}_ &\Rightarrow GEq_{\xrightarrow{\circ}_}(s, n) \Rightarrow GEq_{\xrightarrow{\circ'}_}(s, n)\end{aligned}$$

Proof Note that $\xrightarrow{\circ} \subseteq \xrightarrow{\circ'}_ \Rightarrow Happy_{\xrightarrow{\circ}_}(a, s, n) \Rightarrow Happy_{\xrightarrow{\circ'}_}(a, s, n)$ \square

Consider two preferences that coincide on a subdomain, *i.e.* a subset of the infinite (ultimately periodic) sequences of outcomes. Consider a sequential graph game that involves only sequences from this subdomain. In this case, any equilibrium for this game with respect to one preference is also an equilibrium for this game with respect to the other preference. This is formally stated below.

Lemma 160 *Let g be a sequential graph game involving sequences in S only, and let s be a strategy profile for g .*

$$\begin{array}{c}\xrightarrow{\circ} \mid_S = \xrightarrow{\circ'}_ \mid_S \\ \Downarrow \\ LEq_{\xrightarrow{\circ}_}(s) \Leftrightarrow LEq_{\xrightarrow{\circ'}_}(s) \quad \wedge \quad GEq_{\xrightarrow{\circ}_}(s) \Leftrightarrow GEq_{\xrightarrow{\circ'}_}(s)\end{array}$$

A node of a sequential graph game is said to be an actual choice if at least two arcs are starting from that node.

Definition 161 (Actual choice) *Let $g = \langle E, vl, el \rangle$ be a sequential graph game.*

$$AC(n) \triangleq \exists n', n'' \in V, n' \neq n'' \wedge (n, n'), (n, n'') \in E$$

Consider a sequential graph game. The actual decision makers of a given set of nodes are the agents owning the actual choices that are in the given set of nodes.

Definition 162 (Actual decision maker) *Let $g = \langle E, vl, el \rangle$ be a sequential graph game and let V' be a subset of V .*

$$ADM(V') \triangleq \{vl(n) \mid n \in V' \wedge AC(n)\}$$

The following lemma states that if preferences are irreflexive, a sequential graph game with no actual decision maker has an equilibrium.

Lemma 163 *Let $g = \langle E, vl, el \rangle$ be a sequential graph game.*

$$(\forall \alpha, \forall a, \neg(\alpha \xrightarrow{a} \alpha)) \Rightarrow AMD(V) = \emptyset \Rightarrow \exists s = (g, c), GEq_{\xrightarrow{\circ}_}(s)$$

Proof In such a case, every node has one and only one outgoing node, so only one strategy profile corresponds to that game. This unique strategy profile is an equilibrium by irreflexivity. \square

The chapter 6 rephrases the definition of strict weak order, which is also given below.

Definition 164 (Strict weak order) *A strict weak order is an asymmetric relation whose negation is transitive, i.e. $\alpha \not\prec \beta \wedge \beta \not\prec \gamma \Rightarrow \alpha \not\prec \gamma$.*

The same paper also defines the notions of E-prefix and subcontinuity. These notions are also rephrased below.

Definition 165 (E-prefix) *A binary relation \prec over sequences is said E-prefix when complying with the following formula.*

$$u\alpha \prec u\beta \Rightarrow \alpha \prec \beta$$

Definition 166 (subcontinuity) *A relation over sequences is said subcontinuous when complying with the following formula, where u is any non-empty finite sequence.*

$$\neg(u^\omega \prec \alpha \prec u\alpha)$$

In chapter 6, the notion of dalograph refers to arc-labelled digraph with non-zero outdegree at every node. The main result of that chapter is the following one.

Theorem 167

The preference \prec is included in some \prec' .
The preference \prec' is an E-prefix and subcontinuous strict weak order.
 \Updownarrow
The preference \prec is included in some \prec' .
The preference \prec' is E-prefix, subcontinuous, transitive, and irreflexive.
The non comparability relation \sharp' is transitive.
 \Downarrow
Every dalograph has a \prec -equilibrium.
 \Downarrow
The preference \prec is included in some \prec' .
The preference \prec' is (gen-) E-prefix, (A-) transitive, and irreflexive.

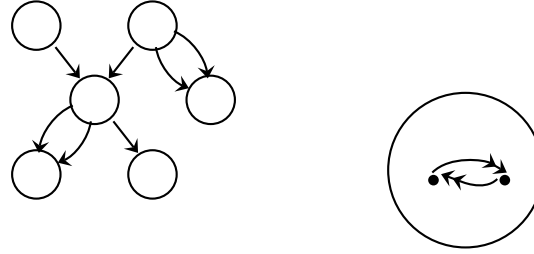
The next lemma invokes the result above. Let g be a sequential graph game with one and only one actual decision maker. Assume that the actual decision maker's preference is a subcontinuous E-prefix strict weak order. In this case, there exists an equilibrium for the game.

Lemma 168 *Let $g = \langle E, vl, el \rangle$ be a sequential graph game.*

$$(\exists a, \forall n \in V, ADM(\{n\}) = a) \Rightarrow \exists s = (g, c), GEq_{\underline{\circ}}(s)$$

Proof The agents who are not the decision maker are locally happy at every node whatever the strategy profile. Transform the sequential graph game by removing all the agents enclosed in the nodes. This yields a dalograph. This dalograph has an equilibrium with respect to the preference of the actual decision maker, by theorem 167. Transform this dalograph strategy profile into a graph-game strategy profile by adding back all the agents that were removed from the nodes. \square

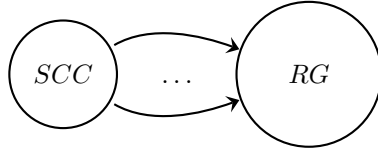
Any binary relation \rightarrow can be decomposed like the left-hand picture below. Each circle represents a strongly connected component, *i.e.* a largest set of nodes that are connected to each other. Such a circle is detailed on the right-hand side, where a double-head arrow represents the transitive closure of the relation \rightarrow .



The traditional decomposition above suggests the following decomposition.

Lemma 169 *A finite directed graph with at least two strongly connected components can be decomposed in a strongly connected component SCC reachable only from itself, a “remainder graph” RG , and possibly some arcs directed from SCC to RG .*

The decomposition of the lemma above is depicted below.



The following lemma says that a sequential graph game such that at most one agent is an actual decision maker along any cycle has at most one actual decision maker per strongly connected component. The notation xEy stands for $(x, y) \in E$.

Lemma 170 *Let $g = \langle E, vl, el \rangle$ be a sequential graph game and let SCC be a strongly connected component of E .*

$$\begin{aligned} \forall n_1, \dots, n_k \in V, n_1 E n_2 E \dots E n_k E n_1 &\Rightarrow |ADM(\{n_1, \dots, n_k\})| = 1 \\ &\Downarrow \\ |ADM(SCC)| &= 1 \end{aligned}$$

Proof Consider a node n owned by agent a and a node n' owned by agent a' , in the same strongly connected component. Assume that agent a (resp. a') is an actual decision maker at n (resp. n'). By strong connectedness, there is a path from n to n' , and a path from n' to n , both paths are along E . This defines a cycle with respect to E , so a equals a' by uniqueness assumption. \square

7.4.2 Sufficient and Necessary Conditions

Assume that an agent's preference is not included in any irreflexive, gen-E-prefix, and A-transitive binary relation. In this case, one can no longer guarantee equilibrium for every sequential graph game, as stated below.

Theorem 171 *If the combination closure of an agent's preference is not irreflexive, then there exists a sequential graph game that involves only this agent and that has no global equilibrium. Let $\xrightarrow{\circ}$ be the preference relations.*

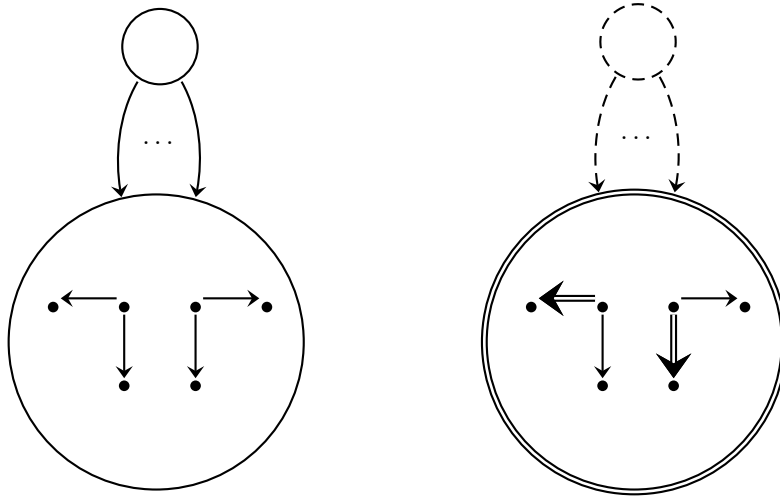
$$(\exists \alpha, \exists a, \alpha \xrightarrow{\circ_a}_c \alpha) \Rightarrow \exists g, \forall c, \neg GEq(g, c)$$

Proof By theorem 167. □

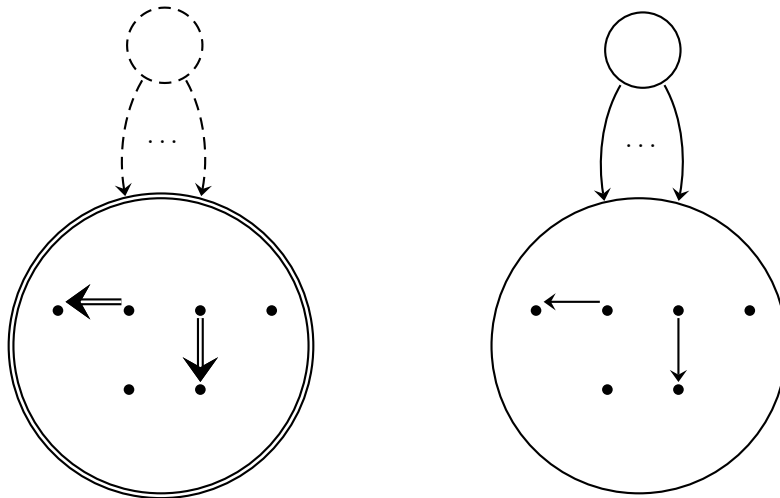
In the following, games are represented by big circles, strategy profiles by big double-lined circles, nodes by small circles, possible moves by arrows, chosen moves by double-lined arrows, and a (possibly empty) sequence of possible/chosen moves by a double-headed arrows.

Theorem 172 *Consider a sequential graph game such that at most one agent is an actual decision maker along any cycle. Assume that the preferences of the agents are all included in some E-prefix and subcontinuous strict weak orders. In this case, there exists a global equilibrium for the sequential graph game.*

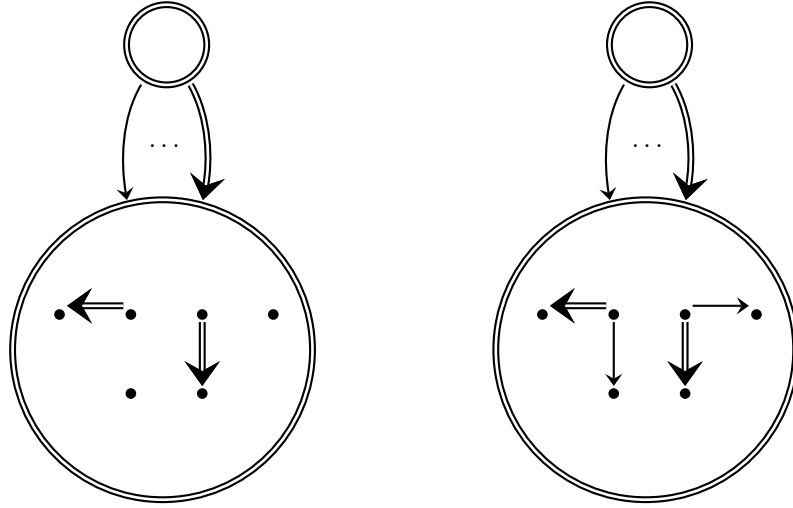
Proof By lemma 159, it suffices to prove the claim when the preferences of the agents are all actual E-prefix and subcontinuous strict weak orders. Proceed by induction on the number of arcs, starting from 1. If a game has exactly one arc, then it has no actual decision maker, so lemma 163 allows concluding. Let n be a non-zero natural number, and assume that any sequential graph game with n arcs or less such that at most one agent is an actual decision maker along any cycle in the graph has an equilibrium. Consider G a game with $n + 1$ node, and such that at most one agent is an actual decision maker along any cycle in the graph. If the game has only one connected component then it has at most one actual decision maker by lemma 170, so it has an equilibrium by lemma 168. If the game has two or more strongly connected components, then it can be decomposed in a non-empty strongly connected component SCC reachable only from itself, a non-empty "remainder graph" RG , and maybe some arcs directed from SCC to RG , according to lemma 169. If there is at most one actual decision maker then we are done by lemma 168. If there are two or more actual decision makers, then there is at least one in RG , because there can be only one in SCC by lemma 170. Some nodes in RG with actual decision makers are represented on the left-hand picture below. The strongly connected component SCC involves at least one node by construction. Since any node has at least one outgoing arc, the game RG has n or less arcs. As a subgame of G , RG is also such that at most one agent is an actual decision maker along any cycle. By induction hypothesis, RG has an equilibrium RS , represented on the lower part of right-hand picture below.



In the strategy profile RS , at least one arc is not chosen, since at each node one and only one arc is chosen. Removing all the non-chosen arcs defines a strategy profile RS' , as shown on the left-hand picture below. The underlying game is named RG' . Putting SCC and RG' back together yields a game G' with n or less arcs, as shown on the left-hand picture below.



As a subgame of G , G' is such that at most one agent is an actual decision maker along any cycle. By induction hypothesis, there is an equilibrium S' for the game G' , as shown on the left-hand picture below. Adding to S' the arcs that were once removed defines a strategy profile for G , as shown on the right-hand picture below.



The remainder proves that S is an equilibrium. It suffices to show that, starting from a given node, the play induced by S makes all agents happy. Note that the substrategy profile of S restricted to RG equals RS . For a node in RG , the current play is also a play induced by RS , and if an agent can convert the current play to another play, then the agent can also do it with respect to RS . Since RS is an equilibrium, the play cannot be improved upon. Now consider a node in SCC , and the plays starting from this node. Consider an agent that can convert the current play to another play. If this other play does not involve any of the arcs that were added back, then this play, which is also a play in the equilibrium S' , does not improve upon the current play. If the other play involves an arc that was added back, let us consider the first such an arc along the play. The corresponding outcome is named x .

$$\begin{array}{c} \vdash \alpha \text{ --- } u \gg \text{ --- } x \rightarrow \text{ --- } \beta \vdash \\ \quad \quad \quad \downarrow \\ \quad \quad \quad \gamma \\ \quad \quad \quad \vdash \end{array}$$

Since α and $u\gamma$ correspond to plays in the equilibrium S' that are convertible to each other by agent a , we have $u\gamma \leq_a \alpha$. Since γ and $x\beta$ correspond to plays in the equilibrium RS that are convertible to each other by agent a , we have $x\beta \leq_a \gamma$. If $x\beta = \gamma$ then $ux\beta = u\gamma$. If $x\beta <_a \gamma$ then $\neg(u\gamma <_a ux\beta)$ by strong acyclicity, so $ux\beta \leq_a u\gamma$ by total ordering. In both cases, $ux\beta \leq_a u\gamma$, so by transitivity $ux\beta \leq_a \alpha$. \square

The following theorem follows the fact that when considering only total orders, the two conditions of lemmas 171 and 172 coincide, as explained in chapter 6.

Theorem 173 *If the preferences of given agents are all total orders then they are E-prefix and subcontinuous iff every sequential graph game with at most one decision maker along every cycle has a global equilibrium.*

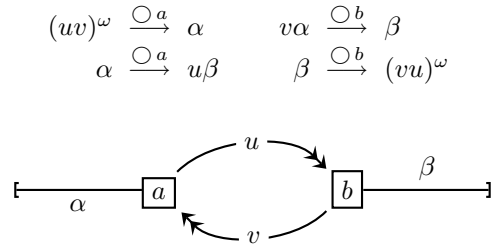
The abstract generalisation [29] of Kuhn's theorem follows the above results.

Corollary 174 *An abstract sequential tree game with acyclic preferences always has a Nash/subgame perfect equilibrium.*

Proof Transform the tree game by the embedding of subsection 7.3.1. Prove that the preferences $\xrightarrow[t2g]{\circ^a}$ are E-prefix. Assume that $u\alpha \xrightarrow[t2g]{\circ^a} u\beta$. So $u\alpha = doc^n oc_1$ and $u\beta = doc^p oc_2$ for some n, p, oc_1 and oc_2 such that $oc_1 \xrightarrow{\circ^a} oc_2$. Since $\xrightarrow{\circ^a}$ is acyclic, $oc_1 \neq oc_2$. So $u = doc^k$ for some k less than or equal to both n and p . So $\alpha = doc^{n-k} oc_1$ and $\beta = doc^{p-k} oc_2$, so $\alpha \xrightarrow[t2g]{\circ^a} \beta$ since $oc_1 \xrightarrow{\circ^a} oc_2$. Now prove that the preferences $\xrightarrow[t2g]{\circ^a}$ are subcontinuous. Assume that $\alpha \xrightarrow[t2g]{\circ^a} u\alpha$. So $\alpha = doc^n oc_1$ and $u\alpha = doc^p oc_2$ for some n, p, oc_1 and oc_2 such that $oc_1 \xrightarrow{\circ^a} oc_2$. So, $oc_1 = oc_2$ which is contradictory since $\xrightarrow{\circ^a}$ is acyclic. By theorem 172 the transformed game has a global equilibrium. Conclude by lemma 148. \square

7.4.3 Further Equilibrium (Non-) Existence

The sufficient condition for equilibrium existence proved in subsection 7.4.2 works for a special subclass of sequential graph games, but it does not work in general, as shown by the example below.



However, the following lemma gives a sufficient condition for equilibrium existence in all sequential graph games. The condition consists in a predicate on the union of the agents' preferences instead of a conjunction of predicates on the agents' preferences taken separately. This condition sounds very strong, but the above example rules out (non-trivial) conditions made of predicates on the agents' preferences taken separately.

Lemma 175 *If the union of the preferences of all agents is an E-prefix and subcontinuous strict weak order, then every sequential graph game has a global equilibrium.*

Proof By lemma 159, it suffices to show that every sequential game has a global equilibrium when replacing every agent's preference by the union of the agents' preferences. Replacing the agents owning the nodes with one same

agent for all nodes yields a convertibility binary relation that includes the original convertibility relation of each agent. Therefore an equilibrium for this new game is also an equilibrium for the original game. By lemma 168, this new game has a global equilibrium, which is also an global equilibrium for the original game. \square

7.5 Conclusion

This chapter defines sequential graph games, and the related notions of strategy profile, convertibility, and local/global preference. Based on this, it defines the notions of local and global equilibrium. They correspond to the notions of Nash and subgame perfect equilibrium respectively, as shown by an embedding. They also both correspond to the abstract Nash equilibria of two different derived C/P games.

Putting the two previous remarks together suggests that the subgame perfect equilibria of a sequential game are the Nash equilibria of a derived C/P game. Working in an abstract setting helps think of this idea, but the notions of sequential graph games and C/P games are not actually needed to express this idea. Indeed, this chapter defines an embedding from sequential games with real-valued payoff functions into strategic games with real-valued payoff functions. The images by the embedding of the subgame perfect equilibria (of the sequential game) are exactly the Nash equilibria (of the strategic game). This result shows that, in some sense, subgame perfect equilibria could be called Nash equilibria of sequential games in lieu of their current Nash equilibria.

The main goal of this chapter is achieved as follows: the chapter considers a subclass of sequential graph games and proves a sufficient condition and a necessary condition for equilibrium existence in this subclass. It does it by invoking results on equilibrium existence in dalographs. The proof of the sufficient condition uses a construction similar to "backward induction". This way, it tries to fully exploit the conceptual power of "backward induction" and seeks for a very general setting where "backward induction" still helps build equilibria. The necessary condition and the sufficient condition coincide when the agents' preferences are total orders, which gives an equivalence property. These results supersede the first generalisation [29] of Kuhn's theorem. This chapter also gives a sufficient condition on the union of the agents' preferences for equilibrium existence in all sequential graph games. However, this condition is strong and it is likely not to hold in many interesting examples and applications. In the same line of work, it may be interesting to study the subclass of sequential graph games where each agent occurs at most once, *i.e.* owns at most one node.

This chapter also leads to the following thoughts. In abstract sequential games, every game involving only one agent has a Nash/subgame perfect equilibrium *iff* every game has a Nash/subgame perfect equilibrium, as shown in [29]. (This does not hold in strategic games, as shown by some basic examples.) This chapter gives a sufficient condition for equilibrium existence for every sequential graph game involving only one agent. It also gives a sufficient condition for equilibrium existence for every game of a subclass of sequential graph games. Actually, this subclass includes games involving only one agent and these two sufficient conditions are the same (in this chapter). However, it

is not obvious whether or not equilibrium existence in games involving only one agent implies equilibrium existence in the subclass mentioned above. So, it may be interesting to seek a "most general" subclass of sequential graph games where the following holds: "If every game involving only one agent has an equilibrium, then every game has an equilibrium". It may be interesting especially because the property "every game involving only one agent has an equilibrium" could be a low-level definition of rationality, *i.e.* not involving further theories such as epistemic logic.

Chapter 8

Abstract Compromised Equilibria

8.1 Introduction

8.1.1 Nash's Theorem

As seen before, not all finite strategic games have a (pure) Nash equilibrium, even though a guarantee of existence is a desirable property. Therefore, Nash weakened the definition of (pure Nash) equilibrium to assure the existence of a "compromising" equilibrium. More specifically, introducing probabilities into the game, he allowed agents to choose their individual strategies with a probability distribution rather than choosing a single strategy deterministically. Subsequently, instead of a single strategy profile chosen deterministically, Nash's probabilistic compromise involves a probability distribution over strategy profiles. So, expected payoff functions are involved instead of payoff functions. This compromise actually builds a new strategic game that is continuous. Nash [36] proved that this new game always has a Nash equilibrium, which is called a probabilistic Nash equilibrium for the original game. A first proof of this result invokes Kakutani's fixed point theorem, and a second proof invokes (the proof-theoretically simpler) Brouwer's fixed point theorem.

8.1.2 Contribution

First, this chapter explores possible generalisations of Nash's result within the BR and CP formalisms. These generalisations invoke Kakutani's fixed point theorem, which is a generalisation of Brouwer's. However, there is a difference between Nash's theorem and its generalisations in the BR and CP formalisms: Nash's probabilised strategic games correspond to finite strategic games since they are derived from them. However, it seems difficult to introduce probabilities within a given finite BR or CP game, since BR and CP games do not have any Cartesian product structure. Therefore, one considers a class of already-continuous BR or CP games. Kakutani's fixed point theorem, which is much more appropriate than Brouwer's in this specific case, helps guarantee the existence of an abstract Nash equilibrium. However, these already-continuous BR

or CP games do not necessarily correspond to relevant finite BR or CP games, so practical applications might be difficult to find.

Second, this chapter explores compromises that are completely different from Nash's probabilistic compromise. Two conceptually very simple compromises are presented in this section, one for BR games and one for CP games. The one for BR games is named or-best response strict equilibrium, and the one for CP games is named change-of-mind equilibrium, as in [30]. Both compromising equilibria are natural generalisations of Nash equilibria for CP games and strict Nash equilibria for BR games. It turns out that both generalisations are (almost) the same: they both define compromising equilibria as the sink strongly connected components of a relevant digraph. Informally, if nodes represent the microscopic level and strongly connected components the macroscopic level, then the compromising equilibria are the macroscopic equilibria of a microscopic world.

Since BR and CP games are generalisations of strategic games, the new compromising equilibria are relevant in strategic games too. This helps see that Nash's compromise and the new compromises are different in many respects. Nash's probabilistic compromise transforms finite strategic games into continuous strategic games. Probabilistic Nash equilibria for the original game are defined as the (pure) Nash equilibria for the continuous derived game. This makes the probabilistic setting much more complex than the "pure" setting. On the contrary, the new compromises are discrete in the sense that the compromising equilibria are finitely many among finitely many strongly connected components. While probabilistic Nash equilibria are static, the new compromising equilibria are dynamic. While Nash equilibria are fixed points obtained by Kakutani's (or Brouwer's) fixed point theorem, the new equilibria are fixed points obtained by a simple combinatorial argument (or Tarski's fixed point theorem if one wishes to stress the parallel between Nash's construction and this one). While probabilistic Nash equilibria are non-computable in general, the new compromising equilibria are computable with low algorithmic complexity. Finally, while probabilistic Nash equilibria are Nash equilibria of a derived continuous game, the new compromising equilibria seem not to be the Nash equilibria of any relevant derived game.

8.1.3 Contents

Section 8.2 defines probabilistic Nash equilibria and states the related Nash's theorem; section 8.3 tries to generalise Nash's result within the BR or CP formalisms; section 8.4 introduces discrete and dynamic new compromising equilibria and proves a few related results.

8.2 Probabilistic Nash Equilibrium

As noted before, not all strategic games have (pure) Nash equilibria. The left-hand example is finite and involves two agents. The right-hand example is infinite and involves only one agent. None of them has a Nash equilibrium.

	h_1	h_2								
v_1	1	0	0	1						
v_2	0	1	1	0						

h_0	h_1	h_2	\dots	h_i	\dots
0	1	2	\dots	i	\dots

Facing this "problem", Nash considered probabilised strategies and strategy profiles. The former refers to probability distributions over the strategies of a given agent. The latter refer to combinations of those, one probability distribution per agent. They are defined below.

Definition 176 Let $\langle \mathcal{A}, \mathcal{S}, P \rangle$ be a finite strategic game.

$$W_a \triangleq \{ \omega : \mathcal{S}_a \rightarrow [0, 1] \mid \sum_{\sigma \in \mathcal{S}_a} \omega(\sigma) = 1 \}$$

$$W \triangleq \bigotimes_{a \in \mathcal{A}} W_a$$

As defined above, probability distributions can be assigned to the finite set of strategies of each agent. This induces a probability distribution over the finite set of strategy profiles. Together with the original payoff functions of a strategic game, this induces an expected-payoff function. These two induced notions are defined as follows.

Definition 177 (Expected Payoff) Let $\langle \mathcal{A}, \mathcal{S}, P \rangle$ be a finite strategic game, and let $w \in W$, $s \in \mathcal{S}$, and $a \in \mathcal{A}$.

$$\mu(w, s) \triangleq \prod_{\alpha \in \mathcal{A}} w_\alpha(s_\alpha)$$

$$E_P(w, a) \triangleq \sum_{\sigma \in \mathcal{S}} \mu(w, \sigma) \cdot P(\sigma, a)$$

The construction above, Nash's construction, is actually a function from finite strategic games to "continuous" strategic games, as stated below.

Proposition 178 (Probabilistic Strategic Games) For $g = \langle \mathcal{A}, \mathcal{S}, P \rangle$ a finite strategic game, $W_g \triangleq \langle \mathcal{A}, W, E_P \rangle$ is a strategic game.

Nash's result reads as follows.

Theorem 179 (Nash [36]) For any finite strategic game g there exists an element of W that is a Nash equilibrium for W_g .

For the left-hand strategic game below, its (only) probabilistic Nash equilibrium is the probability assignment $v_i \mapsto \frac{1}{2}$, $h_i \mapsto \frac{1}{2}$. However, Nash's construction may not work for infinite strategic games. For instance, the right-hand strategic games below has no probabilistic Nash equilibrium. Indeed for any distribution, shifting the distribution to the right by 1 increases the expected payoff by 1.

8.3.1 Continuous BR Nash Equilibrium

A direct application of Kakutani's fixed point theorem.

Lemma 183 *Let $g = \langle \mathcal{A}, \mathcal{S}, (BR_a)_{a \in \mathcal{A}} \rangle$ be a BR game. Assume that \mathcal{S} is a compact and convex subset of \mathcal{R}^n . Assume that the global best response function complies with the following conditions.*

1. $BR^\cap(s)$ is non-empty for all s in \mathcal{S} ,
2. $BR^\cap(s)$ is convex for all s in \mathcal{S} ,
3. BR^\cap is upper semi-continuous.

Then $Eq_g(s)$ for some s in \mathcal{S} .

However, this result does not sound very practical since Nash's theorem is not a corollary of it. Indeed, the BR^\cap function of the embedding of a strategic game into BR game may be empty for some s in \mathcal{S} . For instance in the game below, the BR^\cap function is the constant function returning the empty set.

	h_1	h_2
v_1	1 0	0 1
v_2	0 1	1 0

8.3.2 Continuous CP Nash Equilibrium

CP games can be seen as binary relations when considering the change-of-mind, so a binary-relation-oriented version of Kakutani's theorem is given below.

Lemma 184 (Kakutani for continuous binary relations) *Let C be a compact and convex subset of \mathcal{R}^n . Let \longrightarrow be a binary relation over C such that:*

1. *Any element in C has a successor with respect to \longrightarrow .*
2. *The successors, with respect to \longrightarrow , of any element form a convex set.*
3. *The graph of \longrightarrow is closed.*

Then \longrightarrow is not irreflexive.

Contraposition between the first condition and the conclusion yields the following generalisation of Nash's theorem in CP games.

Lemma 185 (Kakutani for continuous CP games) *Consider a CP game g written $\langle \mathcal{A}, \mathcal{S}, (\overset{+}{\longrightarrow}_a)_{a \in \mathcal{A}}, (\overset{\circ}{\longrightarrow}_a)_{a \in \mathcal{A}} \rangle$. Let \longrightarrow be its change-of-mind relation. Assume that the CP game complies with the following conditions.*

1. *\mathcal{S} is a compact and convex subset of \mathcal{R}^n .*
2. *The change-of-mind relation \longrightarrow is irreflexive.*
3. *The successors, with respect to \longrightarrow , of any element form a convex set.*

4. The graph of \longrightarrow is closed.

Then $Eq_g(s)$ for some s in S .

It is unclear how practical this generalisation is. Indeed, while probabilised strategic games correspond to finite strategic games, it may be difficult to see what kind of discrete CP games those continuous CP games correspond to.

8.4 Discrete and Dynamic Equilibria

This section first defines the strongly connected components (SCC) of a digraph, and a few related properties. Second, it defines change-of-mind equilibrium as an SCC-sink of a relevant digraph. Third, it defines or-best response strict equilibrium as an SCC-sink of a relevant digraph. Fourth, it gives a few examples and compares briefly the three compromising equilibria (probabilistic, change-of-mind, or-best response strict) in the strategic game setting.

8.4.1 Strongly Connected Component

This subsection defines the transitive and reflexive closure of a binary relation, using the usual inductive definition. It gives some properties of this closure, which it uses to define the strongly connected components of a digraph, and the notion of (SCC-) sink. Then, it gives a few equivalent characterisations for SCC-sinks, in general as well as for finite digraphs without sink. This subsection also shows in two different manners the existence of SCC-sinks for finite digraphs. Most of the results discussed in this subsection must already belong to the folklore of graph theory.

Preliminary

Formally, the transitive and reflexive closure of a binary relation is defined as follows.

Definition 186 (Transitive and reflexive closure)

$$\frac{}{s \rightarrow^* s} \quad \frac{s \rightarrow s'}{s \rightarrow^* s'} \quad \frac{s \rightarrow^* s' \quad s' \rightarrow^* s''}{s \rightarrow^* s''}$$

The following lemma says that if several steps (from a given starting point) allow to escape from a set, then one step (from a well chosen starting point) also does.

Lemma 187 *Let \rightarrow be a digraph and let C be a subset of its nodes. If $s \rightarrow^* s'$ for some s in C and s' not in C , then $s \rightarrow^* s_1 \rightarrow s'_1 \rightarrow^* s'$ for some s_1 in C and s'_1 not in C .*

Proof By rule induction on the definition of the transitive and reflexive closure (of \rightarrow). \square

The next lemma says that if several steps (from a given starting point) lead to an end point, then one step (from a well chosen starting point) also does.

Lemma 188 *Let \rightarrow be a digraph and let s and s' be two different nodes such that $s \rightarrow^* s'$. There exists s'' such that $s \rightarrow^* s'' \rightarrow s'$.*

Proof Apply lemma 187 with C defined as the nodes different from s' . \square

Definition and Fast Computation

Any digraph induces equivalence classes named strongly connected components. These components are the nodes of a derived digraph that is actually a dag, i.e. a directed acyclic graph.

Definition 189 (SCC, Shrunk Graph) Consider $\rightarrow \subseteq \mathcal{S} \times \mathcal{S}$.

- The strongly connected component (SCC) of $s \in \mathcal{S}$ is

$$[s] \triangleq \{s' \mid s \rightarrow^* s' \wedge s' \rightarrow^* s\}$$

- The set of SCCs of a graph is $[\mathcal{S}] \triangleq \{[s] \mid s \in \mathcal{S}\}$.
- The shrunk graph of $\rightarrow \subseteq \mathcal{S} \times \mathcal{S}$ is a graph \curvearrowright over the SCCs of \rightarrow and it is defined as follows.

$$C \curvearrowright C' \triangleq \exists s \in C, \exists s' \in C', s \rightarrow s'$$

Building the SCCs has a very low algorithmic complexity, as stated below.

Theorem 190 (Tarjan) The shrunk graph of a finite directed graph can be found in linear time in the sizes of \mathcal{S} and \rightarrow , i.e. with algorithmic complexity $|\mathcal{S}|^2$.

A basic concept in graph theory, a sink is a node of a digraph from which one cannot escape (when moving along the arcs of the digraph). This concept extends to SCCs.

Definition 191 (SCC-Sink) Given \rightarrow a digraph with nodes \mathcal{S} , a node s is a sink if $s \not\rightarrow$, i.e. if s has no outgoing arc with respect to \rightarrow . In the same way, a SCC, named C , of \rightarrow is a SCC-sink if $C \not\curvearrowright$.

Characterisations of SCC-Sinks

The following function defines the nodes of a digraph that are reachable from given nodes by moving along the arcs of the digraph, i.e. by transitive and reflexive closure.

Definition 192 Let \rightarrow be a digraph over \mathcal{S} and let $C \subseteq \mathcal{S}$.

$$\mathcal{U}(C) \triangleq \bigcup_{s \in C} \{s' \mid s \rightarrow^* s'\}$$

The following lemma gives three alternative characterisations of SCC-sink.

Lemma 193 Let \rightarrow be a digraph over \mathcal{S} . The following propositions are equivalent

1. C is a SCC-sink.
2. $C \neq \emptyset$ and $\forall s \in C, \forall s' \in \mathcal{S} \quad s \rightarrow^* s' \Leftrightarrow s' \in C$.
3. $C \neq \emptyset$ and $\forall s \in C, \mathcal{U}(\{s\}) = C$.

4. C is a least (for set inclusion) non-empty fixed-point of \mathcal{U} .

Proof 1) \Rightarrow 2): C is non-empty since it is an SCC. Let s be in C and s' be in \mathcal{S} . If s' is in C then $s \rightarrow^* s'$ by definition of SCC. Conversely, assume $s \rightarrow^* s'$. If s' is not in C then there exist s_1 in C and s'_1 not in C such that $s_1 \rightarrow s'_1$, according to lemma 187. So $C \curvearrowright [s'_1]$ by definition of \curvearrowright , which contradicts C being a SCC-sink. Therefore s' is in C .

2) \Rightarrow 3): by assumption and by definition of \mathcal{U} .

3) \Rightarrow 4): C is non-empty by assumption. It is a fixed point by assumption and by definition of \mathcal{U} . By definition of \mathcal{U} and by assumption, $\emptyset \neq C' \subsetneq C$ implies $\mathcal{U}(C') = C$. This shows that C is a least non-empty fixed point for \mathcal{U} .

4) \Rightarrow 1): first show that C is an SCC. Let s be in C . By definition of \mathcal{U} , $\mathcal{U}(\{s\}) \subseteq \mathcal{U}(C)$. Also, $\mathcal{U}(\mathcal{U}(\{s\})) = \mathcal{U}(\{s\})$. Therefore $\mathcal{U}(\{s\}) = \mathcal{U}(C)$, otherwise it would contradict minimality of the fixed point $\mathcal{U}(C) = C$. So $s \rightarrow^* s'$ for all s and s' in C ; C is an SCC. Now show that C is an SCC-sink. Let s be in C and s' be in \mathcal{S} such that $s \rightarrow s'$. By definition of \mathcal{U} , s' is in $\mathcal{U}(C)$ since s is in C . But $\mathcal{U}(C) = C$ by assumption, so s' is in C . Therefore by definition of \curvearrowright , there exists no SCC C' such that $C \curvearrowright C'$; C is an SCC-sink. \square

Since SCC-sinks are sinks of a derived dag, this suggests that they are inevitable. The left-to-right implication of the second characterisation of lemma 193 suggests that one cannot escape from them. The fourth characterisation suggests atomicity of SCC-sinks.

Existence of SCC-Sinks

There are at least two ways of guaranteeing existence of SCC-sink for finite digraph. First way, every finite dag has a sink. Second way, invoke the third characterisation of lemma 193 together with Tarski's fixed point theorem, as shown below.

Lemma 194 *Given a digraph, \mathcal{U} has a least non-empty fixed point.*

Proof Let \mathcal{G} be a digraph over finite \mathcal{S} . First show that \mathcal{U} has a complete lattice of fixed points. The empty set \emptyset and the full set \mathcal{S} are fixed points of \mathcal{U} . Note that \mathcal{U} is order-preserving ($C_1 \subseteq C_2 \Rightarrow \mathcal{U}(C_1) \subseteq \mathcal{U}(C_2)$) by construction, and that it is defined on $\mathcal{P}(\mathcal{S})$, which is a complete lattice when ordered by set inclusion. Tarski's Fixed Point Theorem allows to state the intermediate claim. Since \mathcal{S} is finite and fixed point for \mathcal{U} , there exists a least non-empty fixed point for \mathcal{U} . \square

SCC-Sinks For Non-Zero Outdegrees

The following function expresses all the nodes that are exactly one step away from given nodes.

Definition 195

$$\text{succ}(C) \triangleq \bigcup_{s \in C} \{s' \mid s \rightarrow s'\}$$

Fixed points of succ are fixed points of \mathcal{U} . (The converse is false.)

Lemma 196

$$\text{succ}(C) = C \Rightarrow \mathcal{U}(C) = C$$

Proof Prove by induction on the definition of the transitive and reflexive closure (of \rightarrow) that $s \in C$ and $s \rightarrow^* s'$ implies $s' \in C$. \square

For finite digraphs without sink, least non-empty fixed points are the same for succ and \mathcal{U} .

Lemma 197 *If a finite digraph \rightarrow has no sink then the following two propositions are equivalent.*

1. C is a least (for set inclusion) non-empty fixed-point of \mathcal{U} .
2. C is a least (for set inclusion) non-empty fixed-point of succ .

Proof 1) \Rightarrow 2): $\text{succ}(C) \subseteq \mathcal{U}(C)$ by definitions of succ , \mathcal{U} , and the transitive and reflexive closure. So $\text{succ}(C) \subseteq C$ since $\mathcal{U}(C) = C$ by assumption. Conversely, let s be in C . Since the digraph has no sink, there exists s' in \mathcal{S} such that $s \rightarrow s'$. If $s' = s$ then $s \in \text{succ}(\{s\}) \subseteq \text{succ}(C)$. Now assume that $s' \neq s$. Since s' is in $\text{succ}(C)$, it is also in C , as just proved. Therefore $\mathcal{U}(\{s'\}) = C$ by the third characterisation of lemma 193. More specifically, $s' \rightarrow^* s$. Since $s' \neq s$, there exists s'' such that $s' \rightarrow^* s'' \rightarrow s$, according to lemma 188. Since s'' is in $\mathcal{U}(C) = C$, the node s is in $\text{succ}(C)$.

2) \Rightarrow 1): $\mathcal{U}(C) = C$ by lemma 196. Let C' be a least non-empty fixed point of \mathcal{U} included in C (it exists by finiteness assumption of the digraph). As proved in 1) \Rightarrow 2), the set C' is also a least non-empty fixed point of succ . So $C' = C$, not to contradict the assumption. \square

8.4.2 Discrete and Dynamic Equilibrium for CP Game

In CP games, (abstract) Nash equilibria are sinks of the change-of-mind relation; change-of-mind equilibria are defined below as SCC-sinks of the change-of-mind relation.

Definition 198 *Let g be a CP game with synopses \mathcal{S} and change-of-mind relation \longrightarrow .*

$$\text{Eq}_g([s]) \triangleq \forall s' \in \mathcal{S}, [s] \not\rightarrow [s']$$

The above definition is a natural extension of (abstract) Nash equilibria in CP games for two reasons: first, both viewpoints are similar since they define equilibria as sinks. Second, an (abstract) Nash equilibrium in a CP game is also a change-of-mind equilibrium of the same CP game since a sink is an SCC-sink.

Since a finite digraph has an SCC-sink, the next result follows directly the above definition.

Lemma 199 *A finite CP game has a change-of-mind equilibrium.*

Informally, change-of-mind equilibria can be interpreted as the "generalised cycles", one of which a CP game will enter and remain in if the play complies with the following two conditions. First, agents plays in turn, *i.e.* in a sequential manner. Second, an agent may go for anything better than his current situation, irrespectively to any long-term planning.

8.4.3 Discrete and Dynamic Strict Equilibrium for BR Game

This subsection extends the or-best response function that pertains to BR games. Using this, it extends the notion of strict Nash equilibrium. It turns out that this generalised notion coincides with SCC-sinks of some well-chosen digraph.

The or-best response function, which expects a synopsis, is generalised so that it expects a set of synopses.

Definition 200 (Generalised or-best response) Let $g = \langle \mathcal{A}, \mathcal{S}, (BR_a)_{a \in \mathcal{A}} \rangle$ be a BR game.

$$BR^\cup(C) \triangleq \bigcup_{s \in C} BR^\cup(s)$$

Note that unfolding the definition of $BR^\cup(s)$ shows that $BR^\cup(C)$ equals $\bigcup_{s \in C, a \in \mathcal{A}} BR_a(s)$.

According to lemma 21, s is a strict Nash equilibrium in a BR game, written $\text{Eq}_g^+(s)$, iff $\{s\} = BR^\cup(s)$. The following definition extends to sets of synopses the concept of strict Nash equilibrium that is defined only for synopses in the first place. More specifically, the generalised equilibria, named or-best response strict equilibria, are the least (with respect to set inclusion) non-empty fixed points of the generalised or-best response function.

Definition 201 (or-best response strict equilibrium) Consider a BR game $g = \langle \mathcal{A}, \mathcal{S}, (BR_a)_{a \in \mathcal{A}} \rangle$.

$$\begin{aligned} \text{Eq}_g^+(C) \triangleq & \quad C = BR^\cup(C) \\ & \wedge \quad C \neq \emptyset \quad \wedge \quad \forall C', \emptyset \neq C' \subsetneq C \Rightarrow C' \neq BR^\cup(C') \end{aligned}$$

The above definition is a natural extension of (abstract) strict Nash equilibria in BR games for two reasons: first, both viewpoints are similar since they define strict equilibria as fixed points of a or-best response function. Second, an (abstract) strict Nash equilibrium in a BR game is also an or-best response strict equilibrium of the same BR game, by definition of BR^\cup .

The following lemma states that the or-best response strict equilibria of a BR game are the SCC-sinks of a derived digraph.

Lemma 202 Let $g = \langle \mathcal{A}, \mathcal{S}, (BR_a)_{a \in \mathcal{A}} \rangle$ be a BR game. Define \rightarrow as follows.

$$\forall s, s' \in \mathcal{S}, \quad s \rightarrow s' \triangleq s' \in BR^\cup(s)$$

The or-best response strict equilibria of g are the SCC-sinks of \rightarrow .

Proof Note that $BR^\cup(C)$ equals $\text{succ}(C)$. □

Equipped with the above lemma, it is possible to state existence of or-best response strict equilibria.

Lemma 203 Any finite BR game has an or-best response strict equilibrium.

Proof By definition of BR games, the digraph \rightarrow has no sink, therefore the result follows lemmas 193, 202, and 197. □

Informally, or-best response strict equilibria can be interpreted as the "generalised cycles", one of which a BR game will enter and remain in if the play complies with the following two conditions. First, agents plays in turn, *i.e.* in a sequential manner. Second, an agent always moves from his current situation to the instant best possible situation, irrespectively to any long-term planning.

8.4.4 Examples and Comparisons

This subsection shows with a few examples of strategic games that Nash's probabilistic equilibrium, change-of-mind equilibrium, and or-best response strict equilibrium are not related by obvious inclusion properties.

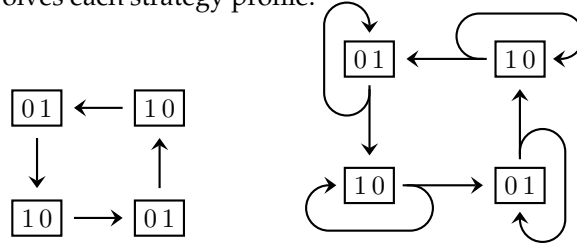
First note that while finite digraphs have least non-empty \mathcal{U} -fixed points, infinite CP games may not have change-of-mind equilibria. Recall that, in the same way, infinite strategic games may not have probabilistic Nash equilibria. This is showed by the example below.

$$\begin{array}{c|c|c|c|c|c} h_0 & h_1 & h_2 & \dots & h_i & \dots \\ \hline 0 & 1 & 2 & \dots & i & \dots \end{array} \quad 0 \rightarrow 1 \rightarrow 2 \dots \rightarrow i \rightarrow \dots$$

When all equilibria involve the same profiles Consider the following strategic game.

	h_1	h_2
v_1	0, 1	1, 0
v_2	1, 0	0, 1

The only Nash equilibrium of the above game involves each strategy profile with probability $\frac{1}{4}$. This game, when seen as a CP game, induces the change-of-mind digraph displayed on the left-hand side below. The only change-of-mind equilibrium also involves each strategy profile. This game, when seen as a BR game, induces the or-best response digraph (defined as in lemma 202) displayed on the right-hand side below. The only or-best response strict equilibrium also involves each strategy profile.



When change-of-mind and or-best response equilibria do not coincide When embedding a strategic game into a CP game or a BR game, the two derived digraphs, and their SCC-sinks, are not always the same. For instance, consider the following strategic game.

	h_1	h_2	h_3
v_1	1, 0	0, 1	0, 2
v_2	0, 2	1, 0	1, 0

The only change-of-mind equilibrium of the strategic game above (seen as a CP game) involves all strategy profiles, while the only or-best response strict equilibrium of the same strategic game (seen as a BR game) involves the four "corner" strategy profiles on the picture above. Actually for every strategic game, every or-best response strict equilibrium is included in some change-of-mind equilibrium.

In the examples of the rest of this subsection, change-of-mind equilibria coincide with or-best response strict equilibria, so they may be referred to as discrete and dynamic equilibria.

The following shows that depending on the considered strategic games, the two sets of strategy profiles involved in the probabilistic Nash equilibria and the discrete and dynamic equilibria may be ordered by inclusion either way, may be disjoint, or may overlap in a non-trivial way. Similarly, no kind of equilibrium gives better "average" payoffs every time.

Discrete and dynamic included in probabilistic The following example highlights an interesting feature of discrete and dynamic equilibria, namely the ability to carve out a part of a game as constituting an equilibrium. This is illustrated below to the right, with six involved profiles and average payoffs of $1/2$ to each agent.

	h_1	h_2	h_3
v_1	0, 1	0, 0	1, 0
v_2	1, 0	0, 1	0, 0
v_3	0, 0	1, 0	0, 1

The only probabilistic Nash equilibrium arises when both agents choose between their options with equal probability, for expected payoffs of $1/3$ to each, which involves all nine possible profiles.

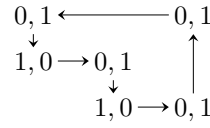
Probabilistic included in discrete and dynamic The following example shows that sometimes, probabilistic Nash equilibria can be more "subtle" than discrete and dynamic equilibria.

	h_1	h_2	h_3
v_1	0, 1	1, 0	0, 1
v_2	1, 0	0, 1	1, -7

If v puts all weight on one row, h will want to put all weight on the columns where he gets a payoff of 1, which will make v reassign weights toward the other row. If v puts weight on both rows, h will prefer the first column to the third, i.e. h_3 will be given probability 0. In other words, the only probabilistic Nash equilibrium involves v_1 , v_2 , h_1 , and h_2 with equal probabilities and expected payoffs of $1/2$. The only discrete and dynamic equilibrium involves all six profiles, with average payoffs of $1/2$ to v and $-2/3$ to h .

Disjoint Compromises Similarly, we can make several rows v -undesirable.

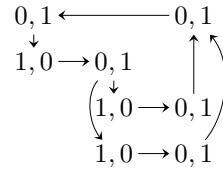
	h_1	h_2	h_3
v_1	0, 1	-7, 0	1, 0
v_2	1, 0	0, 1	-7, 0
v_3	-7, 0	1, 0	0, 1
v_4	0, 0	0, 0	0, 0



In any probabilistic Nash equilibrium, agent v chooses strategy v_4 with full weight and expected payoffs of 0. By contrast, the only discrete and dynamic equilibrium is disjoint from there, involving the previously-observed cycle around the cells with 1, 0 and 0, 1 and average payoffs of $1/2$.

Non-Trivial Overlaps The strategic game where only the last row is undesirable for v exhibits complementary features.

	h_1	h_2	h_3
v_1	0, 1	0, 0	1, 0
v_2	1, 0	0, 1	0, 0
v_3	0, 0	1, 0	0, 1
v_4	-7, 0	1, 0	0, 1



As before, v will avoid the row with the negative payoff and the only probabilistic Nash equilibrium involves the upper nine cells with equal probability and expected payoffs of $1/3$. The only discrete and dynamic equilibrium is as shown in the right-hand figure above, with average payoffs of $1/2$.

8.5 Conclusion

This chapter shows that it is possible to invoke Kakutani's fixed point theorem and partially generalise Nash's theorem in both BR and CP formalisms. However, these partial generalisations have a few drawbacks, so they are only mentioned quickly. While probabilistic Nash equilibrium is static and continuous, this chapter changes this mindset completely: it proposes a dynamic and discrete approach to compromising equilibrium. This approach is implemented independently in both BR and CP formalisms, and it turns out that the resulting notions of compromising equilibrium are almost the same: they are the SCC-sink of some well-chosen digraphs. It is worth noting that while the change-of-mind equilibrium generalises Nash equilibrium, the or-best response strict equilibrium generalises strict Nash equilibrium. It is so because there is no obvious way of generalising strict Nash equilibrium in CP games or Nash equilibrium in BR games. As compared to Nash's approach, which transforms finite games into continuous games, the discrete and dynamic approach does not introduce much complexity. In addition, finding the discrete and dynamic equilibria has a low algorithmic complexity. The discrete and dynamic equilibria also have a reasonable interpretation in terms of "limit cycles of sequential plays" in the BR or CP game: informally, if nodes represent the microscopic level and strongly connected components the macroscopic level,

then the compromising equilibria are the macroscopic equilibria of a microscopic world. Even though this discrete and dynamic view might not have as many practical applications as Nash's theorem does, they are useful in at least one respect: it shows that probabilities are not the only possible way of weakening the definition of Nash equilibrium and thus guaranteeing existence of weakened equilibrium.

Chapter 9

Discrete Non Determinism and Nash Equilibria for Strategy-Based Games

Several notions of game enjoy a Nash-like notion of equilibrium without guarantee of existence. There are different ways of weakening a definition of Nash-like equilibrium in order to guarantee the existence of a weakened equilibrium. Nash's approach to the problem for strategic games is probabilistic, *i.e.* continuous, and static. CP and BR approaches for CP and BR games are discrete and dynamic. This chapter proposes an approach that lies between those two different approaches: a discrete and static approach. Multi strategic games are introduced as a formalism that is able to express both sequential and simultaneous decision-making, which promises a good modelling power. Multi strategic games are a generalisation of strategic games and sequential graph games that still enjoys a Cartesian product structure, *i.e.* where agent actually choose their strategies. A pre-fixed point result allows guaranteeing existence of discrete and non deterministic equilibria. On the one hand, these equilibria can be computed with polynomial (low) complexity. On the other hand, they are effective in terms of recommendation, as shown by a numerical example.

9.1 Introduction

Not all strategic games have a (pure) Nash equilibrium. On the one hand, Nash's probabilistic approach copes with this existence problem with an ad hoc solution: Nash's solution is dedicated to a setting with real-valued payoff functions. On the other hand, CP and BR games propose an abstract and general approach that is applicable to many types of game. Both approaches generalise the notion of Nash equilibrium and guarantee the existence of a weakened Nash equilibrium. There are two main differences between the two approaches though. First, Nash's approach considers finite objects and yields continuous objects, whereas the CP and BR approach preserves finiteness. Second, Nash's approach is static, whereas the CP and BR approach is dynamic: Nash's approach is static because a probabilistic Nash equilibrium can be interpreted

as a probabilistic status quo that is (pure) Nash equilibrium of a probabilised game. The Cartesian product structure enables a static approach. CP and BR approach is dynamic because a CP or BR equilibrium can be interpreted as a limit set of states that are tied together by explicit forces. It may be interesting to mix features from both approaches, and to present for instance a discrete and static notion of equilibrium. Actually, such an approach was already adopted in [27], whose purpose was to provide sequential tree games with a notion of discrete non deterministic equilibrium. This approach assumed partially ordered payoffs, and simple "backward induction" guarantees existence of non deterministic subgame perfect equilibrium. This result is superseded by [29] which adopts a completely different approach, but the discrete non determinism spirit can be further exploited.

9.1.1 Contribution

This chapter introduces the concept of abstract strategic games, which corresponds to traditional strategic games where real-valued payoff functions have been replaced with abstract objects called outcomes. In addition, the usual total order over the reals has been replaced with binary relations, one per agent, that account for agent's preferences over the outcomes. Abstract strategic games thus generalise strategic games like abstract sequential tree games generalise sequential tree games. A notion of Nash equilibrium is defined, but not all abstract strategic games have a Nash equilibrium since traditional strategic games already lack this property.

Like Nash did for traditional strategic games, an attempt is made to introduce probabilities into these new games. However, it is mostly a failure because there does not seem to exist any extension of a poset to its barycentres that is relevant to the purpose. So, instead of saying that "an agent chooses a given strategy with some probability", this chapter proposes to say that "the agent may choose the strategy", without further specification.

The discrete non determinism proposed above is implemented in the notion of non deterministic best response (ndbr) multi strategic game. As hinted by the terminology, the best response approach is preferred over the convertibility preference approach for this specific purpose. (Note that discrete non determinism for abstract strategic games can be implemented in a formalism that is more specific and simpler than ndbr multi strategic games, but this general formalism will serve further purposes.) This chapter defines the notion of ndbr equilibrium in these games, and a pre-fixed point result helps prove a sufficient condition for every ndbr multi strategic game to have an ndbr equilibrium. An embedding of abstract strategic games into ndbr multi strategic games provides abstract strategic games with a notion of non deterministic (nd) equilibrium that generalises the notion of Nash equilibrium. Since every abstract strategic game has an nd equilibrium (under some condition), the discrete non deterministic approach succeeds where the probabilistic approach fails, *i.e.* is irrelevant. This new approach lies between Nash's approach, which is continuous and static, and the abstract approaches of CP and BR games, which are discrete and dynamic. Indeed, this notion of nd equilibrium is discrete and static. It is deemed static because it makes use of the Cartesian product structure, which allows interpreting an equilibrium as a "static state of the game".

This chapter also defines the notion of multi strategic game that is very similar to the notion of ndbr multi strategic game, while slightly less abstract. Multi strategic games are actually a generalisation of both abstract strategic games and sequential graph games. Informally, they are games where a strategic game takes place at each node of a graph. (A different approach to "games network" can be found in [33]) They can thus model within a single game both sequential and simultaneous decision-making mechanisms. An embedding of multi strategic games into ndbr multi strategic games provides multi strategic games with a notion of non deterministic (nd) equilibrium. In addition, a numerical example shows that the constructive proof of nd equilibrium existence can serve as a recommendation to agents on how to play, while the notion of Nash equilibrium, as it stands, cannot lead to any kind of recommendation.

9.1.2 Contents

Section 9.2 defines abstract strategic games and their abstract Nash equilibria. Section 9.3 considers probabilities to relax the definition of Nash equilibrium in abstract strategic games, and concludes that discrete non determinism is required. Section 9.4 proves a pre-fixed point result. Section 9.5 introduces the non deterministic best response multi strategic games and their non deterministic best response equilibria. Then it gives a sufficient condition for these games to have such an equilibrium. Section 9.6 embeds abstract strategic games into ndbr multi strategic games, and thus provides a notion of (existing) non deterministic equilibrium for abstract strategic games. It also gives a few examples. Section 9.7 defines multi strategic games and embeds them into ndbr multi strategic games, and thus provides a notion of non deterministic equilibrium for multi strategic games.

9.2 Abstract Strategic Games

This section defines abstract strategic games and their abstract Nash equilibria. Two embeddings show that abstract strategic games can be seen as either CP games or BR games when preferences are acyclic.

Informally, abstract strategic games are traditional strategic games where real-valued payoff functions have been replaced with abstract objects named outcomes. In addition for each agent, one binary relation over outcomes accounts for the preference of the agent for some outcomes over some others.

Definition 204 (Abstract strategic games) *Abstract strategic games are 4-tuples $\langle \mathcal{A}, S, P, (\overset{\circ}{\rightarrow}_a)_{a \in \mathcal{A}} \rangle$ where:*

- \mathcal{A} is a non-empty set of agents,
- $S = \bigotimes_{a \in \mathcal{A}} S_a$ is the Cartesian product of non-empty sets of individual strategies,
- $P : S \rightarrow Oc$ is a function mapping strategy profiles to outcomes.
- $\overset{\circ}{\rightarrow}_a$ is a binary relation over outcomes, and $oc \overset{\circ}{\rightarrow}_a oc'$ says that agent a prefers oc' over oc .

The example below shows a traditional strategic game on the left-hand side, and an abstract strategic game on the right-hand side.

	h_1	h_2		h_1	h_2
v_1	0	2	1	1	
v_2	0	1	2	0	

	h_1	h_2
v_1	oc_1	oc_2
v_2	oc_3	oc_4

For a given game, each agent can compare strategy profiles by comparing their outcomes using the function P .

Notation 205 Let $\langle \mathcal{A}, S, P, (\xrightarrow{a})_{a \in \mathcal{A}} \rangle$ be an abstract strategic game and let s and s' be in S .

$$s \xrightarrow{a} s' \triangleq P(s) \xrightarrow{a} P(s')$$

Happiness of an agent is defined below, in a convertibility preference style.

Definition 206 (Agent happiness) Let $\langle \mathcal{A}, S, P, (\xrightarrow{a})_{a \in \mathcal{A}} \rangle$

$$Happy(a, s) \triangleq \forall s' \in S, \neg(s'_{-a} = s_{-a} \wedge s \xrightarrow{a} s')$$

As usual, Nash equilibrium means happiness for all agents.

Definition 207 (Nash Equilibrium)

$$Eq_g(s) \triangleq \forall a \in \mathcal{A}, Happy(a, s)$$

There exists a natural embedding of abstract strategic games into CP games, as described below.

Lemma 208 Let $g = \langle \mathcal{A}, S, P, (\xrightarrow{a})_{a \in \mathcal{A}} \rangle$ be a strategic game.

$$s \xrightarrow{+a} s' \triangleq s_{-a} = s'_{-a},$$

Then $g' = \langle \mathcal{A}, S, (\xrightarrow{+a})_{a \in \mathcal{A}}, (\xrightarrow{a})_{a \in \mathcal{A}} \rangle$ is a CP game and the embedding preserves and reflects Nash equilibria.

$$Eq_g(s) \Leftrightarrow Eq_{g'}(s)$$

When agents' preferences are acyclic, there exists also a natural embedding of abstract strategic games into BR games, as described below.

Lemma 209 Let $g = \langle \mathcal{A}, S, P, (\xrightarrow{a})_{a \in \mathcal{A}} \rangle$ be a strategic game. Assume that the \xrightarrow{a} are acyclic.

$$BR_a(s) \triangleq \{s_{-a}\} \times \{s' \in S_a \mid \forall s'' \in S_a, \neg(s_{-a}; s' \xrightarrow{a} s_{-a}; s'')\}$$

Then $g' = \langle \mathcal{A}, S, (BR_a)_{a \in \mathcal{A}} \rangle$ is a BR game and the embedding preserves and reflects Nash equilibria.

$$Eq_g(s) \Leftrightarrow Eq_{g'}(s)$$

9.3 From Continuous to Discrete Non Determinism

This section tries to apply Nash's probabilistic compromise to an instance of abstract strategic games. Facing a half-failure, it notices that continuous non determinism, *i.e.* probabilities, carry "too much" information. Indeed, only a notion of discrete non deterministic strategies is needed to characterise the probabilistic Nash equilibria of the example. These non deterministic strategies are defined as non-empty subsets of strategies.

Consider the following abstract strategic game involving agents v and h . The game has no (pure) Nash equilibrium.

oc_1	$\xrightarrow{\bigcirc V}$	oc_3	oc_3	$\xrightarrow{\bigcirc H}$	oc_4	v_1	<table> <tr> <th>h_1</th> <th>h_2</th> </tr> <tr> <td>oc_1</td> <td>oc_2</td> </tr> <tr> <td>oc_3</td> <td>oc_4</td> </tr> </table>	h_1	h_2	oc_1	oc_2	oc_3	oc_4
h_1	h_2												
oc_1	oc_2												
oc_3	oc_4												
oc_4	$\xrightarrow{\bigcirc V}$	oc_2	oc_2	$\xrightarrow{\bigcirc H}$	oc_1	v_2	<table> <tr> <th>h_1</th> <th>h_2</th> </tr> <tr> <td>oc_1</td> <td>oc_2</td> </tr> <tr> <td>oc_3</td> <td>oc_4</td> </tr> </table>	h_1	h_2	oc_1	oc_2	oc_3	oc_4
h_1	h_2												
oc_1	oc_2												
oc_3	oc_4												

Mixed strategies for abstract strategic games are defined the same way they are defined for traditional strategic games, *i.e.* through probability distributions. Mixed strategy profiles are of the following form, where α and β are probabilities that are chosen by agent v and agent h respectively.

$$\alpha\beta(v_1, h_1) + \alpha(1 - \beta)(v_1, h_2) + (1 - \alpha)\beta(v_2, h_1) + (1 - \alpha)(1 - \beta)(v_2, h_2)$$

In the traditional setting, this yields expected payoff functions. Since payoffs and probabilities are both real numbers, the expected payoffs are also real numbers. It is therefore natural to compare them by using the usual total order over the reals, *i.e.* the same order that is used when comparing payoffs of pure strategy profiles. In the abstract setting though, mixing the strategies induces a new type of object, say "expected outcomes". These expected outcomes are objects of the following form, where α and β are probabilities that are chosen by agent v and agent h respectively.

$$\alpha\beta oc_1 + \alpha(1 - \beta)oc_2 + (1 - \alpha)\beta oc_3 + (1 - \alpha)(1 - \beta)oc_4$$

These new objects are not outcomes a priori, so there is no obvious way to compare them a priori. The most natural way may be the following one. When both α and β are either 0 or 1, the expected outcome looks like an outcome. For instance, if α and β equal 1, the expected outcome is as follows.

$$1.oc_1 + 0.oc_2 + 0.oc_3 + 0.oc_4$$

Along the two preference relations that are defined in the strategic game above, it is possible to define preferences among these four specific expected outcomes. For instance, $oc_1 \xrightarrow{\bigcirc v} oc_3$ yields the following.

$$1.oc_1 + 0.oc_2 + 0.oc_3 + 0.oc_4 \xrightarrow{\bigcirc v} 0.oc_1 + 0.oc_2 + 1.oc_3 + 0.oc_4$$

When either α or β is either 0 or 1, the new preference relations can be extended naturally, *i.e.* consistently with the original preferences. Informally, if agent v prefers oc_3 over oc_1 then, by extension, he will prefer mixed outcomes

giving more weight to oc_3 than to oc_1 . For instance, if $\alpha' < \alpha$ then the preference of agent v is extended as follows.

$$\alpha oc_1 + 0.oc_2 + (1 - \alpha)oc_3 + 0.oc_4 \xrightarrow{\circ v} \alpha' oc_1 + 0.oc_2 + (1 - \alpha')oc_3 + 0.oc_4$$

However, extending non-trivially the preferences to all expected outcomes would require an artificial choice. Indeed, consider the following two expected outcomes obtained by $\alpha = \frac{1}{2}$ and $\beta = \frac{1}{2}$ for the first one, and by $\alpha = \frac{1}{2}$ and $\beta = \frac{1}{3}$ for the second one.

$$\begin{array}{ccc} \frac{1}{4}oc_1 + \frac{1}{4}oc_2 & + & \frac{1}{4}oc_3 + \frac{1}{4}oc_4 \\ \frac{1}{6}oc_1 + \frac{1}{3}oc_2 & + & \frac{1}{6}oc_3 + \frac{1}{3}oc_4 \end{array}$$

In the strategic game above, these expected outcomes correspond to mixed strategy profiles that agent h can convert to each other. On the one hand, the weights attributed to oc_1 and oc_2 are better in the first expected outcome, according to h . On the other hand, the weights attributed to oc_3 and oc_4 are better in the second expected outcome, according to h . These two arguments sound "contradictory". Moreover, there is nothing in the original preference relation that suggests to give priority to one argument over the other. So it is reasonable to say that h prefers neither of these expected outcomes. Thus are defined the extensions of the preference relations. This completes the definition of the (probabilistic) abstract strategic game derived from the finite abstract strategic game example above. In such a setting, the probabilistic Nash equilibria are the mixed strategy profiles $\alpha\beta(v_1, h_1) + \alpha(1 - \beta)(v_1, h_2) + (1 - \alpha)\beta(v_2, h_1) + (1 - \alpha)(1 - \beta)(v_2, h_2)$, where $0 < \alpha < 1$ and $0 < \beta < 1$. Therefore almost all mixed strategy profiles are probabilistic Nash equilibria, which does not seem not be a desirable property.

In the probabilistic setting, a strategy of an agent is said to be "used" if the agent gives a non-zero probability to this strategy. With this terminology, the above remark can be rephrased as follows. In the above strategic game with its above probabilistic extension, a mixed strategy profile is a probabilistic Nash equilibrium *iff* both agents use both their strategies. This suggests that, in abstract strategic games, the actual values of the probabilities are irrelevant to the Nash equilibrium predicate. Only their being zero or not is relevant. This motivates the definition of discrete non deterministic strategies that only says which strategies are used. Note that an agent must use at least one strategy, like in the probabilistic (and the pure) setting. For each agent, a discrete non deterministic strategy can therefore be seen as a non-empty subset of the set of its strategies.

9.4 A Simple Pre-Fixed Point Result

This section proves a pre-fixed-point result, *i.e.* the existence of a y such that $y \preceq F(y)$ for all F and \preceq that comply with given constraints.

Meet semi-lattices are defined below like in the literature. They are posets that guarantee existence of greatest lower bound of any two elements. The terminology of "meet" seems to come from the set-theoretic intersection.

Definition 210 (Meet semi-lattice) A meet semi-lattice is a partially ordered set (S, \preceq) , i.e. the binary relation \preceq is reflexive and transitive, such that any sets with two elements has a greatest lower bound.

Defined as a specific type of posets, meet semi-lattices have algebraic properties. (Actually, meet semi-lattices are sometimes defined as algebraic structures from where an ordering is derived.)

Definition 211 In a partially ordered set, a greatest lower bound of two elements is unique, which induces the binary operator "greatest lower bound". This operator is commutative and associative, which enables the definition of the greatest lower bound of any non-empty finite subset of S . Let us call \inf this greatest lower bound function.

Given a function from a meet semi-lattice to itself, a meeting point is an element of the lattice such that every decreasing sequence that starts with the meeting point is not too much "scattered" by the function. This is accurately described below.

Definition 212 Let (S, \preceq) be a meet semi-lattice with least element m , and let \inf be the infimum function corresponding to \preceq . Let F be a function from S to itself and let x be in S . Assume that for all $x_1 \dots x_n$ such that $m \neq x_1 \preceq \dots \preceq x_n \preceq x$, we have $\inf(F(x_1), \dots, F(x_n), x) \neq m$. If $x \neq m$ then x is said to be a F -meeting point, and one writes $M_F(x)$.

The F -meeting point predicate is preserved by the greatest lower bound operator used with the image of the point by F , as stated below.

Lemma 213 Let (S, \preceq) be a meet semi-lattice with least element m ; let \inf be the infimum function corresponding to \preceq ; and let F be a function from S to itself. The following formula holds.

$$M_F(x) \Rightarrow M_F \circ \inf(x, F(x))$$

Proof Assume $M_F(x)$, so x and m are different. By reflexivity $x \preceq x$, so $\inf(x, F(x)) \neq m$ by definition of meeting point. Assume $m \neq x_1 \preceq \dots \preceq x_n \preceq \inf(x, F(x))$, so $m \neq x_1 \preceq \dots \preceq x_n \preceq x \preceq x$ since $\inf(x, F(x)) \preceq x$ by definition of \inf . So $\inf(F(x_1), \dots, F(x_n), F(x), x) \neq m$ since x is a F -meeting point. Therefore $\inf(F(x_1), \dots, F(x_n), \inf(F(x), x)) \neq m$, by associativity of the greatest lower bound operator underlying the infimum function \inf . So $\inf(x, F(x))$ is a F -meeting point. \square

The F -meeting point predicate preservation can be combined with the assumption that there exists no infinite strictly decreasing sequence. In this case, iteration of lemma 213 yields a non-trivial pre fixed point of F .

Lemma 214 Let (S, \preceq) be a meet semi-lattice with least element m , and assume that \preceq is well-founded. Let F be a function from S to itself. If there exists a F -meeting point, then there exists a F pre fixed point different from m , i.e. there exists $y \neq m$ such that $y \preceq F(y)$.

Proof Assume $M_F(x_0)$. An infinite sequence of elements of S is built as follows. It starts with x_0 , and it is gradually defined by induction. Assume x_n, \dots, x_0 such that $M_F(x_n)$, and $x_{k+1} = \inf(x_k, F(x_k))$ for all $0 \leq k < n$.

Let $x_{n+1} = \inf(x_n, F(x_n))$. By Lemma 213, $M_F(x_{n+1})$. By well-foundedness assumption, there exists n such that $x_{n+1} = x_n$, which means that $x_n = \inf(x_n, F(x_n))$, so $x_n \preceq F(x_n)$. Moreover, $M_F(x_n)$ by construction of the sequence, so $x_n \neq m$. \square

9.5 Non Deterministic Best Response Multi Strategic Games

Using the concept of discrete non deterministic strategy, this section defines non deterministic best response multi strategic games and their non deterministic Nash equilibria. This section also proves a sufficient condition for non deterministic Nash equilibrium existence in every non deterministic best response multi strategic game. These results will be used to guarantee existence of non deterministic equilibrium for abstract strategic games and non deterministic multi strategic game.

Informally, non deterministic best response multi strategic games involve agents who play on several (abstractions of) strategic games at the same time. Agents' strategies are non deterministic, *i.e.* on each game each agent has to choose one or more (pure) strategies among his available strategies. When all the opponents of an agent have chosen their non deterministic strategies on all games, a function tells agent a what his non deterministic best responses are. These games are formally defined below.

Definition 215 (Non deterministic best response multi strategic games)

An ndbr multi strategic game is a pair $\langle (S_a^i)_{a \in \mathcal{A}}^{i \in I}, (BR_a)_{a \in \mathcal{A}} \rangle$ complying with the following.

- I is a non-empty set of indices and \mathcal{A} is a non-empty set of agents.
- For all a in \mathcal{A} , BR_a is a function from Σ_{-a} to Σ_a ,
where $\Sigma = \bigotimes_{a \in \mathcal{A}} \Sigma_a$ and $\Sigma_a = \bigotimes_{i \in I} \mathcal{P}(S_a^i) - \{\emptyset\}$ and $\Sigma_{-a} = \bigotimes_{a' \in \mathcal{A} - \{a\}} \Sigma_{a'}$.

Elements of Σ_a are called nd strategies for a , and elements of Σ are called nd strategy profiles.

Informally, an agent is happy with an nd strategy profile if its own nd strategy is included in its best responses against other agents' nd strategies. Agents' happiness is formally defined as follows.

Definition 216 (Happiness) Let $g = \langle (S_a^i)_{a \in \mathcal{A}}^{i \in I}, (BR_a)_{a \in \mathcal{A}} \rangle$ be an ndbr multi strategic game, and let σ be in Σ .

$$Happy(\sigma, a) \triangleq \sigma_a \subseteq BR_a(\sigma_{-a})$$

As usual, (non deterministic) Nash equilibrium amounts to happiness for all agents.

Definition 217 (Non deterministic Nash equilibrium) Let g be an ndbr multi strategic game, $g = \langle (S_a^i)_{a \in \mathcal{A}}^{i \in I}, (BR_a)_{a \in \mathcal{A}} \rangle$.

$$Eq_g(\sigma) \triangleq \forall a \in \mathcal{A}, Happy(\sigma, a)$$

The individual best response functions can be combined into a collective best response function from the non deterministic profiles into themselves.

Definition 218 (Combined best response) *Given an ndbr multi strategic game with $(BR_a)_{a \in \mathcal{A}}$ a family of agent best responses. The combined best response is a function from Σ to itself defined as follows.*

$$BR(\sigma) \triangleq \bigotimes_{a \in \mathcal{A}} BR_a(\sigma_{-a})$$

Through the combined best response function, the non deterministic Nash equilibria are characterised below as nd profiles included in their images by the combined best response.

Lemma 219 *An ndbr equilibrium for $g = \langle (S_a^i)_{a \in \mathcal{A}}^{i \in I}, (BR_a)_{a \in \mathcal{A}} \rangle$ is characterised as follows.*

$$Eq_g(\sigma) \Leftrightarrow \sigma \subseteq BR(\sigma)$$

Like in BR games, it is easy to define agents' strict happiness.

Definition 220 (Strict happiness) *Let $g = \langle (S_a^i)_{a \in \mathcal{A}}^{i \in I}, (BR_a)_{a \in \mathcal{A}} \rangle$ be an ndbr multi strategic game, and let σ be in Σ .*

$$Happy^+(\sigma, a) \triangleq \sigma_a = BR_a(\sigma_{-a})$$

Then, (non deterministic) strict Nash equilibrium is defined as strict happiness for all agents.

Definition 221 (Non deterministic strict Nash equilibrium) *Let g be an ndbr multi strategic game, $g = \langle (S_a^i)_{a \in \mathcal{A}}^{i \in I}, (BR_a)_{a \in \mathcal{A}} \rangle$, and let σ be in Σ .*

$$Eq_g^+(\sigma) \triangleq \forall a \in \mathcal{A}, Happy^+(\sigma, a)$$

The following embedding of non deterministic best response multi strategic games into BR games preserves and reflects equilibria.

Lemma 222 *Let $g = \langle (S_a^i)_{a \in \mathcal{A}}^{i \in I}, (BR_a)_{a \in \mathcal{A}} \rangle$ be an ndbr multi strategic game. Define BR'_a with a Cartesian product below, for $\sigma \in \Sigma$.*

$$BR'_a(\sigma) \triangleq \{\sigma_{-a}\} \times \bigotimes_{i \in I} \mathcal{P}(BR_a^i(\sigma_{-a})) - \{\emptyset\}$$

Where $BR_a^i(\sigma_{-a})$ is the i -projection of $BR_a(\sigma_{-a})$. Then the object g' defined by $g' = \langle \mathcal{A}, \Sigma, (BR'_a)_{a \in \mathcal{A}} \rangle$ is a BR game and Nash equilibria correspond as follows.

$$Eq_{g'}(\sigma) \Leftrightarrow Eq_g(\sigma)$$

Proof Let σ be in $\Sigma = \bigotimes_{a \in \mathcal{A}} \bigotimes_{i \in I} (\mathcal{P}(S_a^i) - \{\emptyset\})$. The following chain of equivalences proves the claim. $Eq_{g'}(\sigma) \Leftrightarrow \forall a, \sigma \in BR'_a(\sigma) \Leftrightarrow \forall a, \sigma \in \{\sigma_{-a}\} \times \bigotimes_{i \in I} \mathcal{P}(BR_a^i(\sigma_{-a})) - \{\emptyset\} \Leftrightarrow \forall a, \sigma_a \in \bigotimes_{i \in I} \mathcal{P}(BR_a^i(\sigma_{-a})) - \{\emptyset\} \Leftrightarrow \forall a, \sigma_a \subseteq BR_a(\sigma_{-a}) \Leftrightarrow Eq_g(\sigma)$ \square

The remainder of the section invokes the fixed-point results of section 9.4, but prior to that, a meet lattice needs to be identified.

Lemma 223 *Given an ndbr multi strategic game $\langle (S_a^i)_{a \in \mathcal{A}}^{i \in I}, (BR_a)_{a \in \mathcal{A}} \rangle$, the poset $(\Sigma \cup \{\emptyset\}, \subseteq)$ is a meet semi-lattice with least element \emptyset .*

A first equilibrium existence result is given below.

Lemma 224 *Given an ndbr multi strategic game $\langle (S_a^i)_{a \in \mathcal{A}}^{i \in I}, (BR_a)_{a \in \mathcal{A}} \rangle$, if there exists a BR -meeting point, then there exists a non deterministic Nash equilibrium.*

Proof By lemma 214, if there exists a BR -meeting point, then there exists a non-empty pre fixed point σ for BR , i.e. there exists $\sigma \neq \emptyset$ such that $\sigma \subseteq BR(\sigma)$. It is an ndbr equilibrium by definition. \square

The main equilibrium existence result is stated below.

Lemma 225 *Consider an ndbr multi strategic game $\langle (S_a^i)_{a \in \mathcal{A}}^{i \in I}, (BR_a)_{a \in \mathcal{A}} \rangle$. Let σ be in Σ . Assume that for all agents a , for all $\gamma_1 \dots \gamma_n$ in Σ_{-a} , if $\gamma_n \subseteq \dots \subseteq \gamma_1 \subseteq \sigma_{-a}$ then $\cap_{1 \leq k \leq n} BR_a(\gamma_k) \cap \sigma_a \neq \emptyset$. In this case, the game has an ndbr equilibrium.*

Proof By lemma 224, it suffices to show that there exists a BR -meeting point. Let us prove that σ is such a meeting point. First of all, σ is non-empty since it belongs to Σ . Second, assume $\sigma^1 \dots \sigma^n$ in Σ such that $\sigma^1 \subseteq \dots \subseteq \sigma^n \subseteq \sigma$. So for all agents a , $\sigma_{-a}^1 \subseteq \dots \subseteq \sigma_{-a}^n \subseteq \sigma_{-a}$. By assumption, $\cap_{1 \leq k \leq n} BR_a(\sigma_{-a}^k) \cap \sigma_a \neq \emptyset$, which amounts to $(\cap_{1 \leq k \leq n} BR(\sigma^k) \cap \sigma)_a \neq \emptyset$. Since this holds for all a , we have $\cap_{1 \leq k \leq n} BR(\sigma^k) \cap \sigma \neq \emptyset$. Therefore σ is a BR -meeting point. \square

Equilibria are preserved when "increasing" the best response functions, as stated below.

Lemma 226 *Let $g = \langle (S_a^i)_{a \in \mathcal{A}}^{i \in I}, (BR_a)_{a \in \mathcal{A}} \rangle$ and $g' = \langle (S_a^i)_{a \in \mathcal{A}}^{i \in I}, (BR'_a)_{a \in \mathcal{A}} \rangle$ be two ndbr multi strategic games such that for all a in \mathcal{A} , for all γ in Σ_{-a} , $BR_a(\gamma) \subseteq BR'_a(\gamma)$. In this case, the following implication holds.*

$$\text{Eq}_g(\sigma) \Rightarrow \text{Eq}_{g'}(\sigma)$$

Proof Since for all a in \mathcal{A} , for all γ in Σ_{-a} , $BR_a(\gamma) \subseteq BR'_a(\gamma)$, it follows that for all σ in Σ , $BR(\sigma) \subseteq BR'(\sigma)$. So, $\sigma \subseteq BR(\sigma)$ implies $\sigma \subseteq BR'(\sigma)$. \square

9.6 Discrete and Static Compromising Equilibrium for Abstract Strategic Games

Subsection 9.6.1 embeds abstract strategic games into ndbr multi strategic games, and thus provides a notion of non deterministic Nash equilibrium for abstract strategic games. Finally it proves equilibrium existence for all abstract strategic games. Subsection 9.6.2 gives an example of such non deterministic Nash equilibrium on a strategic game with real-valued payoff functions. The building of the equilibrium is described step by step. Subsection 9.6.3 suggests by a numerical example that the constructive proof of existence of such an equilibrium yields a notion of recommendation that is better on average than playing randomly.

9.6.1 Non Deterministic Equilibrium Existence

The following definition extends orders over functions' codomains to orders over (restrictions of) functions.

Definition 227 Let f and f' be functions of type $A \rightarrow B$, and let A' be a subset of A . Let \prec be an irreflexive binary relation over B , and let \preceq be its reflexive closure.

$$\begin{aligned} f \preceq^{A'} f' &\triangleq \forall x \in A', f(x) \preceq f'(x) \\ f \prec^{A'} f' &\triangleq f \preceq^{A'} f' \wedge \exists x \in A', f(x) \prec f'(x) \end{aligned}$$

One simply writes $f \prec f'$ instead of $f \prec^A f'$.

For example if the two functions are represented by vectors of naturals with the usual total order, then $(1, 1) < (1, 2)$, $(0, 2) < (1, 2)$ and $(0, 1) < (1, 2)$.

The extension above preserves strict partial orders, as stated below.

Lemma 228 If \prec is a strict partial order over B , then the derived \prec^A over functions of type $A \rightarrow B$ is also a strict partial order.

Given finitely many functions of the same type, given an order on the codomain, given finitely many subsets of the domain that are totally ordered by inclusion, one of these functions is maximal for the extension order with respect to each of the subsets. This is proved by the following.

Lemma 229 Let E be a finite set of functions of type $A \rightarrow B$. Let \prec be an irreflexive and transitive binary relation over B , and let \preceq be its reflexive closure. For any $\emptyset \neq A_0 \subseteq \dots \subseteq A_n \subseteq A$ there exists an f in E that is maximal with respect to all the extended orders from \prec^{A_0} to \prec^{A_n} .

Proof By induction on n . First case, $n = 0$. There exists an f in E that is maximal, i.e. have no successor, with respect to \prec^{A_0} since \prec^{A_0} is a partial order. Second case, assume that the claim holds for n and let $\emptyset \neq A_0 \subseteq \dots \subseteq A_{n+1}$ be subsets of A . Let f in E be maximal with respect to all the $\prec^{A_0} \dots \prec^{A_n}$. Let f' be maximal with respect to $\prec^{A_{n+1}}$ among the functions in E that coincide with f on A_n . Since $\emptyset \neq A_0 \subseteq \dots \subseteq A_n$, the function f' is maximal in E with respect to all the $\prec^{A_0} \dots \prec^{A_n}$. Let f'' in E be such that $f' \preceq^{A_{n+1}} f''$. So $f' \preceq^{A_n} f''$ since $A_n \subseteq A_{n+1}$. Therefore f' and f'' coincide on A_n by maximality of f' with respect to \prec^{A_n} . Since f and f' coincide on A_n , the functions f and f'' also coincide on A_n . Therefore $\neg(f' \prec^{A_{n+1}} f'')$ by definition of f' , which shows that f' is also maximal with respect to $\prec^{A_{n+1}}$. \square

Below, an ndbr multi strategic game is built from an abstract strategic game. This ndbr multi strategic game always has an ndbr equilibrium. In the notation $s \xrightarrow[\gamma]{\circ_a} s'$ below, the strategies s and s' are seen as functions from S_a to the outcomes.

Lemma 230 Let $g = \langle \mathcal{A}, S, P, (\xrightarrow[\gamma]{\circ_a})_{a \in \mathcal{A}} \rangle$ be an abstract strategic game. Assume that the $\xrightarrow[\gamma]{\circ_a}$ are strict partial orders, i.e. irreflexive and transitive. For each agent a and

each γ in Σ_{-a} , the following defines a subset of S_a .

$$BR_a(\gamma) \triangleq \{s \in S_a \mid \begin{array}{l} \forall s' \in S_a, \neg(s \xrightarrow{a}_{S_{-a}} s') \quad \wedge \\ \forall s' \in S_a, \neg(s \xrightarrow{a}_{\gamma} s') \quad \wedge \\ \exists c \in \gamma, \forall s' \in S_a, \neg(s \xrightarrow{a}_{\{c\}} s') \end{array} \}$$

The object $\langle (S_a)_{a \in \mathcal{A}}, (BR_a)_{a \in \mathcal{A}} \rangle$ is an ndbr multi strategic game, and it has an ndbr equilibrium.

Proof Let a be an agent. First note that, through the function P , every strategy s in S_a can be seen as a function from S_{-a} to the outcomes Oc . Let $\gamma_1 \subseteq \dots \subseteq \gamma_n$ be in Σ_{-a} and let c be in γ_1 . According to lemma 229, there exists an s in S_a that is maximal with respect to $\xrightarrow{a}_{\{c\}}, \xrightarrow{a}_{\gamma_1}, \dots, \xrightarrow{a}_{\gamma_n}, \xrightarrow{a}_{S_{-a}}$. By definition of BR_a (third conjunct), this strategy s is in all the $BR_a(\gamma_1), \dots, BR_a(\gamma_n)$. First, this shows that BR_a returns non-empty sets, so $\langle (S_a)_{a \in \mathcal{A}}, (BR_a)_{a \in \mathcal{A}} \rangle$ is indeed an ndbr multi strategic game. Second, this game has an ndbr equilibrium by lemma 225. \square

The ndbr equilibrium for the derived ndbr multi strategic game is called a non deterministic equilibrium for the original abstract strategic game. Other similar definitions are possible for non deterministic equilibrium. Especially definitions more generous than the one in lemma 230, which guarantee the existence of a non deterministic equilibrium, according to lemma 226. This yields the following lemma.

Lemma 231 Let $g = \langle \mathcal{A}, S, P, (\xrightarrow{a})_{a \in \mathcal{A}} \rangle$ be a strategic game. Assume that the \xrightarrow{a} are strict partial orders, i.e. irreflexive and transitive. For each agent a and each γ in Σ_{-a} , the following defines three subsets of S_a .

$$\begin{aligned} BR_a^1(\gamma) &\triangleq \{s \in S_a \mid \forall s' \in S_a, \neg(s \xrightarrow{a}_{S_{-a}} s')\} \\ BR_a^2(\gamma) &\triangleq \{s \in S_a \mid \forall s' \in S_a, \neg(s \xrightarrow{a}_{\gamma} s')\} \\ BR_a^3(\gamma) &\triangleq \{s \in S_a \mid \exists c \in \gamma, \forall s' \in S_a, \neg(s \xrightarrow{a}_{\{c\}} s')\} \\ BR_a^4(\gamma) &\triangleq \{s \in S_a \mid \forall s' \in S_a, \exists c \in \gamma, \neg(s \xrightarrow{a}_{\{c\}} s')\} \end{aligned}$$

The object $\langle \mathcal{A}, S, (BR_a^i)_{a \in \mathcal{A}} \rangle$, for i between 1 and 4, is an ndbr multi strategic game, and it has an ndbr equilibrium.

The first three BR_a^i above correspond to the three conjuncts of the BR_a of lemma 230, and BR_a^4 is even more generous than BR_a^2 . Note that BR^2 and BR^4 somewhat relate to the notions of dominated strategy, studied in [15] and [32], and rationalizability, studied in [8] and [43]. These notions are more recently discussed in [18], for instance. (These notions are also related to "backward induction", but this thesis does not further explore the matter.)

The rest of the subsection discusses a few properties of these equilibria.

Definition 232 (Cartesian union) Let $\bigotimes_{i \in I} A_i$ be a cartesian product. The Cartesian union is defined as follows within $\bigotimes_{i \in I} A_i$.

$$p_i(D) \triangleq \{x_i \mid x \in D\}$$

$$B \cup^\times C \triangleq \bigotimes_{i \in I} p_i(B) \cup p_i(C)$$

Where p_i is the projection on A_i .

The equilibria related to BR^1 define a simple structure, as stated below.

Lemma 233 *The ndbr equilibria related to BR^1 are the elements of Σ that are included in $\bigotimes_{a \in \mathcal{A}} \{s \in S_a \mid \forall s' \in S_a, \neg(s \xrightarrow{a}_{S_{-a}} s')\}$.*

Proof For all σ in Σ , $BR^1(\sigma) = \bigotimes_{a \in \mathcal{A}} \{s \in S_a \mid \forall s' \in S_a, \neg(s \xrightarrow{a}_{S_{-a}} s')\}$. \square

The following lemma states that the ndbr equilibria related to BR^3 define a Cartesian union lattice.

Lemma 234 *Let $g = \langle \mathcal{A}, S, P, (\xrightarrow{a})_{a \in \mathcal{A}} \rangle$ be a strategic game. Assume that the \xrightarrow{a} are strict partial orders, i.e. irreflexive and transitive. If equilibrium is defined through BR^3 then the following holds.*

$$Eq_g(\sigma) \wedge Eq_g(\sigma') \Rightarrow Eq_g(\sigma \cup^\times \sigma')$$

Proof It suffices to prove that $\sigma \cup^\times \sigma' \subseteq BR^3(\sigma \cup^\times \sigma')$. Let s be in $\sigma \cup^\times \sigma'$. If s is in σ then it is also in $BR^3(\sigma)$ by equilibrium assumption. So for every agent a , there exists c in σ_{-a} (so c is also in $\sigma_{-a} \cup^\times \sigma'_{-a}$) such that for all s' in S_a , $\neg(s \xrightarrow{a}_{\{c\}} s')$. Therefore s is in $BR^3(\sigma \cup^\times \sigma')$. Same scenario if s is in σ' . So $\sigma \cup^\times \sigma' \subseteq BR^3(\sigma \cup^\times \sigma')$. \square

The following lemma states that the ndbr equilibria related to BR^4 define a Cartesian union lattice.

Lemma 235 *Let $g = \langle \mathcal{A}, S, P, (\xrightarrow{a})_{a \in \mathcal{A}} \rangle$ be a strategic game. Assume that the \xrightarrow{a} are strict partial orders, i.e. irreflexive and transitive. If equilibrium is defined through BR^4 then the following holds.*

$$Eq_g(\sigma) \wedge Eq_g(\sigma') \Rightarrow Eq_g(\sigma \cup^\times \sigma')$$

Proof It suffices to prove that $\sigma \cup^\times \sigma' \subseteq BR^4(\sigma \cup^\times \sigma')$. Let s be in $\sigma \cup^\times \sigma'$. If s is in σ then it is also in $BR^4(\sigma)$ by equilibrium assumption. So for all s' in S_a , there exists c in σ_{-a} such that $\neg(s \xrightarrow{a}_{\{c\}} s')$, and each of these c also belongs to $\sigma_{-a} \cup^\times \sigma'_{-a}$. Therefore s is in $BR^4(\sigma \cup^\times \sigma')$. Same scenario if s is in σ' . So $\sigma \cup^\times \sigma' \subseteq BR^4(\sigma \cup^\times \sigma')$. \square

9.6.2 Example

The proof of existence of an ndbr equilibrium is constructive, so it provides for free an algorithm that computes such an equilibrium. The time complexity of the algorithm is polynomial with respect to the number of strategy profiles $|S|$ (when each agent has at least two available strategies). Indeed informally, each call to BR dismisses at least one strategy of one agent, so it dismisses at least one profile. Therefore BR is called at most $|S|$ times. Each use of BR invokes all the BR_a , which needs (at most) to consider each agent strategy and decide whether or not this strategy is a best response. So there are at most $|S|$ such decisions. Such a decision requires at most $3 \times |S|$ calls to a \xrightarrow{a} . Therefore

the time complexity of finding an equilibrium is at most cubic in the number of profiles $|S|$. This is a very rough approximation whose only purpose is to show that complexity is polynomial.

The example below is a strategic game with natural-valued payoff functions. Preference between naturals invokes the usual total order over the naturals. Let us apply the equilibrium algorithm to the game.

	h_1	h_2	h_3	h_4	h_5
v_1	0 0	3 2	2 2	2 1	3 0
v_2	3 3	0 2	0 0	2 1	3 2
v_3	2 2	1 0	0 1	3 1	0 2
v_4	1 0	1 2	1 2	1 2	1 0
v_5	2 0	1 0	0 0	0 1	0 0

Informally, V may "play" either v_1 or v_2 or v_3 or v_4 or v_5 . In that context, column h_5 is smaller than h_1 according to agent H , so h_5 is not a best response of agent H . In the same way row v_5 is smaller than row v_3 . In addition, row v_4 is not a best response of agent V because for each column h_i , row v_4 is smaller than some other row. Therefore, the combined best responses for the whole game are rows v_1 to v_3 and columns h_1 to h_4 , as shown below.

	h_1	h_2	h_3	h_4
v_1	0 0	3 2	2 2	2 1
v_2	3 3	0 2	0 0	2 1
v_3	2 2	1 0	0 1	3 1

Now, agent V may play either v_1 or v_2 or v_3 . Column h_4 is not a best response of agent H because for each row v_1 to v_3 , column h_4 is smaller than some other column. So the game "shrinks" again as show below.

	h_1	h_2	h_3
v_1	0 0	3 2	2 2
v_2	3 3	0 2	0 0
v_3	2 2	1 0	0 1

In the same way, for columns h_1 to h_3 , row v_3 is smaller than some other row.

	h_1	h_2	h_3
v_1	0 0	3 2	2 2
v_2	3 3	0 2	0 0

Column h_3 is smaller than column h_2 . This yield the following irreducible game.

	h_1	h_2
v_1	0 0	3 2
v_2	3 3	0 2

Therefore $\{v_1, v_2\} \times \{h_1, h_2\}$ is a non deterministic equilibrium for the original game.

9.6.3 Comparisons

This section first shows that playing according to the equilibrium that is built by the proof of existence is better than playing randomly. Second, it shows that probabilistic Nash equilibrium cannot serve as a recommendation. Finally, it relates the non deterministic equilibria of abstract strategic games and the orbest response strict equilibria.

Consider the following class of games G , where $*$ can take two values, namely 1 and -1 , and where the preferences are along the usual order $-1 < 1$.

	h_1	h_2
v_1	$*$ $*$ $*$ $*$	$*$ $*$ $*$ $*$
v_2	$*$ $*$ $*$ $*$	$*$ $*$ $*$ $*$

For each agent, the arithmetic mean of its payoff over the four payoff functions of all games in the class is 0, by a "simple symmetry". However for each agent, the arithmetic mean of its payoff over the payoff functions inside the ndbr equilibrium of all games in the class is $3/8$.

Lemma 236 For a game g in G , let $eq(g)$ be the ndbr equilibrium built by the proof of lemma 230.

$$\frac{1}{|G|} \times \sum_{g \in G} \frac{1}{|eq(g)|} \times \sum_{s \in eq(g)} P(s, v) = \frac{3}{8}$$

Proof In the games of class G , considering only the payoffs of one agent yields matrices like below. The first row displays the matrices whose two rows are equivalent to agent v (the agent choosing the row of the matrix). The second row displays the matrices whose two rows are not equivalent to agent v . The figure on the top of matrices represent how many actual matrices they represent up to rotation.

1	2	2	1																
<table><tr><td>1</td><td>1</td></tr><tr><td>1</td><td>1</td></tr></table>	1	1	1	1	<table><tr><td>1</td><td>-1</td></tr><tr><td>1</td><td>-1</td></tr></table>	1	-1	1	-1	<table><tr><td>1</td><td>-1</td></tr><tr><td>-1</td><td>1</td></tr></table>	1	-1	-1	1	<table><tr><td>-1</td><td>-1</td></tr><tr><td>-1</td><td>-1</td></tr></table>	-1	-1	-1	-1
1	1																		
1	1																		
1	-1																		
1	-1																		
1	-1																		
-1	1																		
-1	-1																		
-1	-1																		
4	2	4																	
<table><tr><td>1</td><td>1</td></tr><tr><td>1</td><td>-1</td></tr></table>	1	1	1	-1	<table><tr><td>1</td><td>1</td></tr><tr><td>-1</td><td>-1</td></tr></table>	1	1	-1	-1	<table><tr><td>1</td><td>-1</td></tr><tr><td>-1</td><td>-1</td></tr></table>	1	-1	-1	-1					
1	1																		
1	-1																		
1	1																		
-1	-1																		
1	-1																		
-1	-1																		

The payoff that is induced by the first matrix is 1: whatever may be the matrix of agent h , agent v gets 1. In the same way, the fourth matrix induces -1 . Now assume that the payoff matrix of agent v is the second one. When the payoff matrix of agent h ranges over the matrices above, $eq(g)$ will involve sometimes only the left-hand side, sometimes only the right-hand side and sometimes (the same number of times by symmetry), and sometimes both. So on average, the third matrix yields payoff 0. Same scenario for the third matrix.

The payoff that is induced by the sixth matrix is 1. The fifth matrix induces payoff 1 too, because an equilibrium involves only the first row whatever the matrix of agent h is. As to the seventh matrix it induces payoff 0 "by symmetry", because an equilibrium involves only the first row whatever the matrix of agent h is. Therefore the arithmetic mean is $\frac{1+4+2-1}{16} = \frac{3}{8}$. \square

Now, consider a 2-agent real-life situation that is modelled below in two different (yet correct) ways by agent v , to the left, and agent h , to the right. Each game has only two probabilistic Nash equilibria, namely $((v_1 \mapsto \frac{1}{2}, v_2 \mapsto \frac{1}{2}), (h_1 \mapsto \frac{1}{2}, h_2 \mapsto \frac{1}{2}))$ and $((v_3 \mapsto \frac{1}{2}, v_2 \mapsto \frac{1}{2}), (h_3 \mapsto \frac{1}{2}, h_2 \mapsto \frac{1}{2}))$. (Idea of a proof: if agents use all their strategies, the weights given to v_1 and v_3 , resp. h_1 and h_3 , are the same, by contradiction. This yields a contradiction.) However, there is no way for agents to collectively choose one of them in a non-cooperative setting.

	h_1	h_2	h_3		h_3	h_2	h_1
v_1	3 2	2 3	0 0	v_3	3 2	2 3	0 0
v_2	2 3	3 2	2 3	v_2	2 3	3 2	2 3
v_3	0 0	2 3	3 2	v_1	0 0	2 3	3 2

So, probabilistic Nash equilibria cannot serve as recommendations to agents on how to play because even for strategic games over rational numbers there is no algorithm that finds such an equilibrium and that is robust to strategy renaming. However, this paper's equilibrium computation is robust to strategy renaming because the proof/algorithm is independent from the names of the strategies. Since it is more efficient than random play, it can serve as a recommendation.

The rest of the subsection establishes a connection between the non deterministic equilibria of abstract strategic games and the or-best response strict equilibria of abstract strategic games seen as BR games.

Lemma 237 *Let $g = \langle \mathcal{A}, S, P, (\overset{\circ}{\rightarrow})_{a \in \mathcal{A}} \rangle$ be an abstract strategic game. Assume that the $\overset{\circ}{\rightarrow}_a$ are strict partial orders, i.e. irreflexive and transitive. For each agent a and each γ in Σ_{-a} , the following defines a subset of S_a .*

$$BR_a^3(\gamma) \triangleq \{s \in S_a \mid \exists c \in \gamma, \forall s' \in S_a, \neg(s \overset{\circ}{\rightarrow}_{\{c\}} s')\}$$

The object $g^3 = \langle \mathcal{A}, S, (BR_a^3)_{a \in \mathcal{A}} \rangle$ is an ndbr multi strategic game.

For each agent a and each s in S , the following defines a non-empty subset of S , more specifically an element of Σ .

$$BR'_a(s) \triangleq \{s_{-a}\} \times \{s' \in S_a \mid \forall s'' \in S_a, \neg(s' \overset{\circ}{\rightarrow}_{\{s_{-a}\}} s'')\}$$

The object $g' = \langle \mathcal{A}, S, (BR'_a)_{a \in \mathcal{A}} \rangle$ is a BR game. Then the following holds.

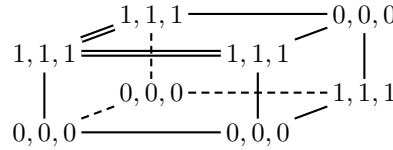
$$Eq_{g'}^+(C) \Rightarrow Eq_g(p(C))$$

Where $p(C) = \bigotimes_{a \in \mathcal{A}} p_a(C)$ is the smallest (for set inclusion) ndbr multi strategy profile including C .

Proof By lemma 231, g^3 is an ndbr multi strategic game, and by lemma 209, g' is a BR game. By assumption and by definition of or-best response strict equilibrium, $C = BR^{\cup}(C)$. One must show that $p(C) \subseteq BR^3(p(C))$. Let s be in $p(C)$. It suffices to prove that s_a is in $BR_a^3(p_{-a}(C))$ for all a . Let a be an agent. By definition of p , s_a is in $p_a(C)$, and by definition of p_a there exists c in S_{-a} such that $c; s_a$ is in C . Such a c is in $p_{-a}(C)$. Let $s_1 \rightarrow_a s_2$ stand for $s_2 \in BR'_a(s_1)$ and \rightarrow stand for $\bigcup_{a \in \mathcal{A}} \rightarrow_a$. Case split on whether or not

$c; s_a \rightarrow_a c; s_a$. If $c; s_a \rightarrow_a c; s_a$ then, by definition, $\neg s_a \xrightarrow[\{c\}]{\bigcirc_a} s'$ for all s' in S_a , so s_a is in $BR_a^3(p_{-a}(C))$. Now assume that $c; s_a \not\rightarrow_a c; s_a$. By definition, BR_a^3 returns non-empty sets, so $c; s_a \not\rightarrow_a c; s'_a$ for some $s'_a \neq s_a$. Recall that C is a \rightarrow strongly connected component, so $c; s'_a \rightarrow^+ c; s_a$, where \rightarrow^+ is the transitive closure of \rightarrow . This decomposes in $c; s'_a \rightarrow^+ c'; s''_a \rightarrow_a c'; s_a \rightarrow^+ c; s_a$ for some c' and s''_a . More specifically, $c'; s''_a \rightarrow_a c'; s_a$, with c' in $p_{-a}(C)$. This implies that s_a is in $BR_a^3(p_{-a}(C))$. \square

The ndbr equilibrium given by the implication $Eq_g^+(C) \Rightarrow Eq_g(p(C))$ above is not always strict. More specifically, the following example shows that $Eq_g^+(C) \wedge \neg Eq_g^+(p(C))$ for some or-best response equilibrium C . The example involves three agents each of whose converts along one dimension of the cube. Outcomes are payoff functions. The three nodes linked by doublelines constitute an or-best response strict equilibrium. The projection of the equilibrium is made of the four upper nodes, and the combined best response to this is the whole cube due to the two right-hand nodes in the back.



9.7 Discrete and Static Compromising Equilibrium for Multi Strategic Games

This section defines multi strategic games and their non deterministic strategy profiles. Then, it defines a notion of preference among sequences of sets of outcomes, which yields a notion of (global) nd equilibrium. An embedding of these multi strategic games into the ndbr multi strategic games shows that multi strategic games always have an nd equilibrium. Since sequential graph games can be embedded into multi strategic games, this also provides sequential graph games with a notion of nd equilibrium that enjoys guarantee of existence. More subtle notions of equilibrium could be defined, which would yield stronger results. However, the notion that is defined here intends to be a simple one. In addition, the section discusses a relevant embedding of strategic games into multi strategic games.

multi strategic games have a graph-like structure. At each node of the game, all agents play a strategic game to choose the outcome as well as the move to the next node. Another strategic game corresponds to this next node. So, a play in a multi strategic game is an infinite sequence of local plays in strategic games followed by outcomes. The notion of multi strategic game is defined below.

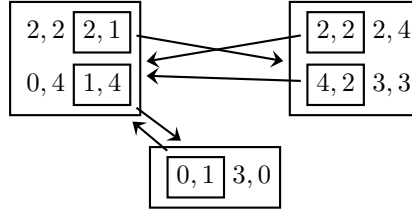
Definition 238 (multi strategic game) A multi strategic game is a pair $\langle S, (P^n)_{n \in V} \rangle$ that complies with the following.

- $S = \bigotimes_{n \in V} \bigotimes_{a \in A} S_a^n$, where V is a non-empty set of indices, A is a non-empty set of agents, and S_a^n is a non-empty set of strategies.

- P^n is of type $S^n \rightarrow OC \times V$, where OC is a non-empty set of outcomes.

The agents' (pure) strategies are the elements of $S_a = \bigotimes_{n \in V} S_a^n$ and the (pure) strategy profiles are the elements of S .

The following example depicts a multi strategic game that involves two agents, say *vertical* and *horizontal*. At each node *vertical* chooses the row and *horizontal* chooses the column. The game involves natural-valued payoff functions. The first figure corresponds to agent *vertical* and the second figure to agent *horizontal*. At a node, if a payoff function is enclosed in a small box with an arrow pointing from the box to another node, it means that the corresponding strategy profile leads to this other node. If a payoff function is not enclosed in a small box, it means that the corresponding strategy profile leads to the same node. For instance below, If the play start at the top-left node and if the agents choose the top-left profile, then both agents get payoff 2 and the same strategic game is played again. Whereas if the agents choose the top-right profile, *vertical* gets payoff 2 and *horizontal* gets payoff 1, and the agents have to play in the top-right node.



Starting from every node, a strategy profile induces infinite sequences of outcomes. It is therefore possible to define local/global equilibria for multi strategic games in the same way that they are defined for sequential graph games. Then, it is possible to embed sequential graph games into multi strategic games in a way that preserves and reflects local/global equilibria. (The embedding consists in seeing a node of a sequential graph game as a 1-agent strategic game.)

It is possible to embed strategic games into multi strategic games such that Nash equilibria correspond to local/global equilibria. (The embedding consists in seeing a strategic game as a one-node multi strategic game looping on itself).

However, since not all strategic games have a Nash equilibrium, not all multi strategic games have a local/global equilibrium. That is why non determinism comes into play.

Definition 239 (Non deterministic strategies and profiles) Let $\langle S, (P^n)^{n \in V} \rangle$ be a multi strategic game. Let $\Sigma_a^n \triangleq \mathcal{P}(S_a^n) - \{\emptyset\}$ be the set of local nd strategies of agent a at node n . Let $\Sigma_a \triangleq \bigotimes_{n \in V} \Sigma_a^n$ be the set of nd strategies of agent a (accounting for choices at all nodes). Let $\Sigma \triangleq \bigotimes_{a \in \mathcal{A}} \Sigma_a$ be the set of nd strategy profiles. Also, let $\Sigma^n \triangleq \bigotimes_{a \in \mathcal{A}} \Sigma_a^n$ be the set of local nd strategy profiles at node n . For σ in Σ , the objects σ_a , σ^n , and σ_a^n correspond to the projections of σ on Σ_a , Σ^n , and Σ_a^n .

Consider an nd strategy profile. At a given node, the outcome and the next node that are prescribed by the profile may be undetermined, because of non

determinism. However, the outcome is element of a determined set of outcomes and the next node is element of a determined set of nodes. The same phenomenon arises at each possible next nodes. Therefore, any path starting from a node and non deterministically following the prescription of the nd strategy profile yields a sequence whose k th element is element of a determined set of outcomes. These sets are defined below.

Definition 240 (Induced sequence of sets of outcomes) Let $g = \langle S, (P^n)^{n \in V} \rangle$ be a multi strategic game and let σ be in Σ . The induced sequence $seq(\sigma, n)$ is an infinite sequence of non-empty subsets of outcomes.

$$seq(\sigma, n) \triangleq fst \circ P^n(\sigma^n) \cdot seq(\sigma, snd \circ P^n(\sigma^n))$$

Where $f(Z) = \cup_{x \in Z} f(x)$ for any $f : X \rightarrow Y$ and $Z \subseteq X$, and the projection operations are defined such that $fst((x, y)) = x$ and $snd((x, y)) = y$.

Inclusion of nd strategy profiles implies component-wise inclusion of their sequences of sets of outcomes, as stated below.

Lemma 241 Let $g = \langle S, (P^n)^{n \in V} \rangle$ be a multi strategic game.

$$\sigma \subseteq \sigma' \Rightarrow \forall n \in V, \forall k \in \mathbb{N}, \quad seq(\sigma, n)(k) \subseteq seq(\sigma', n)(k)$$

For every agent, a preference binary relation over outcomes can be extended to a preference over sequences of sets of outcomes, as defined below. This extended preference amounts to component-wise preference for all sets in the sequence. However, there are other "natural" ways of extending preference over outcomes to preference over sequences of sets of outcomes (using limit sets or lexicographic ordering, for instance). So, what follows is only an example.

Definition 242 (Preference over sequences) Let \prec be a binary relation over a set E . It can be extended to non-empty subsets of E as follows.

$$X \prec^{set} Y \triangleq \forall x \in X, \forall y \in Y, \quad x \prec y$$

This can be extended to sequences of non-empty subsets of E . Below, α and β are sequences of non-empty subsets of E .

$$\alpha \prec^{fct} \beta \triangleq \forall k \in \mathbb{N}, \quad \alpha(k) \prec^{set} \beta(k)$$

The definition of \prec^{set} implies the following result.

Lemma 243 $\emptyset \neq X' \subseteq X \quad \wedge \quad \emptyset \neq Y' \subseteq Y \quad \wedge \quad X \prec^{set} Y \Rightarrow X' \prec^{set} Y'$

Next lemma states that the preference extension \prec^{fct} preserves an ordering property.

Lemma 244 If \prec is a strict partial order then \prec^{fct} is also a strict partial order.

The following result shows a preservation property involving \prec^{fct} and strategy inclusion.

Lemma 245 Let $g = \langle S, (P^n)^{n \in V} \rangle$ be a sequential graph game. Assume that $\gamma \subseteq \delta$ are both in Σ_{-a} . Then the following holds.

$$\text{seq}(\delta; \sigma_a, n) \xrightarrow{a}_{fct} \text{seq}(\delta; \sigma'_a, n) \Rightarrow \text{seq}(\gamma; \sigma_a, n) \xrightarrow{a}_{fct} \text{seq}(\gamma; \sigma'_a, n)$$

Proof By assumption and definition, $\text{seq}(\delta; \sigma_a, n)(k) \xrightarrow{a}_{set} \text{seq}(\delta; \sigma'_a, n)(k)$ for all k in \mathbb{N} . Since $\gamma \subseteq \delta$, lemma 241 implies that $\text{seq}(\gamma; \sigma_a, n)(k) \subseteq \text{seq}(\delta; \sigma_a, n)(k)$ and $\text{seq}(\gamma; \sigma'_a, n)(k) \subseteq \text{seq}(\delta; \sigma'_a, n)(k)$. Since these sets are all non-empty and by lemma 243, $\text{seq}(\gamma; \sigma_a, n)(k) \xrightarrow{a}_{set} \text{seq}(\gamma; \sigma'_a, n)(k)$ for all k . \square

Below, multi strategic game are embedded into ndbr multi strategic games. The corresponding ndbr multi strategic games always has an ndbr equilibrium, which is interpreted as multi strategic games always having an nd equilibrium. However, the embedding is not the only relevant one. It is intended to be a simple one. More subtle embeddings can yield to stronger results of nd equilibrium existence.

Lemma 246 Let $g = \langle S, (P^n)^{n \in V} \rangle$ be a multi strategic game. For each agent a , assume that his preference over outcomes \xrightarrow{a} are strict partial orders, i.e. irreflexive and transitive. For each agent a and each γ in Σ_{-a} , the following defines an element of Σ_a .

$$BR_a(\gamma) \triangleq \bigotimes_{i \in I} BR_a^n(\gamma)$$

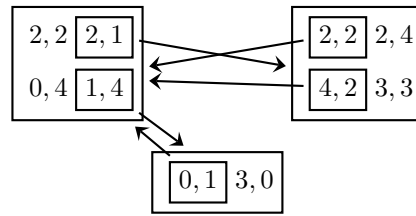
Where

$$BR_a^n(\gamma) \triangleq \{s^n \mid s \in S_a \wedge \forall s' \in S_a, \neg(\text{seq}(\gamma; s, n) \xrightarrow{a}_{fct} \text{seq}(\gamma; s', n))\}$$

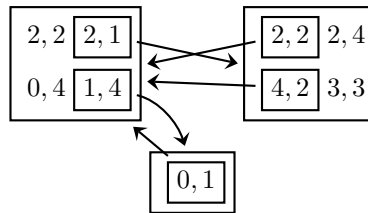
The object $\langle (S_a^n)_{a \in \mathcal{A}}, (BR_a)_{a \in \mathcal{A}} \rangle$ is an ndbr multi strategic game, and it has an ndbr equilibrium.

Proof First prove that $BR_a^n(\gamma)$ is non-empty: the set of the $\text{seq}(\gamma; s, n)$, when s ranges over S_a , is finite and non-empty. So at least one of the $\text{seq}(\gamma; s, n)$ is maximal with respect to \xrightarrow{a}_{fct} which is a strict partial order by lemma 244. Second, assume that $\gamma \subseteq \delta$ for γ and δ in Σ_{-a} and prove $BR_a^n(\gamma) \subseteq BR_a^n(\delta)$: let c be in $BR_a^n(\gamma)$. By definition, $c = s^n$ for some strategy s in S_a such that $\neg(\text{seq}(\gamma; s, n) \xrightarrow{a}_{fct} \text{seq}(\gamma; s', n))$ for all $s' \in S_a$. So $\neg(\text{seq}(\delta; s, n) \xrightarrow{a}_{fct} \text{seq}(\delta; s', n))$ for all $s' \in S_a$, by contraposition of lemma 245. So c belongs to $BR_a^n(\delta)$. Therefore, the ndbr multi strategic game has an ndbr equilibrium by lemma 225. \square

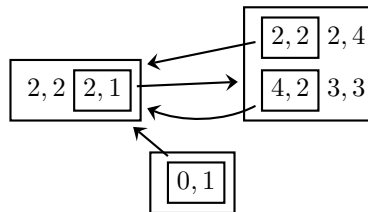
In the lemma above, the definition of BR says that at a given node and in a given context (other agents have chosen their nd strategies), an agent dismisses any of its options that induces a sequence worse than a sequence induced by some other option. Since the result above is constructive, it provides an algorithm for finding an nd equilibrium. An example is given below. The game involves two agents, namely *vertical* and *horizontal*. Agent *vertical* chooses the rows and is rewarded with the first figures given by the payoff functions.



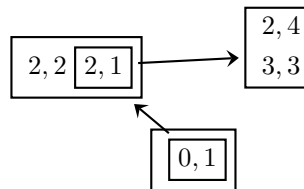
In the beginning, all agents consider all of their options. At the bottom node, only agent *horizontal* has an actual decision to take. If he chooses right, he gets an infinite sequence of 0. (*vertical* gets an infinite sequence of 3, but *horizontal* does not take it into account.) If he chooses left, he gets an infinite sequence of (non-zero) positive numbers whatever *vertical*'s strategy may be, which is better than 0 at any stage of the sequence. So *horizontal* dismisses his right strategy at the bottom node, as depicted below.



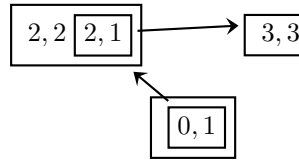
Now agent *vertical* considers the top-left node. If he chooses his bottom strategy, the induced sequence involves only 0 and 1. If he choose his top strategy, the induced sequence involves only numbers that are equal to or greater than 2, which is better. So *vertical* dismisses his top strategy at the top-left node node, as depicted below.



Now agent *horizontal* considers the top-right node. If he chooses his left strategy, the induced sequence involves only 1 and 2. If he choose his right strategy, the induced sequence involves only 3 and 4, which is better. So agent *horizontal* dismisses his left strategy at the top-right node, as depicted below.



Eventually, agent *vertical* dismisses one of his strategy at the top-right node, which yields the global nd equilibrium below. Said otherwise, for each agent, fore each node, the agent cannot get better sequence by changing his strategy.



Since sequential graph games can be embedded into multi strategic games, lemma 230 also provides a notion of global nd equilibrium for sequential graph games, with the guarantee of equilibrium existence. However, it is also possible to define more subtle notions of equilibrium with this guarantee.

9.8 Conclusion

This chapter introduces the notion of abstract strategic game, which is a natural and minimalist abstraction of traditional strategic games with real-valued pay-off functions. It also defines the notion of multi strategic game which is a generalisation of both abstract strategic games and sequential graph games. Multi strategic games can therefore model decision-making problems that are modelled by either strategic games or sequential graph/tree games. Since these new games can express both sequential and simultaneous decision-making within the same game, they can also model more complex decision-making problems. The chapter also defines non deterministic best response multi strategic games. While somewhat more abstract, they are structurally similar to multi strategic games: Cartesian product and graph-like structure. Via a prefixed point result that is also proved in the chapter, existence of ndbr equilibrium is guaranteed for ndbr multi strategic games (under some sufficient condition). Instantiating this result with (more) concrete best response functions provides different notions of non deterministic equilibrium for multi strategic games. A few examples show the effectiveness of the approach, in terms of numerical result as well as algorithmic complexity (polynomial and low). This approach is discrete and static, so it lies between Nash's probabilistic approach and the CP an BR approach.

Chapter 10

Conclusion

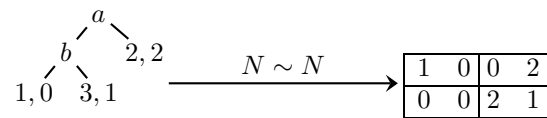
This conclusion consists of two parts. The first part recalls the main results of the thesis and adds further remarks that were not mentioned in the chapters. The second part suggests a few research directions. A one-page graphical summary is presented thereafter.

10.1 Summary and Additional Remarks

This section first recalls the main game theoretic context of the thesis. Second, it compares BR and C/P approaches. Third, it discusses (pure) equilibrium existence in sequential tree/graph games. Fourth, it mentions some computability issues that are related to Nash equilibrium. Fifth, it discusses the different approaches for compromising equilibria.

Context

This thesis is based on very few key concepts and results of traditional game theory: strategic game, Nash equilibrium, Nash's theorem, sequential game, Nash equilibrium for sequential game, subgame perfect equilibrium, and Kuhn's theorem. This context is represented by the following picture, where sequential games are embedded into strategic games to define sequential Nash equilibria.



BR versus C/P

This thesis designs two very abstract approaches of the notions of game and Nash equilibrium, namely conversion/preference and best-response approaches. These frameworks are explicitly or implicitly referred to throughout the dissertation. They are similar but one of them may suit better a given situation. For instance, the conversion/preference approach is used to study (pure)

Nash equilibria, while the BR approach is used to deal with discrete non-determinism. Indeed the BR approach is slightly more abstract, which allows "aiming at the essential". The drawback is that it gives a slightly less accurate understanding of the problem.

Pure Equilibria

Kuhn's results states Nash equilibrium existence for all real-valued sequential tree games. This does not hold in strategic games, which are simultaneous decision-making processes. Therefore, this thesis studies general sequential structures where Nash equilibrium existence is guaranteed by practical necessary and sufficient conditions on the agents' preferences. Here, practical means that the form of the condition is $\bigwedge_{a \in \mathcal{A}} P(\overset{\circ}{\rightarrow}^a)$, where P is a predicate on preferences. Indeed, if a condition mixes the preferences such that it cannot be written in the conjunctive form above, this may be unrealistic for the real world. For instance, "all the preferences coincide" is a such a condition, which yields a system with no obvious game theoretic aspect. (No conflict; no game.)

Sequential Tree Games

This thesis defines abstract sequential tree games and proves that the following five propositions are equivalent:

- Agents' preferences over the outcomes are acyclic.
- Every sequential game has a Nash equilibrium.
- Every sequential game has a subgame perfect equilibrium.
- Every one-agent sequential game has a Nash equilibrium.
- Every one-agent sequential game has a subgame perfect equilibrium.

The equivalence is formalised using Coq, which required a Coq proof of a topological sorting lemma. Note that the condition "preferences over the outcomes are acyclic" is practical/conjunctive.

Sequential Graph Games

Beside abstract sequential tree games, the thesis seeks more general structures that allow similar results. Let there be sequential graph games. On a subclass of sequential graph games, a practical/conjunctive sufficient condition and a practical/conjunctive necessary condition for global equilibrium existence are proved, and the two conditions coincide when preferences are total orders. The proof relies on results about path optimisation in graphs, and it also invokes a generalisation of "backward induction". However, on the whole class of sequential graph games, guaranteeing equilibrium existence would not be practical/conjunctive for agents' preferences. Therefore it seems that the above-mentioned subclass is a very general class of games with sequential decision-making where equilibrium existence is still guaranteed (under practical/conjunctive conditions).

Computability of Probabilistic Nash Equilibria

Consider the following class of 1-agent, 2-strategy games $g(x)$ with x in \mathbb{R} . If the agent chooses strategy h_1 , he gets payoff 0, and if he chooses strategy h_2 , he gets payoff x .

h_1	h_2
0	x

If Nash equilibria were computable in general, they would be computable for the above-mentioned subclass of games; said otherwise, there would exist a computable function f from \mathbb{R} to \mathbb{R} such that $(h_1 \mapsto f(x), h_2 \mapsto 1 - f(x))$ is a Nash equilibrium for $g(x)$ for all x . However, a computable function is continuous, as explained in Weihrauch's book [55]. Moreover, if x is negative then the only Nash equilibrium is $(h_1 \mapsto 1, h_2 \mapsto 0)$, whereas if x is positive then the only Nash equilibrium is $(h_1 \mapsto 0, h_2 \mapsto 1)$, which shows that such an f is discontinuous, hence a contradiction.

Nevertheless, the following sketches the proof of non-uniform existence of a computable Nash equilibrium. By the Tarski-Seidenberg principle (mentioned *e.g.* in [5]), every non-empty semi-algebraic set on a real closed field has an element whose components are all in the real closed field. The set of Nash equilibria of a given strategic game is semi-algebraic. The computable reals form a real closed field. Therefore, every strategic game with computable payoffs has a computable Nash equilibrium. Note that the above-mentioned continuity argument says that these computable Nash equilibria are not uniformly computable. Said otherwise, for each such strategic game, there exists a program that "converges" towards a Nash equilibrium, but there exists no program expecting such a strategic game and converging towards a Nash equilibrium.

However, if the payoff functions are restricted to rational numbers (with decidable equality), then there are uniform computability (and complexity) results in the literature about these specific Nash equilibria. Some of these results [41] are due to Papadimitriou.

Compromising Equilibria

Beside specific sequential graph games (including sequential tree games), the games defined in the thesis offer no guarantee of (pure) equilibrium existence, unless under non-practical conditions. This thesis shows that applying Nash's probabilistic approach for these abstract games is irrelevant. So it finds other means of weakening the definition of equilibrium in order to guarantee their existence, while following the features of Nash's approach to some extent.

Discrete Non-Deterministic Equilibria

The notion of discrete non-deterministic strategy is introduced: instead of saying that "an agent chooses a strategy with some probability", one says that "the agent may choose the strategy". The notion of multi-strategic games is also introduced. Such a game amounts to playing an abstract strategic game at each node of a sequential graph game. At a given node/abstract strategic game, the next move is no longer chosen by a single agent. Instead, the short-term

outcome and the next game to be played result from the play of the current abstract strategic game. This structure thus generalises both concepts of strategic game and sequential games, and it is able to express both sequential and simultaneous decision-making in a single game. This grants multi-strategic games a great modelling power, which is made effective by a result of non-deterministic equilibrium existence.

To prove the non-deterministic equilibrium existence mentioned above, this thesis defines non-deterministic best-response multi-strategic games, which are multi-strategic games where outcomes and preferences are replaced with slightly more abstract best-response functions. A pre-fixed point result helps prove ndbr-equilibrium existence for ndbr-multi-strategic games. Finally, instantiating this result in the slightly less abstract multi-strategic games proves the result.

The above-mentioned instantiation that is given in the thesis is simple; its main goal is to show that, in multi-strategic games, there exists a notion of non-deterministic equilibrium with guarantee of existence. However, it is possible and desirable to imagine more subtle instantiations that are more efficient with respect to real-world problems. For instance, the thesis discusses a few of them for the specific case of abstract strategic games.

This notion of discrete non-deterministic equilibrium is different from the notion of probabilistic Nash equilibrium in at least four respects: First, it is discrete whereas probabilistic Nash equilibria are continuous. Second, the proof of existence is constructive (provided that the preferences comply with the excluded middle principle) and an equilibrium can be actually computed (provided that the preferences are decidable), whereas probabilistic Nash equilibria are not computable, as seen above. Third, the discrete non-deterministic equilibria form a simple structure (that depends on the instantiation of ndbr-equilibrium in multi-strategic games), whereas probabilistic Nash equilibria form a complicated structure. Fourth, the equilibrium that is computed by the proof is "special" with respect to the structure of all such equilibria. Therefore, such an notion can lead to a notion of recommendation to players on how to play, whereas Nash equilibria cannot serve as a recommendation. A numerical example for strategic games shows that this recommendation is effective, compared to random play. A low algorithmic complexity makes it practical too.

As seen above, probabilistic Nash equilibria and discrete non-deterministic equilibria present different features. However, a well-chosen instantiation of the ndbr-equilibrium existence in real-valued strategic games helps establish a connection between them. More specifically, consider the nd-equilibrium that is computed by the proof of equilibrium existence instantiated with the best response BR^4 of chapter 9. In this case, every probabilistic Nash equilibrium uses only strategies that are also used by the nd-equilibrium. This suggests that discrete non-deterministic equilibria are not only a possible substitute for probabilistic Nash equilibria, but they may also help understand Nash equilibria better.

Dynamic Equilibria

There is no guarantee of Nash equilibrium existence in BR and C/P games, since they are generalisation of strategic games that already lack this prop-

erty. Unlike multi-strategic games, BR and C/P games do not enjoy a Cartesian product structure, which means that agents do not actually choose their own strategy for good. Therefore, the discrete non-deterministic approach cannot be applied. However, BR and C/P games induce finite digraphs whose strongly connected components induce directed acyclic graphs. The sinks of these dag are seen as the dynamic equilibria of the game. They are guaranteed to exist and they enjoy nice structural properties.

As shown by the thesis, there are at least three ways of compromising the definition of Nash equilibrium: the continuous and static way of Nash; a dynamic and discrete way; and in between, a discrete and static way. There does not seem to be any simple relation between the two extreme ways, namely Nash's way and the dynamic way. Nevertheless, the thesis establishes a connection between nd-equilibria and dynamic equilibria, using the instance of best response BR^3 of chapter 9. Together with the above-mentioned connection between probabilistic Nash equilibria and nd-equilibria, this last remark establishes an indirect connection between probabilistic Nash equilibria and dynamic equilibria.

10.2 Further Research

This section first suggests a few possible continuations of the thesis. In these continuations, like in the thesis, the basic notion of equilibrium is that of Nash equilibrium. Most of the work consists in abstracting, generalising, extending this notion. Second, the section mentions a few issues that suggests a need for some alternative viewpoints, where (pure) Nash equilibrium is no longer considered the only possible basic notion of equilibrium.

Continuation of the thesis

Necessary and Sufficient Condition for Equilibrium

In a subclass of sequential graph games, there is a necessary condition and a sufficient condition for equilibrium existence. These conditions do not coincide; it would be interesting to know whether or not there exists a practical/conjunctive necessary and sufficient condition on the preferences for equilibrium existence.

A Low-Level Definition of Rationality

A substantial part of the proof of the result mentioned above consists in proving equilibrium existence for 1-agent games. It suggests that equilibrium existence for 1-agent games may imply equilibrium existence for all games in the subclass. It would be interesting to question this because in many classes of games, rationality of an agent may be defined as the existence of equilibrium for every 1-agent game. This definition is low-level in the sense that it is defined using only the concepts of equilibrium: it does not refer to further high-level concepts such as logic of knowledge. Ultimately, it would be interesting to seek the "most" general classes of games where such rationality of agents guarantees the existence of an equilibrium for every game.

Non-Deterministic Equilibria

The instance of nd-equilibrium for multi-strategic games that is discussed in the thesis is likely to be improvable. It would be interesting to design an improvement and validate it through a concrete example. It would be very nice if seeking such an improvement lead to a both necessary and sufficient condition on the instance for equilibrium existence. If this last goal is too complex in multi-strategic games, one can start with abstract strategic games.

New Kinds of Equilibria

Half-Social Animal

The notion of Nash equilibrium belongs to non-cooperative game theory, which is often related to selfishness. Stronger statements are relevant: an agent that takes part in a Nash equilibrium needs only to know his payoffs in some situations of the game; he need not know his opponents payoffs at all; and he actually need not be aware of his opponents' existence at all. Conversely, if two agents are fully selfish yet fully aware of the game, a Nash equilibrium may not be an equilibrium. For instance, consider the game below. The strategy profile (v_2, h_2) is not an equilibrium since both agents will change their choice simultaneously and get payoff 2. No communication is required between the agents that are still fully selfish, so this idea of equilibrium also belongs to non-cooperative game theory.

	h_1	h_2
v_1	2 2	0 0
v_2	0 0	1 1

To make this more concrete, replace 0 with "death", 1 with "one-year imprisonment", and 2 with "one-million-euro prize". Consider two persons that are in two isolated cells. In each cell there is one button. The two persons are told that if none of them pushes his button within ten minutes, they are both sentenced to one-year imprisonment. If both push the button, they both get the prize, and otherwise they are killed. It is likely that most of the people would push the button.

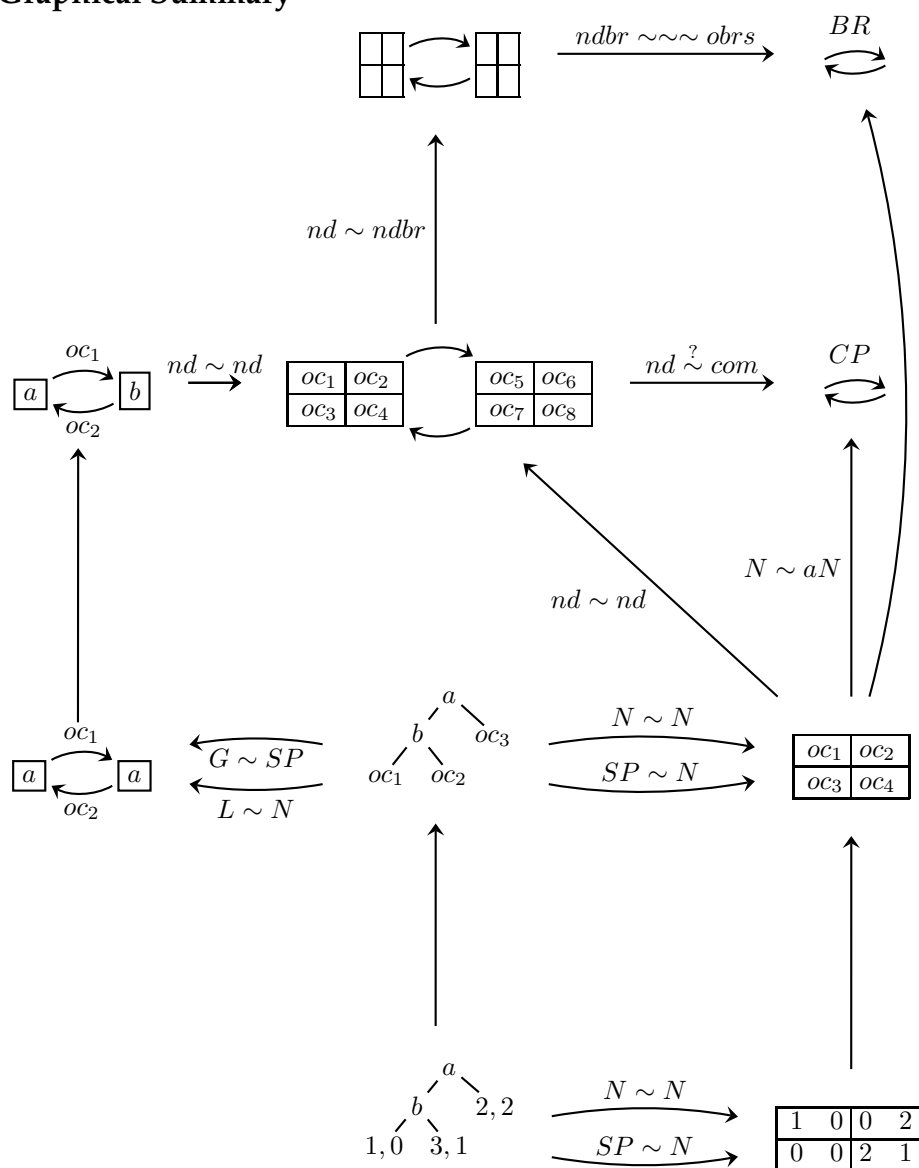
Dynamic Inspiration for Static Purpose

Postulate: there must exist a relevant notion of (pure) equilibrium with guarantee of existence in strategic games. For such a notion, all the profiles of the following game would be equilibria, by symmetry.

	h_1	h_2
v_1	1 0	0 1
v_2	0 1	1 0

For all the profiles to be equilibria, any agent that gets payoff 0 can reason as follows: "the other agent knows that I would like to change my choice to improve upon my payoff, so he may change his choice simultaneously to counter

me". Continuing this reasoning may produce concepts that are very closed to the notions of dynamic compromising equilibria that are discussed in this thesis. So, the dynamic approach for "short-sighted" agents may be useful to a static approach with agents that are aware of their opponents.



Bibliography

- [1] The Coq proof assistant, version 8.1, <http://coq.inria.fr/>.
- [2] Fabio Acerbi. Plato: Parmenides 149a7-c3. a proof by complete induction? *Archive for History of Exact Sciences*, 55(1):57–76, 2000.
- [3] Anonymous. Program evaluation research task. Summary report Phase 1 and 2, U.S. Government Printing Office, Washington, D.C., 1958.
- [4] Aristotle. Organon, Prior Analytics, 350 BC.
- [5] Saugata Basu, Richard Pollack, and Marie-Francoise Roy. *Algorithms in Real Algebraic Geometry*. Springer, 2003.
- [6] W. J. Baumol and S. M. Goldfeld. *Precursors in Mathematical Economics: An Anthology*, pages 3–9. London School of Economics and Political Science, 1968. Kuhn, H. W., Preface to Waldegrave’s Comments: Excerpt from Montmort’s Letter to Nicholas Bernoulli.
- [7] Elwyn R. Berlekamp, John Horton Conway, and Richard K. Guy. *Winning Ways for Your Mathematical Plays*. A K Peters, Ltd., 2003.
- [8] B. Douglas Bernheim. Rationalizable strategic behavior. *Econometrica*, 52:1007–1028, 1984.
- [9] Yves Berthot and Pierre Castéran. *Interactive Theorem Proving and Program Development Coq’Art: The Calculus of Inductive Constructions*. Springer, 2004.
- [10] David Blackwell. An analog of the minimax theorem for vector payoffs. *Pacific Journal of Mathematics*, 6:1–8, 1956.
- [11] F. Blanqui, S. Coupet-Grimal, W. Delobel, S. Hinderer, and A. Koprowski. CoLoR, a Coq Library on rewriting and termination. In *Workshop on Termination*, 2006. <http://color.loria.fr/>.
- [12] Emile Borel. La théorie du jeu et les equations intégrales à noyau symétrique. *Comptes Rendus Hebdomadaires des séances de l’académie des Sciences*, 173:1304–1308, 1921.
- [13] Chafika Chettaoui, Franck Delaplace, Pierre Lescanne, Mun’delanji Vestergaard, and René Vestergaard. Rewriting game theory as a foundation for state-based models of gene regulation. In *Computational Methods in Systems Biology, International Conference, CMSB 2006*, 2006.

- [14] Augustin A. Cournot. *Recherches sur les Principes Mathématiques de la Théorie des Richesses*. Hachette, Paris, 1838.
- [15] David Gale. A theory of n -person games with perfect information. *Proceedings of the National Academy of Sciences of the United States of America*, 39:496–501, 1953.
- [16] Paul Gochet and Pascal Gribomont. *Logique, méthode pour l'informatique fondamentale*. Hermes, 1991.
- [17] Timothy G. Griffin and João Luís Sobrinho. Metarouting. In *SIGCOMM '05: Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 1–12, New York, NY, USA, 2005. ACM Press.
- [18] John Hillas and Elon Kohlberg. Foundations of strategic equilibrium. In R.J. Aumann and S. Hart, editors, *Handbook of Game Theory with Economic Applications*. 2002.
- [19] Thomas Hobbes. *Leviathan, or The Matter, Forme and Power of a Common Wealth Ecclesiasticall and Civil*. Andrew Crooke, London, 1651.
- [20] Aumann R. J. and M. Maschler. Game theoretic analysis of a bankruptcy problem from the Talmud. *Journal of Economic Theory*, 36:195–213, 1985.
- [21] A. B. Kahn. Topological sorting of large networks. *Commun. ACM*, 5(11):558–562, 1962.
- [22] Shizuo Kakutani. A generalization of Brouwer's fixed point theorem. *Duke Math. J.*, 8:457–459, 1941.
- [23] Donald E. Knuth. *The Art of Computer Programming*, volume 1, second edition. Addison Wesley, 1973.
- [24] Thomas Krieger. On pareto equilibria in vector-valued extensive form games. *Mathematical Methods of Operations Research*, 58:449–458, 2003.
- [25] Harold W. Kuhn. Extensive games and the problem of information. *Contributions to the Theory of Games II*, 1953.
- [26] Daniel J. Lasser. Topological ordering of a list of randomly-numbered elements of a network. *Commun. ACM*, 4(4):167–168, 1961.
- [27] Stéphane Le Roux. Non-determinism and Nash equilibria for sequential game over partial order. In *Proceedings Computational Logic and Applications, CLA '05*. DMTCS Proceedings, 2006. <http://www.dmtcs.org/pdfpapers/dmAF0106.pdf>.
- [28] Stéphane Le Roux. Acyclicity and finite linear extendability: a formal and constructive equivalence. Research report RR2007-14, LIP, École normale supérieure de Lyon, 2007. <http://www.ens-lyon.fr/LIP/Pub/rr2007.php>.
- [29] Stéphane Le Roux. Acyclicity of preferences, Nash equilibria, and subgame perfect equilibria: a formal and constructive equivalence. Research report RR2007-18, LIP, École normale supérieure de Lyon, 2007.

- [30] Stéphane Le Roux, Pierre Lescanne, and René Vestergaard. A discrete Nash theorem with quadratic complexity and dynamic equilibria. Research report IS-RR-2006-006, JAIST, 2006.
- [31] Pierre Lescanne. Cp games: a tutorial. Technical report, LIP, 2006.
- [32] Robert Duncan Luce and Howard Raiffa. *Games and Decisions*. John Wiley and Sons, New York, 1957.
- [33] M. Manceny. *Réseaux de jeux. Une extension de la théorie des jeux pour la modélisation des interactions locales. Applications aux réseaux de régulation génétique*. PhD thesis, Université d'Évry-Val d'Essonne, 2006.
- [34] Francesco Maurolico. *Arithmeticonum libri duo*, 1575.
- [35] M.P.Jarnagin. Automatic machine methods of testing pert networks for consistency. Technical Memorandum K-24/60, U. S. Naval Weapons Laboratory, Dahlgren, Va, 1960.
- [36] John Nash. Equilibrium points in n-person games. *Proceedings of the National Academy of Sciences*, 36:48–49, 1950.
- [37] John Nash. Non-cooperative games. *Annals of Mathematics*, 54, 1951.
- [38] R.P. Nederpelt, J.H. Geuvers, and R.C. de Vrijer, editors. *Selected Papers on Automath*, volume 133 of *Studies in Logic and the Foundations of Mathematics*. North-Holland, Amsterdam, 1994.
- [39] John von Neumann and Oskar Morgenstern. *Theory of Games and Economic Behavior*. Princeton Univ. Press, Princeton, 1944.
- [40] Martin J. Osborne and Ariel Rubinstein. *A Course in Game Theory*. The MIT Press, 1994.
- [41] Christos H. Papadimitriou. On the complexity of the parity argument and other inefficient proofs of existence. *Journal of Computer and System Sciences*, 48(3):498–532, 1994.
- [42] Giuseppe Peano. *Arithmetices principia, nova methodo exposita*. 1889.
- [43] DG Pearce. Rationalizable strategic behavior and the problem of perfection. *Econometrica*, 52:1029–1050, 1984.
- [44] FO van Megen PO Borm and SO Tijs. A perfectness concept for multicriteria games. *Mathematical Methods of Operations Research (ZOR)*, 49:401–412, 1999.
- [45] Vaughan Pratt. Origins of the calculus of binary relations. In *Logic in Computer Science*, 1992.
- [46] Jean Jacques Rousseau. *Du Contrat Social ou Principes du Droit Politique*. Marc-Michel Rey, Amsterdam, 1762.
- [47] J.P. Seldin and J.R. Hindley, editors. *To H.B. Curry: Essays on Combinatory Logic, Lambda Calculus and Formalism*. Academic Press, London, 1980.

- [48] Reinhard Selten. Spieltheoretische Behandlung eines Oligopolmodells mit Nachfrageträgheit. *Zeitschrift für die gesamte Staatswissenschaft*, 121, 1965.
- [49] Reinhard Selten. Reexamination of the perfectness concept for equilibrium points in extensive games. *International Journal of Game Theory*, 4, 1975.
- [50] Herbert A. Simon. A behavioral model of rational choice. *The Quarterly Journal of Economics*, 69(1):99–118, 1955.
- [51] Edward Szpilrajn. Sur l’extension de l’ordre partiel. *Fund. Math*, 1930.
- [52] A.S. Troelstra and D. Van Dalen. *Constructivism in mathematics. An introduction.*, volume 1, chapter 6, section 1.2. North-Holland, 1988.
- [53] René Vestergaard. A constructive approach to sequential Nash equilibria. *Information Processing Letter*, 97:46–51, 2006.
- [54] John von Neumann. Zur Einführung der transfiniten Zahlen. *Acta Litterarum ac Scientiarum Regiae Universitatis Hungaricae Franciscus-Josephinae Section Scientiarum Mathematicarum*, 1:199–208, 1923.
- [55] Klaus Weihrauch. *Computable analysis: an introduction.* Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2000.
- [56] Alfred North Whitehead and Bertrand Russell. *Principia Mathematica.* Cambridge Mathematical Library, 1910.
- [57] Ernst Zermelo. Über eine Anwendung der Mengenlehre auf die Theorie des Schachspiels. *Proceedings of the Fifth International Congress of Mathematicians*, 2, 1912.