Extensive Form⁺

Describing Games Linear Nature

Diploma Thesis

submitted by

Aleksandr Mattal

supervised by

Professor Dr. Thomas Wilke

Research Group Theoretical Computer Science
Department of Computer Science
Christian-Albrechts-Universität zu Kiel

August 2011

Abstract

In this work we restrict ourselves to finite dynamic non-cooperative games satisfying perfect recall. At first, a broadly recognized form of describing such games is described, namely Extensive Form. After that, the motivation behind the idea of extending that form is presented and a formal (and informal) definition of the extended Extensive Form (which we call Extensive Form$^+$) is given.

Finally, a notion of strategy (Partial Assessment with Pure Allocation, or shortly PAPA) and a solution concept based on that notion (Partial Sequential Equilibrium, or shortly PSE) for Extensive Form$^+$ are proposed. As a main result, we prove that every Extensive Form$^+$ games has at least one PSE as a solution.

Acknowledgements

I would really like to thank my thesis supervisor, Professor Dr. Thomas Wilke, for giving me the chance to write a thesis on my own topic of choice, for his continuous support and guidance. I want to thank Mihhail Aizatulin for the inspiration, support and many thought-provoking conversations. Also, I would like to thank the former Professor of Mathematics Vladimir Pyasetski for his support, advice and remarkable recommendations of literature related with the subject of this work.

Moreover, I want to thank Professor Peter Bro Miltersen, Troels Bjerre Sørensen, Jens Burmeister and Professor Dr. Thomas Slawig for helpful discussions.

And last but not least, sincere thanks to my family and friends, who have always been there for me.

Contents

1

Introduction

Life is full of situations where success of a participant depends on its choices and choices of other participants. Game Theory provides mathematical foundation for answering questions about such situations (in terms of Game Theory called games).

Probably the main question Game Theory tries to answer is how to play games 'rationally' or how to 'solve' these games. 'Playing rationally' or 'solving' games can be described as finding a way to play the game in a way that satisfies certain desired conditions, for example, so that a player could guarantee itself some expected payoff at the end of the game.

Solving games can be divided into three major stages: first, we describe a game. There are several known ways to describe games, one of the most important known to date being Extensive Form which is often being used to describe dynamic[1] games. As a second step we define what a strategy for the chosen form is. Strategy defines how should participants behave when playing the game. Finally, in order to define what 'playing rationally' means, we choose something what is called a solution concept. It describes a way how different strategies could be compared with each other, how one strategy can be said being 'better' than another one (or even all of them) and which of them is (are) considered to be a 'solution' of the game. There are many solution concepts proposed, most notably Nash Equilibrium and the whole family of Equilibrium concepts based on it.

In this work we will concentrate our view on finite[2] dynamic non-cooperative games[3] satisfying perfect recall[4] described in Extensive Form. We propose a new form of describing games that we call Extensive Form$^+$ which essentially is an extension of Extensive Form. After showing how Extensive Form games are solved, we propose a notion of strategy and a solution concept for games described using Extensive Form$^+$. The main result is that there always exists at least one such solution for any finite Extensive Form$^+$ game satisfying perfect recall (in the form these games are introduced in this thesis).

[1] Games where some actions are made after others.

[2] Games with finite structure (more formally in the next chapter).

[3] Games where each participant acts independently and in its own interests.

[4] Games where no participant can forget what actions it made during the play (more formally in the next chapter).

From the beginnings of Game Theory which take start from works of Borel (Applications aux Jeux de Hasard, 1938) and von Neumann together with Morgenstern (Theory of Games and Economic Behavior, 1944, (16)), this field has flourished from trying to solve simply formulated games. One of such games, still remaining unsolved, is a game of Poker. Our proposal is mostly motivated by this game (to be more correct, by its variation called No-Limit Texas Hold'em) and a beautiful publication (6) on this game showing that during betting rounds in a Heads Up Tournament setting[1] with small stacks[2] of the game it suffices to use only two actions, ALL-IN and FOLD, to play near-optimally. However, if we don't restrict a player to use only these two actions and try to describe the original structure of betting rounds in Poker using Extensive Form, it results in an enormous size of the game's and its strategy's description and, consequently, in more resource demanding calculations when trying to find a solution for the game.

Our proposal of Extensive Form$^+$ is aimed to give us the ability to describe structures similar to the structure of betting rounds in Poker, in a more natural way. Even though the form, as it is proposed in this work, still doesn't provide any advantage in formulating such complex games as Poker, we show that it can already be used to describe simpler games with similar structure in a more adequate and, for some of them, much more compact way than if they were described using Extensive Form.

As compared with the original version of the thesis that was submitted in partial fulfillment of the requirements for my degree, this version contains minor changes.[3]

[1] A setting where there are only two players who recursively play the same game until one of them has money to play with.

[2] A situation when all players have a small amount of money (relative to Big Blind which is a default bet in the game) to bet with.

[3] Several small typing mistakes are corrected; a footnote on p.12 is added; in chapter eight (Future Research) there is one paragraph that is fully excluded from the thesis.

1.1 Basic Notation and Terminology

In this thesis we refer to a Player in the game as to it/him if it has no name and him/her dependent on a name if he has one.

$\mathcal{N}_0$ denotes the set of natural numbers including 0,
$\mathcal{N}$ denotes the set of natural numbers excluding 0,
$\mathcal{Z}$ denotes a set of integer numbers,
$\mathcal{R}$ denotes a set of real numbers.

Sets are typically denoted by capital letters, elements by lower-case letters, functions by calligraphic letters.

Operations

There are several operations we often use throughout this thesis:

$\times$	denotes Cartesian product,
$\Delta(X)$	denotes a set of all possible probability distributions over elements of set X,
$X\backslash Y$	denotes a set that contains all elements $x \in X$ such that $x \notin Y$,
$Span(X)$	denotes a set of linear combinations of the elements of some set X over $\mathcal{R}$.
$\dot{\cup}$	denotes a union of sets that don't contain common elements.
$\cdot$	denotes multiplication on $\mathcal{R}$. For $a, b \in \mathcal{R}$ we write $a \cdot b$ instead of $\cdot(a, b)$,

Graphs

A finite directed graph is a pair (V, E) where V is a finite set of vertices (or nodes) and $E \subseteq V \times V$ is a set of edges. A path is a finite sequence of nodes $v^0 v^1 ... v^l$ such that $\forall_{1 \le i \le l} : (v_{i-1}, v_i) \in E$. With $(v, v') - path$ we denote a set of all nodes in a path that starts in v and stops in v'.

A finite directed tree is a finite directed graph (V, E) where exists exactly one node v_0 such that for every $v \in (V\backslash\{v_0\})$ exists exactly one path that starts in v_0 and ends in v. We call v_0 the root of the tree.

$V^+ : V \to 2^V, v \mapsto \{v' | v' \in V, (v, v') \in E\}$ is a function returning all successors for a given node.

$E^+ : V \to 2^E, v \mapsto \{(v, v') | v' \in V, (v, v') \in E\}$ is a function returning all outgoing edges for a particular node.

V^v denotes all such nodes $v' \in V$ (including v) that there exists a path starting in v and ending in v'. E^v denotes a set of all edges that belong to these paths. (V^v, E^v) (or, alternatively, $(V, E)^v$) denotes a subtree structure with a root v.

Theorems

When proving a theorem/lemma/corrolary/remark, we mostly use following two symbols:

$\lightning$ denotes contradiction symbol.
$\blacksquare$ denotes the end of the proof (QED symbol).

2

Defining Extensive Form$^+$

In this chapter we first define Extensive Form and show a couple of examples of how certain game structures are formulated in Extensive Form. Then we mention a certain structure that cannot be adequately described in Extensive Form and propose a definition of the extended Extensive Form (further called Extensive Form$^+$) which fixes that problem.

2.1 Extensive Form

One of the most accepted ways to formulate dynamic games, first introduced by John von Neumann and Oskar Morgenstern in 1944 (16) and then extended by Harold Kuhn in 1953 (5), is called Extensive Form. Usual presentation of games in Extensive Form is quite elegant. The game is represented by a directed tree, where all the possible plays of the game are depicted by all the paths from the root of the tree (the beginning of the play) to the leafs (all the possible endings of the play). Each node is called a state in the game or a decision node and every one of them, except leafs, belongs to one of the players or the nature. Each outgoing edge depicts an action available to the owner of the node. All leafs have no owners and are called terminal nodes. Each terminal node is assigned a *numerical* payoff value for each player.

Example Game: Homework Assignment

As an example we will show a simple game (Figure 1) with following rules:

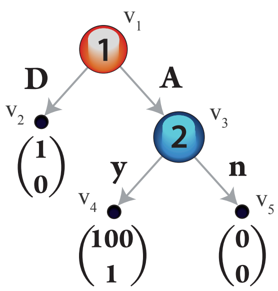

Figure 1: Homework Assignment.

Player 1 (let's name him Bob) has a home assignment in philosophy. Bob is bad at philosophy and even if he would sit all night doing his homework, he would probably get a bad grade for it. However, Bob has a friend named Amy (Player 2) who goes to the same class as Bob. So Bob has a choice to make: either he will do (action D) the homework himself or he could go to school next morning and ask (action A) Amy to do his homework in exchange for, say, one apple. If he asks Amy, she may either say 'yes' (action y) in which case Bob's grade is guaranteed to be good and Amy gets an apple[1]. Or she could say 'no' (action n) - then Bob gets 0 points for his assignment and keeps his apple.

[1] Utility functions defined on terminal nodes of this game show how many apples will Amy get and what the expected grade for Bob's homework will be if the node is reached. Upper number in brackets belongs to Player 1

2.1.1 Formulating Uncertainty

Uncertainty within a game can be formulated with nature nodes (nodes belonging to nature). All the outgoing edges of a nature node are then assigned probability values, which define with what probability will the following state of the game (node) take place. Naturally, the sum of all the probabilities assigned to the outgoing edges of such node should be 1. Nature can be viewed as yet another player who doesn't compete with anybody and has a predetermined strategy, represented by probability distribution among the outgoing edges, which is known to all players in the game.

Example Game: Guess the Coin Flip

This example shows a simple game (Figure 2)[1] where nature is involved. The game is described as follows: Bob (Player 1) goes to school and is offered to play a following game: Someone flips a perfect coin (a coin which lands on either side with probability 0.5) and Bob tries to guess which side is it going to land on. If Bob guesses correctly he wins one Euro. If not - he loses one. Moreover, the game happens only if Bob accepts the offer (action A). If he refuses (action R) then nothing happens (payoff 0).

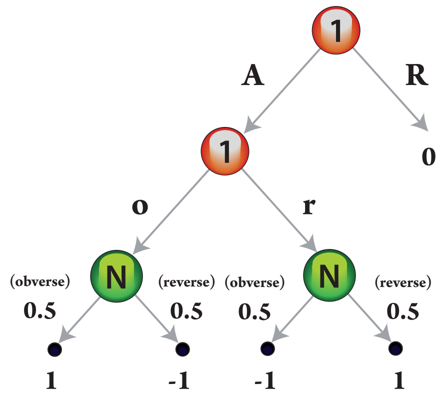

Figure 2: Guess the Coin Flip.

2.1.2 Perfect and Imperfect Information

It could also be the case when a player at some state has no clue, in which state exactly it is at the moment. For example, this could happen if players act simultaniously (at the same time), if some players cannot observe certain moves (that is, some actions are made in private) or if some of the players forgets some of the actions it made during the play.

For such cases, while constructing the game, we divide the set of nodes into independent sets called information sets. They can contain one or more nodes. Sets consisting of only one node are called *trivial* or *singleton* information sets. Each *nontrivial* (consisting of more than 1 node) information set can contain only nodes belonging to the same player. Such nontrivial sets mean that when a player reaches one of its nodes, he doesn't know in which node of that information set he is really at. Naturally, each of these nodes has the same set of actions available (reasonable, because if you would have different sets of actions in two nodes belonging to the same information set, then according to what actions are available to you, you would be able to decide, which of these two states in the information set you are at. This would mean that you have a clue about where you are in that information set, which contradicts the definition of an information set). Nature is not involved in nontrivial information sets, because nature always has a clue which state it is at.

Games where (*not*) all the information sets are trivial, are called games with (*im*)perfect information.

[1] The reader can see that not all nodes are named as it was done on Figure 1. Through this work, whenever possible, we will try to omit denoting such game structure as node or edge names on figures. It is made with intention to make visual examples intuitive and easy to understand.

Example Game: Rock, Paper, Scissors

In this example we show a simple game of imperfect information (Figure 3):

Bob (Player 1) and his younger brother Edward (Player 2) found a coin. After a short discussion they agree to play a following game to decide who will get the coin: Player 1 and Player 2 have three alternatives to choose from: 'Rock', 'Paper' and 'Scissors'. They choose one of them at the same time. 'Rock' is said to beat 'Scissors', 'Scissors' beat 'Paper', 'Paper' beats 'Rock'. If the player's choice beats the choice of his opponent, the player wins (payoff 1) the game and gets the coin. In case of loss the payoff is −1 and if both players choose the same thing, it's considered a tie (payoff 0) and no one gets the coin. Because both players choose at the same time, the game involves imperfect information.

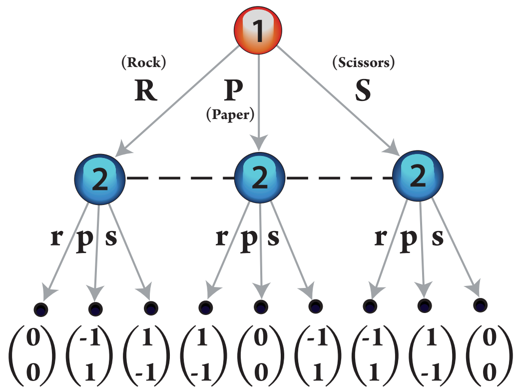

Figure 3: Rock, Paper, Scissors.

2.1.3 Formal Definition

Definition 1. *A tuple $G = (P, A, V, E, I, \lambda, \sigma_N, (\tau_p)_{p\in P})$ is a finite Extensive Form n-player game where:*

- *P is a finite nonempty set of players with $|P| = n$.*
- *Let $p \in P$ be some player. Then A_p is a set of actions available to that player and $A := \dot\cup_{p\in P} A_p$ is a set of actions, available to all players.*
- *(V, E) is a nonempty finite directed tree and $v_0 \in V$ be its root node. Any path from the root node to a leaf is called a* play*. $V_T \subseteq V$ is a set of leafs of (V, E).*

  Let V_p define the set of nodes belonging to each player $p \in P$ and V_N define the nodes belonging to nature, so that $V = V_T \dot\cup V_N \dot\cup (\dot\cup_{p\in P} V_p)$. Similarly, let $E_p := \dot\cup_{v\in V_p} E^+(v)$ define edges belonging to player p and $E_N := \dot\cup_{v\in V_N} E^+(v)$ - the edges of nature.

  Let's assume that all players play, that is $\forall p \in P : V_p \neq \varnothing$.
- *For each player $p \in P$, I_p is a partition of V_p, dividing nodes into* information sets *(in such a way that if two nodes end up in the same information set, the player cannot say in which node of this set he is exactly in the game). $I_N = V_N$ is a set of nature's information sets. All information sets of nature are* trivial *(singleton), because nature always knows where it is at in the game. $I := (\dot\cup_{p\in P}(I_p)) \dot\cup I_N$ is a set of all information sets of the game. I is then a partition of nonterminal nodes $V \backslash V_T$.*

- $\lambda : (E \backslash E_N) \to A$ *is an* action assignment *function, assigning every* player-controlled *edge an action (that is, nature isn't assigned any actions). The function* λ *has to fulfill following five constraints*[1]*:*

  ⋆ *First, for simplicity reasons, let's assume all actions contained in* A *are used:*

  $$\forall a \in A \exists e \in E : \lambda(e) = a.$$

  ◦ *Now let* $e = (v, v') \in E$ *and* $v \in s \in I_p$ *for some player* $p \in P$. *Then* $\lambda(e) = a$ *implies:*

    ⋆ *a belongs to player p*

    $$a \in A_p,$$

    ⋆ *that player i would be able to play a from every node of the information set s*

    $$\forall v'' \in s \exists v''' \in V : \lambda((v'', v''')) = a,$$

    ⋆ *that a would be available only once in v*

    $$\forall e' \in (E^+(v) \backslash e) : \lambda(e') \neq a,$$

  (⋆.5) *and that a wouldn't be available outside the information set s*

  $$\forall v'' \in (V \backslash s) \forall e' \in E^+(v'') : \lambda(e') \neq a.$$

- $\sigma_N : E_N \to (0, 1]$ *is the 'strategy of nature', a function which distributes probabilities among nature actions. It represents uncertainty in the game.* σ_N *can be seen as a predetermined strategy of nature*[2]. *That is also the reason why we don't need* nontrivial *information sets for nature nodes.* σ_N *has to satisfy only one constraint:*

  $$\forall v \in V_N : \sum_{e \in E^+(v)} \sigma_N(e) = 1.$$

- $(\tau_p : V_T \to \Re)_{p \in P}$ *is a collection of payoff functions, one for each player. They define the numerical outcomes for each of the player for the case the game ended in the particular terminal node.*

Let Γ^{EF} be the set of all finite games that can be described in Extensive Form using this definition.

Remark 1. *The finiteness of* G *follows from finiteness of* V.

Further in this work we will also use a notion of a subgame. A subgame G^v beginning in a node $v \in V$ (called the initial node of the subgame) is a part of the game structure, containing all the game information contained in the subtree $(V, E)^v$. It has to satisfy following two conditions:

- The initial node v should belong to a singleton information set.
- If $v' \in V^v$ and $v' \in s$ for some $s \in I$ then $\forall v'' \in s : v'' \in V^v$. In other words, no information set that has a node in V^v is allowed to have nodes outside the subtree $(V, E)^v$.

[1] It should be mentioned that through these constraints, we automatically satisfy the condition of having the same set of actions available for the nodes belonging to the same information set.

[2] σ_N is a function taken from Harsanyi transformation introduced for Extensive Form games in (4).

2.2 Motivation for Extensive Form$^+$

While being both simple and sufficient enough to cover a huge amount of games we are interested in, Extensive Form may be a little too impractical for our needs. At the end of the day, we are mostly interested in solving those games. There is an enormous amount of literature on complexity of solving games - and for the most part, games tend to be hard to solve (Finding even 2-player Nash Equilibrium is $PPAD$-Complete). Only some small class of games have been proven to have a good general solution. These results show that in order to get some improvement in solving 'hard' games, we should exploit their structure (this thought can be well supported with a quote of Papadimitrou: "The PPAD-completeness of Nash suggests that any approach to finding Nash equilibria that aspires to be efficient [...] should explicitly take advantage of computationally beneficial special properties of the game in hand" (from (11) p.49)).

2.2.1 Lack of Structure in Extensive Form

So what we truly care about, is the structure of the game. This includes payoffs at the end of the game and how they are formed. However, seeing only *numerical* payoffs at the leafs of the game tree, as it is defined in Extensive Form, makes it difficult to see if there is any exploitable structure behind them.

Imagine we describe somekind of a 'real' game in Extensive Form. If, by any chance, some action in 'real' game forms final payoffs (for example, one of the players commits some of his resources to the game, so that these resources could later be won by some player) - then we wouldn't see it in Extensive Form. All we can see in such a representation - is the resulting payoff and not what formed it.

2.2.2 Real Life

In real life, many problems are formulated by a set of rules. And in many cases, this natural description of the game is much more compact than its equal description in Extensive Form game. Moreover, it is often the case that if we change a certain parameter in the natural description, the length of the description (almost) doesn't change. But if we look at the description of the same problem in Extensive Form, this 'small change' of natural description may result in a huge change in the Extensive Form description (example: try to formulate 3 rounds of betting in No-Limit Texas Holdem Poker where players have stacks of 5 Big Blinds as a tree, in Extensive Form - and then try to do the same thing for a situation when players have 1000 Big Blinds each. While the natural description remains almost as long as it was before, the size of its description in Extensive Form blows up extremely). Thus we may need a more adequate way to describe games.

Rules often describe not only the order in which players have their turns or when some information is revealed (whether privately or publicly) - but also the structure of how the payoffs are formed (example: the same 3 betting rounds in Poker or commiting resources by firms in economic scenarios).

2.3 Extensive Form$^+$

While we can (theoretically) describe how *each action alone* forms a certain part of the payoff in Extensive Form, we cannot describe the connection between the structure behind these actions and final payoffs (example: in Poker, betting amount in the first round *depends* on the *total count of chips* player has in the beginning, betting amount in the second round *depends* on the *betting amount* of both players in the first round and so on... and in the end the final payoff for the winner is *the sum of all commitments* he has made along the play). By adding the possibility to describe these connections between actions and how they affect final payoffs, we allow many games to be represented in a more compact and structured manner. We call that form the Extensive Form$^+$.

2.3.1 Informal Definition

A good way to imagine a game in Extensive Form$^+$ is to compare this representation to the original Extensive Form:

An Extensive Form game is just played according to the tree structure and at the end you just get some payoff. There is no description of how the payoffs are formed - they are just there from the beginning, at the leafs of the tree.

A game in Extensive Form$^+$ goes as follows: the play goes as in the Extensive Form, from the root to the bottom of the game tree. During the play, players and nature can commit certain amount of (their (existing)) resources, these may as well be 0 or negative values. These commitments are restricted by linear constraints, assigned to all actions. Constraints should restrict possible commitments from both sides (that is, possible values of the commitment must form a closed interval on Z). These constraints are described as a linear combination of all the commitments, done during actions preceding the current action during the play. The number of constraints is not limited by one constraint from each side - there may as well be more (example: $x \in \mathcal{Z}, x \geq y + z, x \geq 3, x \leq 18 + 4z$, where y and z are some commitments done in the play before the commitment x is made). Informally, if one doesn't write anything on the edge e, the corresponding edge's commitment can be assigned only 0 as a value.. An edge where its commitment can take only a single value is called a *trivial action*. Edges with commitments that can take a value out of a set of more than 1 value are called *nontrivial action*s.

The final payoff of the play is defined as a linear combination of all commitments during the play.

Because of the definition of the payoff function and constraints of the commitments for every action, we can bind actions with ones that took place earlier in the play (and, for example, we can formulate initial resources available to both players by binding the variable of the first action with some number).

IMPORTANT: As it is still unclear how to deal with information sets containing nodes of nontrivial actions's edges, in this work we restrict our view on games, where nodes belonging to nontrivial actions belong to singleton information sets and the game starting after a nontrivial action is a subgame (which also means that these actions are public - that is, other players observe the size of commitment made by a player during a nontrivial action).

2.3.2 Formal Definition

Let $P, A, V, E, I, \lambda, \sigma_N$ be defined exactly as in the definition of Extensive Form. Before we can formally define Extensive Form$^+$, we will have to define constraint structure $(C_a)_{a\in A}$, differently defined payoff functions $(\tau_p)_{p\in P}$ and two more constraints on I:

- First, let's deal with the constraint structure. Before we do it, we will have to define following three functions[1]:

  * $Path : V \to 2^A, v \mapsto \{\lambda(e) | e = (v', v'') \in (E \setminus E_N), v', v'' \in (v_0, v) - path\}$ be a function returning all actions taken on a path from the root v_0 to selected node v.
  * $Path_N : V \to 2^E, v \mapsto \{e | e = (v', v'') \in E_N, v', v'' \in (v_0, v) - path\}$ be a function returning all edges that belong to nature on a path from the root v_0 to v.
  * $Prec : A \to 2^A, a \mapsto \bigcap_{v\in V, \exists e\in E^+(v):\lambda(e)=a} Path(v)$ be a set of actions that 'preceded' a, no matter which edge e with $\lambda(e) = a$ we've chosen.

  Now let $a \in A$ be some action. The constraint structure $C_a \in 2^{Span(Prec(a)\cup\{1\})}$ is a set of linear combinations of actions, preceding a (1 is added to that actions list, because, for example, the first action taken in the game cannot depend on preceding actions and, therefore, should be constrained numerically). For $c \in C_a$ we write $c = (\sum_{a'\in A} c_{a'} a') + c_{rest}$, where, according to definition, $a' \notin Prec(a) \Rightarrow c_{a'} = 0$. With C_a we can describe linear constraints that we set on the value that will be assigned to the action. $(C_a)_{a\in A}$ describes all such constraints in the game. Now we define a function called *allocation* that assigns actions their values:

  Definition 2. *A function $t : A \to Z$ is called allocation if it satisfies all the constraints described in $(C_a)_{a\in A}$:*

  $$\forall a \in A \forall c \in C_a, c = (\sum_{a'\in A} c_{a'} a') + c_{rest} : (\sum_{a'\in A} c_{a'} t(a')) + c_{rest} \leq 0.$$

  Let T be the set of all possible allocations for some game $G \in \Gamma^{EF^+}$.

  Now that we have defined allocation, it can be shown, how the constraints are described. Say, we want to assign action a constraints $a \geq 3, a \geq a' + 2, a \leq 6 - a''$ where $a', a'' \in A$ are some actions that 'preceded' a in the game. Then $C_a = \{-a + 3, -a + a' + 2, a + a'' - 6\}$.

  Constraint structure $(C_a)_{a\in A}$ has to satisfy following four constraints:

  * Let $h_a = |C_a|$. The first constraint we lay on C_a is that all actions should be constrained:
  $$\forall a \in A : h_a \neq 0,$$
  * Also, a has to be included itself in every constraint $c = (\sum_{a'\in A} c_{a'} a') + c_{rest}$ of C_a:
  $$\forall c \in C_a : c_a \neq 0,$$
  * There have to exist at least one allocation, because it is needed to define payoffs later:
  $$|T| \geq 1,$$

[1] We should shortly note that we will quite often use these three functions in this thesis

* And because we describe finite games, there have to be only finitely many allocations:

$$|T| \neq \infty.$$

We divide A into 2 disjunct subsets $A^{trivial}$ and $A^{nontrivial}$, labeling each action as trivial or nontrivial. An action a is called *nontrivial* if and only if there can appear a situation in the game, where a player can choose one of multiple values to assign to a:

($\diamond$.0)

$$a \in A^{nontrivial} \Leftrightarrow \exists t, t' \in T : (\forall a' \in Prec(a) : t(a') = t'(a')) \wedge (t(a) \neq t'(a))$$

Otherwise, a is called *trivial*. As a consequence, all actions constrained to have a single numerical value from the beginning (for example, restricted by 0 from both sides) are trivial. A game containing only trivial actions is also called *trivial game*, as well as a game containing at least one nontrivial action is called *nontrivial*.

- Instead of numerical values, each terminal node $v \in V_T$ is assigned, similarly to actions, a certain linear combination of all preceding actions (that is, all actions that lay on the v_0, v-path). $(\tau_p)_{p \in P} : V_T \to Span(A \cup \{1\})$ with $\forall p \in P \forall v \in V_T : (\tau_p)_{p \in P}(v) \in Span(Path(v) \cup \{1\})$.

  Given an allocation t we can define a payoff for each player in that node. Payoff for player $p \in P$ in node v with $\tau_p(v) = (\sum_{a' \in A} c_{a'} a') + c_{rest}$, given allocation t, is $\tau_p^t(v) = (\sum_{a' \in A} c_{a'} t(a')) + c_{rest}$.

- The third difference between Extensive Form and Extensive Form$^+$ is rather a restriction we lay on games in this work (because we still don't know what to do when including nodes belonging to a nontrivial action, into nontrivial information sets). Let $v, v' \in V, e = (v, v') \in E, a = \lambda(e) \in A^{nontrivial}$. Then it must hold:

($\diamond$.1) $\{v\} \in I$,

($\diamond$.2) $G^{v'}$ is a subgame (G denotes constructed Extensive Form+ game).

As an implication, this constraint is explicit for trivial games as they don't contain non-trivial actions.

Definition 3. *A tuple $G = (P, A, V, E, I, \lambda, \sigma_N, (C_a)_{a \in A}, (\tau_p)_{p \in P})$ is a finite Extensive Form$^+$ game.*

Let Γ^{EF^+} be the set of all games that can be described in Extensive Form$^+$ using this definition.

Remark 2. *The finiteness of G follows from finiteness of V and T.*

2.3.3 Perfect and Imperfect Recall

A game G where a player never forgets what he has done in the past is called a game satisfying perfect recall. In other words, if, for any two nodes in the same information set belonging to some player p, the actions of this player made before reaching any of these nodes, are the same.

Definition 4. *G satisfies perfect recall if it holds:*

$$\forall p \in P \forall s \in I_p \forall v, v' \in s : (Path(v) \cap A_p) = (Path(v') \cap A_p).$$

Definition holds for both Extensive Form and Extensive Form$^+$ games.

Example Game: Dice Game

In this example (Figure 4) we present the game[1] in Extensive Form+ that we will then later use in the following chapters, to illustrate how our proposals of the new form work and what kind of advantage do they give us. In illustrating Extensive Form+ games we will omit constraints of actions that are constrained by 0 and if the action is constrained only by one value from both sides ($a \le 3$ and $a \ge 3$) we will simply write $a = 3$. The rules of this game are as follows:

Bob (Player 1) plays a dice game with his friends: when it's Bob's turn to play, he begins (action B) the game with a certain amount of money (only one Euro coins allowed). Let's say Bob has three Euros ($B = 3$). First, Bob bets any amount of money he wants (of course, no more money than he brought with himself and not less than 0). Then a dice is thrown and it is memorized whether an odd (O) or even (E) number came out. Then Bob bets again - this time no more than he already bet in the first round. Then again a dice is thrown. Depending on the sequence ((OO), (OE), (EO) or (EE)) the dice showed in these two rounds, Bob gets his money back plus a certain profit denoted by payoff (if payoff is negative, he has to pay his friends). Let b_1 denote the bet in the first round, b_2 in the second. Then the payoffs are: for (OO) it is $(-b_1 - b_2)$, for (OE) it's $(-b_1 + 3b_2)$, for (EO) it's $(2b_1 - b_2 + 1)$ and for (EE) it's $(-2b_1 + 2b_2 - 1)$.

Formal Definition:

$G = (P, A, V, E, I, \lambda, \sigma_N, (C_a)_{a \in A}, (\tau_p)_{p \in P})$

$P = \{1\}$,

$A_1 = \{B, B_1, B_2', B_2''\}$,
$A = A_1$,

$V_1 = \{v_0, v_1, v_3, v_4\}$,
$V_N = \{v_2, v_5, v_6\}$,
$V_T = \{v_7, v_8, v_9, v_{10}\}$,
$V = V_1 \dot\cup V_N \dot\cup V_T$,

$e_0 = (v_0, v_1)$, $e_1 = (v_1, v_2)$,
$e_2 = (v_2, v_3)$, $e_3 = (v_2, v_4)$,
$e_4 = (v_3, v_5)$, $e_5 = (v_4, v_6)$,
$e_6 = (v_5, v_7)$, $e_7 = (v_5, v_8)$,
$e_8 = (v_6, v_9)$, $e_9 = (v_6, v_{10})$,

$E_1 = \{e_0, e_1, e_4, e_5\}$,
$E_N = \{e_2, e_3, e_6, e_7, e_8, e_9\}$,
$E = E_1 \dot\cup E_N$,

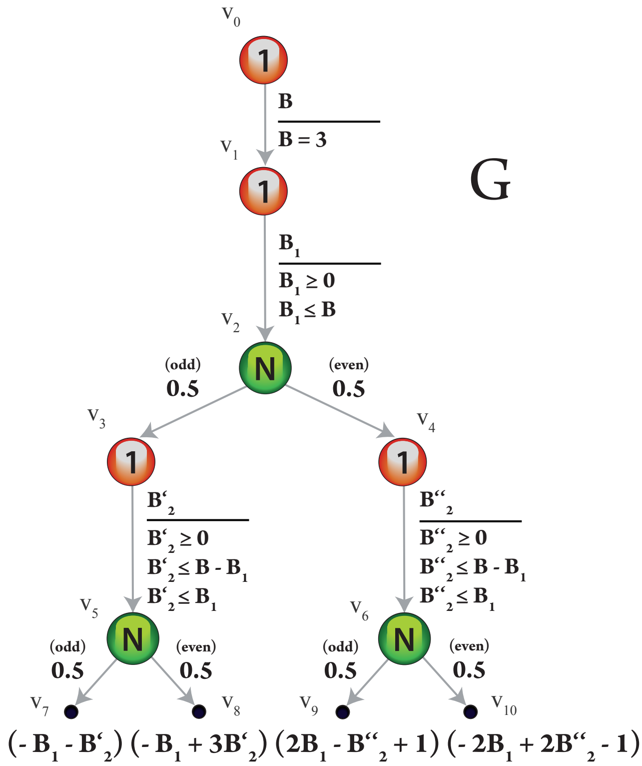

Figure 4: Dice Game.

$\lambda : e_0 \mapsto B$
$\quad e_1 \mapsto B_1$
$\quad e_4 \mapsto B_2'$
$\quad e_5 \mapsto B_2''$,

$\forall e \in E_N : \sigma_N = 0.5$,

$C_B = \{-B + 3, B - 3\}$,
$C_{B_1} = \{-B_1, B_1 - B\}$,
$C_{B_2'} = \{-B_2', B_2' + B_1 - B\}$,
$C_{B_2''} = \{-B_2'', B_2'' + B_1 - B\}$,

$\tau_1 : v_7 \mapsto -B_1 - B_2'$
$\quad v_8 \mapsto -B_1 + 3B_2'$
$\quad v_9 \mapsto 2B_1 - B_2'' + 1$
$\quad v_{10} \mapsto -2B_1 + 2B_2'' - 1$.

[1]This game has a structure very similar to that in the game of Craps. Craps is a well-known casino game, often played in an informal setting because all that players need to play it are three to six dice (depending on the game variation played).

3

Notion of Strategy - Extensive Form

It's nice to be able to formulate games in a seemingly adequate manner. But how should we play such games? How do we describe what kind of actions should be taken in a certain situation? For that we need to define what a plan, or strategy, means. In this chapter we present several definitions of strategy that already exist for Extensive Form games.

3.1 Strategy

Strategy is a plan for the whole[1] game. That means, a *strategy for player i* should explain, what should be done in each of player i information sets. Imagine we have defined a strategy for each player. A combination of these strategies defines a strategy for the whole game and is called a *strategy profile.*

We define strategies similar to how they are defined in the book 'Game Theory: Analysis of Conflict' (8):

3.1.1 Pure Strategy Profile

Let $G = (P, A, V, E, I, \lambda, \sigma_N, (\tau_p)_{p \in P}) \in \Gamma^{EF}$. Let $p \in P$, $s \in I_p$. With $A(s) = \{\lambda(e) | e \in E^+(v)\}$ we denote the set of actions available for player p in s. Pure strategy defines which single action should p choose from $A(s)$ for each $s \in I_p$. So the strategy space of pure strategies for player p is $\times_{s \in I_p} A(s)$.

Definition 5. *$\sigma_p \in \times_{s \in I_p} A(s)$ is a pure strategy for a player $p \in P$ in G.*

Definition 6. *$\sigma \in \times_{p \in P} \times_{s \in I_p} A(s)$ is a pure strategy profile for G.*

We also write $\sigma = (\sigma_p)_{p \in P} = (\sigma_{p,s})_{p \in P, s \in I_p}$, where σ_p means a pure strategy for player p and $\sigma_{p,s}$ is an action that is chosen by player p in the information set s.

[1] That doesn't mean it should only explain what should be done in reachable game states (for example, let v be a state of the game belonging to player 1, $E^+(v) = \{e_1, e_2\}$, $a_1 = \lambda(e_1)$, $a_2 = \lambda(e_2)$ and player 1 commits to playing some strategy that says to always use a_1 in v. One may think that because of player 1 commitment to the strategy, he never chooses to play a_2 - so all the game states following e_2 are unreachable and, therefore, it wouldn't be neccessary to define a strategy for those states). Strategy should be defined for all states, even if the state is unreachable in the sence of this example.

Let $a \in A$ be some action. Let $p \in P, s \in I_p$ so that $a \in A(s)$. For simplicity reasons, we will write $\sigma_{p,s}(a) = 1$ if $a = \sigma_{p,s}$ and $\sigma_{p,s}(a) = 0$ if $a \neq \sigma_{p,s}$, which describes probability that player p will take an action a in s.

In order to solve games using strategies (that is, to say which strategies are the 'best' ones to use), we will have to compare them. In order to be able to do that, we define the *value of the game* for a certain strategy profile under the condition that players commit to playing it. First, we define a *utility function* $u^\sigma_{p \in P} : V \to \mathfrak{R}$, a function that recursively defines value for each node of the game according to a strategy profile σ:

$$\forall p \in P \forall v \in V_T : u^\sigma_p(v) = \tau_p(v),$$
$$\forall p \in P \forall s \in I_p \forall v \in s : u^\sigma_p(v) = \sum_{e=(v,v') \in E^+(v)} (\sigma_{p,s}(\lambda(e)) \cdot u^\sigma_p(v')),$$
$$\forall p \in P \forall v \in V_N : u^\sigma_p(v) = \sum_{e=(v,v') \in E^+(v)} (\sigma_N(e) \cdot u^\sigma_p(v')).$$

We call $u^\sigma_p(v_0)$ the value of the game for player p for strategy profile σ. There is also another way to calculate the value of the game:

$$u^\sigma_p(v_0) = \sum_{v \in V_T} ((\Pi_{a \in Path(v)} \sigma(a))(\Pi_{e \in Path_N(v)} \sigma_N(e)) \tau_p(v)).$$

Both ways are equivalent in the sense that they give us the same value for each σ we choose. In the following chapters we will use both definitions.

3.1.2 Mixed Strategy Profile

Pure strategies allow no randomness to be present in the strategies. For example, in the game 'Rock, Paper, Scissors', whatever pure strategy you commit to, your opponent can easily choose a strategy which allows him to win every time you play (which seems to be disappointing, especially considering the fact that the rules of the game are equal for both players). So it seems that if we want to somehow play the game 'optimally', pure strategies are not a plausible choice for strategies to choose from. Thus, we need a richer choice of strategies. Instead of choosing one single strategy, let's choose one of the pure strategies with certain probability in the beginning of the game.

Definition 7. *$\sigma \in \times_{p \in P} \Delta(\times_{s \in I_p} A(s))$ is a mixed strategy profile for G.*

Mixed strategies seem to present a much better (richer) choice of strategies for solving games but they are a bit too complicated: how do you write simply a mixed strategy that consists of, say, ten pure strategies that have non-zero probabilities? And what if the strategy assigns all pure strategies non-zero probabilities? And how do you answer the question 'what should I do in that state of the game?'? It all seems to be rather complicated[1] to use (even though there are several important results proven for mixed strategies). Therefore, we need a more adequate definition of strategy.

[1]Originally, mixed strategies were used for games written in a form of a matrix (games described in Normal Form or Strategic Form). And in those games it seemed to be quite comfortable to prove results with mixed strategies. For Extensive Form, however, it is not the case.

3.1.3 Behavioral Strategy Profile

In pure strategies, we chose to play a single action with probability 1 in each of player's information sets. Behavioral strategy, instead of assigning the probability 1 to some action, distributes probabilities among all actions used in the information set. So the strategy space of behavioral strategies for player $p \in P$ would be $\times_{s \in I_p} \Delta(A(s))$.

Definition 8. *$\sigma \in \times_{p \in P} \times_{s \in I_p} \Delta(A(s))$ is a behavioral strategy profile for G.*

We will also write $(\sigma_p)_{p \in P}$ or $(\sigma_{p,s})_{p \in P, s \in I_p}$, where σ_p means a behavioral strategy of a player p, $\sigma_{p,s}$ means a strategy for information set $s \in I_p$ and $\sigma_{p,s}(a)$ (we will sometimes write simply $\sigma(a)$) means a probability assigned to action $a \in A(s)$.

Value of the game for behavioral strategies is defined exactly as it is defined for pure strategies. We call $u_p^\sigma(v_0)$ the value of the game for player p for strategy profile σ. We will later also need an additional utility function $u_{p \in P, s \in I_p}^\sigma : s \times A(s) \to \mathfrak{R}$, a function that says what is the expected payoff for a player p if he plays an action $a \in A(s)$ in some node $v \in s$ in one of its information sets $s \in I_p$. It is defined as follows:

Let $p \in P$, $s \in I_p$, $v \in s$, $a \in A(s)$ and $v' \in V^+(v)$ such that $\lambda((v, v')) = a$.

$$u_{p,s}^\sigma(v, a) := u_p^\sigma(v').$$

Also we will need a notion of completely mixed behavioral strategy (profile):

Definition 9. *A behavioral strategy σ_p is completely mixed if $\forall a \in A_p : \sigma(a) > 0$.*

Definition 10. *A behavioral strategy profile σ is completely mixed if $\forall a \in A : \sigma(a) > 0$.*

Behavioral strategies describe behavior of players in each of their information sets, regardless what moves were taken before the player reached it. Such strategies are much easier to write down and much simpler than mixed strategies, in the sense that a behavioral strategy can be drawn on the game tree, just like a pure strategy[1]. It turns out, which is a remarkable result (5), that for the case of finite Extensive Form games satisfying perfect recall, behavioral strategies and mixed strategies are equivalent in the sense that for each mixed strategy profile there is an equivalent behavioral strategy profile with the same game value and vice versa.

[1] We will show it on examples in the next Chapter.

4

Solving Games - Extensive Form

Having a way to describe games (Extensive Form) and a definition of strategy for games, we can now begin to answer the question 'how should such games be played?'. In order to do that, we have to compare strategies and choose 'better' ones. A concept explaining how and in what sense are some strategies better than other ones, is called a solution concept.

In this chapter, we will describe, probably, the most accepted solution concept to this day, namely Nash Equilibrium (*further NE*). Then we will go through a couple of arguments against using Nash Equilibrium in the form it was originally described by John Nash in (9) and (10) and present two of it's refinements, Subgame Perfect Nash Equilibrium (*further SPNE*) and Sequential Equilibrium (*further SE*), that solve some of the issues with Nash Equilibrium. The last concept, Sequential Equilibrium, will be later used in the proposal for solving a subclass of EF^{+} games.

All three Solution Concepts will be defined similarly to how they are defined in the book 'Game Theory: A Multi-Leveled Approach' (14)

4.1 Nash Equilibrium

Let $G \in \Gamma^{EF}$, Θ be the strategy space for G and $\sigma = (\sigma_p)_{p \in P} \in \Theta$ be some strategy profile for that game (we intentionally don't define what kind of strategy is used). Let p be some player in G, Θ_p be the strategy space for that player and $u_p^{\sigma^*}$ be a utility function that gives out the game value for player p if played by strategy σ^*. Now we will write (σ_p, σ_{-p}) instead of σ, where σ_p is a strategy of a player p and σ_{-p} denotes strategies of all the other players in σ. If there exists no such other strategy $\sigma'_p \in \Theta_p$ for player p that $(\sigma'_p, \sigma_{-p}) \in \Theta$ would result in a bigger value of the game for player p than the value of σ then σ_p is called to be the best reply to σ_{-p}.

Definition 11. *σ_p is the best reply to σ_{-p} if $\forall \sigma'_p \in \Theta_p : u_p^{(\sigma_p,\sigma_{-p})} \geq u_p^{(\sigma'_p,\sigma_{-p})}$.*

The best reply seems to be a good choice of a strategy if you know the strategies other players have chosen. But what would happen if one of these 'other' players would decide to change its strategy? That could, of course, change your value of the game dramatically. What John Forbes Nash, Jr. has proposed (9) and (10) in 1950 was a concept of an Equilibrium (nowadays called Nash Equilibrium), a strategy profile where a strategy of each player is the best reply to other players strategies.

Definition 12. *$(\sigma_p)_{p \in P}$ is Nash Equilibrium if $\forall p \in P$ holds: σ_p is the best reply to σ_{-p}.*

We noted already that Nash Equilibrium does not define what kind of strategies should be used. It just defines a criteria for a strategy profile to be considered a Nash Equilibrium. A good question is, of course, if such solution exists for a chosen strategy space Θ in some game G. A beautiful result, proven by Nash ((9), (10)), is that a Nash Equilibrium in mixed strategies exists for every finite game. Consequently, as there exists an 'equal' behavioral strategy profile for every mixed strategy profile in finite Extensive Form games, there exists a Nash Equilibrium in behavioral strategies for every finite Extensive Form game (5).

A good way to think of Nash Equilibrium is as of a good choice for a minimal condition we demand from a strategy to call it a solution. Because all that NE demands from a strategy profile, is that no *single*[1] rational participant would have an intention to change its strategy.

Example

Now let's take a game of 'Rock, Paper, Scissors' from Figure 3 and take a careful look at it. Take a look at Figure 5[2]. On the left side is depicted a pure strategy profile $\sigma_1 = \{R, p\}$ and on the right side is a behavioral strategy σ_2 with $\forall a \in \{R, P, S, r, p, s\} : \sigma_2(a) = \frac{1}{3}$.

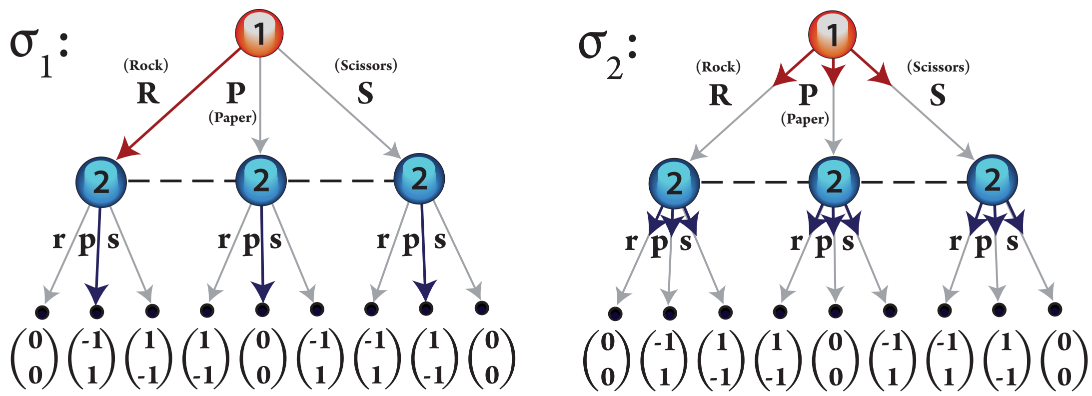

Figure 5: Example of two different strategies for Rock, Paper, Scissors.

σ_1 is not[3] a NE because its value for Player 1 is -1 and if Player 1 would change his strategy to S then his game value would increase to 1.

σ_2 is a NE with a game value 0 for both players. It means that if Player 1 holds to this strategy (which is, essentially, to choose 'Paper', 'Rock' or 'Scissors' randomly[4]) then Player 2 couldn't change his strategy to increase his expected game value.

[1]That means that there may exist such a Nash Equilibrium where, if two players change their strategy at the same time, they both might get a better game value from it. As an example, take a look on Figure 6. Both strategies are considered to be a Nash Equilibrium - but if both players would change their strategy in the NE on the right side to their strategies in NE on the left side, both would increase their game value.

[2]Length of a colored bold line in relation to the length of the edge its on, depicts the probability assigned to the action assigned to that edge.

[3]This example is mainly to show how can a pure strategy profile be depicted in visual form. 'Rock, Paper, Scissors' doesn't have NE in pure strategies at all (because whatever strategy you choose, one of the players could change its strategy to increase his game value).

[4]It means that if both players would play this game rationally (that is, according to this strategy, randomly), it would be a game of pure luck. Ironically, there exist world championships for 'Rock, Paper, Scissors' and even books written on that game (we leave this statement without reference).

4.1.1 Non-credible Threats

Despite being widely accepted as a good solution concept, Nash Equilibrium has several major issues. First issue is that for many games there are too many[1] Nash Equilibria in the game and some of them might not be as good as the other ones. For example, an NE might possess somewhat called *non-credible threat.*

A non-credible threat is player's action in the strategy profile which that player, if he is rational, would not play (because it would then lessen that (and, probably, other) player's expected payoff in the game following that action). In order to better understand that issue, let's take a look at a simple example:

Example

As an example, we take again the Home Assignment game from Figure 1. It has four possible pure strategy profiles, two of which are NE. Both of them are depicted on Figure 6 and one of them has a non-credible threat in it:

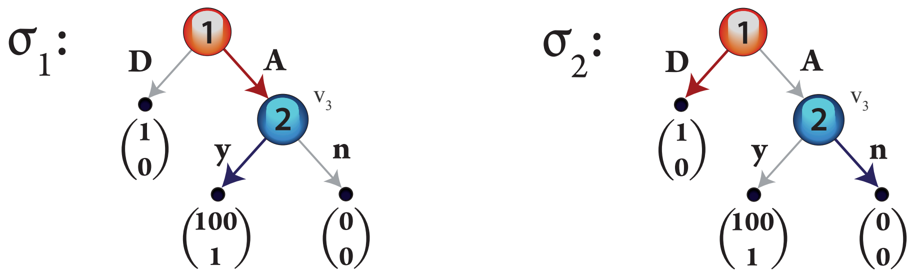

Figure 6: Example of a non-credible threat.

While σ_1 seems to be a reasonable strategy, σ_2 doesn't because it contains a non-credible threat at v_3: Player 2 (Amy) threats Player 1 (Bob) to refuse doing homework for Bob which makes it a better choice for Bob to do his assignment himself. On the other side, if Bob makes his homework himself, Amy wouldn't have an intention to choose y in v_3 as it wouldn't change her outcome (she gets no apple either way). Threat is called non-credible because if v_3 would actually be reached during the play, choosing n wouldn't be rational. A careful reader might see that a strategy σ_2 in the subgame G^{v_3} is not a Nash Equilibrium in that case. That thought leads us to the first refinement of NE presented in this chapter: Subgame Perfect Nash Equilibrium.

[1]For example, in the game on Figure 2, all pure/mixed/behavioral strategy profiles are NE. But what makes it interesting is that the concept of Nash Equilibrium is ignorant to the choice Bob makes. It is indifferent to the choice between risking or not risking to lose money. It is a property all the Equilibrium concepts we describe here have: they are all based on calculating expected payoff. And expectation is blind to the amount of risk it holds.

4.2 Subgame Perfect Nash Equilibrium

In order to solve the issue with non-credible threats, we will demand that a strategy profile would also be a Nash Equilibrium for each subgame. Obviously, such strategy profile is always a Nash Equilibrium because the game is also a subgame of itself.

Definition 13. *A strategy profile σ is Subgame Perfect Nash Equilibrium if it is a Nash Equilibrium for each subgame of G.*

As an example, on Figure 6, σ_1 is SPNE[1] while σ_2 isn't. SPNE works great with games with Perfect Information as it sorts all NE containing non-credible threats out. But SPNE works on subgames and cannot dissect nontrivial information sets structure. The idea of SPNE is especially useless in games where the only subgame is the game itself (as Poker, for example) because in this case each NE is automatically SPNE and that means that if there are too many NE in such a game to choose from (and it is more often so than not so), SPNE doesn't refine our choice at all. All these objections lead us to another refinement of Nash Equilibrium, namely Sequential Equilibrium.

4.3 Sequential Equilibrium

For this section, let $G = (P, A, V, E, I, \lambda, \sigma_N, (\tau_p)_{p\in P}) \in \Gamma^{EF}$.

Apart from previous two concepts we have described, this one uses only behavioral strategies. Before we define it, we will jump into a couple of simple definitions and then explain what we need them for:

Definition 14. $\beta \in \times_{s\in I}\Delta(s)$ *is called a belief system.*

A belief system is simply a probability distribution over nodes in every information set. Now we combine a behavioral strategy with a belief system and define this combination as an assessment:

Definition 15. $(\sigma, \beta) \in (\times_{p\in P} \times_{s\in I_p} \Delta(A(s))) \times (\times_{s\in I}\Delta(s))$ *is called an assessment.*

It doesn't make much sense to have a combination of some behavioral strategy and some belief system that aren't connected to eachother in some particular way. In order to connect them we will denote with $\mathcal{P}_\sigma(v) = (\Pi_{a\in Path(v)}\sigma(a))(\Pi_{e\in Path_N(v)}\sigma_N(e))$ the probability[2] that v will be reached if played σ. Also, let for each $s \in I$ denote with $\mathcal{P}_\sigma(s) = \sum_{v\in s}\mathcal{P}_\sigma(v)$ the probability that the information set s will be reached if played σ.

Definition 16. *Assessment (σ, β) is called Bayesian consistent if*

$$\forall s \in I : \mathcal{P}_\sigma(s) > 0 \Rightarrow (\forall v \in s : \beta(v) = \frac{\mathcal{P}_\sigma(v)}{\mathcal{P}_\sigma(s)}).$$

In that way beliefs of players are determined by σ on those information sets that are reached with nonzero probability. That means that if σ is completely mixed, β can be directly reconstructed from it[3], which leads us to a stronger consistency criterium:

[1] Unique for this game, whether using pure or behavioral strategies.

[2] Note that we have already seen that structure in Chapter 3 when we defined the value of the game.

[3] Nature's strategy doesn't interfere in this case because we have define it on an open interval: $\sigma_N : E_N \to (0, 1]$.

Definition 17. *Assessment* (σ, β) *is called consistent if there exists a sequence* $(\sigma^k, \beta^k)_{k\in\mathcal{N}}$ *of assessments that satisfy:*

- $\forall k \in \mathcal{N} : \sigma^k$ *is completely mixed,*
- $\forall k \in \mathcal{N} : (\sigma^k, \beta^k)$ *is Bayesian consistent,*
- $lim_{k\to\infty}(\sigma^k, \beta^k) = (\sigma, \beta)$.

Because of having the belief system, we could define a criterium saying that if a player assigns any action a positive probability with σ, then it should be an action guaranteeing the best possible expected payoff in its information set:

Definition 18. *Assessment* (σ, β) *is sequentially rational if for every* $p \in P, s \in I_p, a \in A(s)$ *it holds that if* $\sigma(a) > 0$ *then:*

$$\sum_{v\in s}(\beta(v)u^{\sigma}_{p,s}(v,a)) = max_{a'\in A(s)}\sum_{v\in s}(\beta(v)u^{\sigma}_{p,s}(v,a')).$$

Definition 19. *Assessment* (σ, β) *is a Sequential Equilibrium if it is consistent and sequentially rational.*

Now that we have quickly defined what a Sequential Equilibrium is, we have to note that we will neither say how to find it (in most cases it is not so easy) or explain it in detail. To prove the main result later in this thesis, we will only use the definition of it and the following theorem:

Theorem 1 (Kreps, Wilson '82)**.** *Every finite Extensive Form game satisfying perfect recall has a Sequential Equilibrium.*

And to convince the reader that Sequential Equilibrium is the best solution concept (for the games we describe) among all three in this Chapter, we remark the following result:

Theorem 2. *Every Sequential Equilibrium is subgame perfect and, in particular, a Nash Equilibrium.*

The proof of this theorem can be found in ((13), Lemma 4.2.6).

5

Transforming Extensive Form$^+$ games into Extensive Form

We have already said in the Chapter 2 that games with a certain structure could be formulated in a much more adequate way using Extensive Form$^+$. In this chapter we will formally define, how to convert any $G \in \Gamma^{EF^+}$ into its 'equal' game $G' \in \Gamma^{EF}$.

The reader might ask, 'Why should we reduce G to some Extensive Form game?' - in that way we will be able to use the knowledge collected about Extensive Form games, in Extensive Form$^+$ games. In that way we can also show the actual advantage the Extensive Form$^+$ gives us (see examples in the end of the Chapter: Figures 7.1, 7.2, 7.3).

5.1 Transforming Trivial Games

Let $G = (P, A, V, E, I, \lambda, \sigma_N, (C_a)_{a\in A}, (\tau_p)_{p\in P}) \in \Gamma^{EF^+}$ be a finite Extensive Form$^+$ game satisfying perfect recall and $A^{nontrivial} = \varnothing$.

First, let's prove there is only one unique allocation for such a game.

Remark 3. $|T| = 1$

Proof. Assume there exist two different allocations $t, t' \in T, t \neq t'$ for G. Then there exists a play $\nu = (v_0, ..., v_g)$ in G with actions $a \in Path(v_g)$ assigned along it such that there exists an action a_i among these actions, with $t(a_i) \neq t'(a_i)$. Let a_i be chosen in such a way that $\forall a \in Prec(a_i) : t(a) = t'(a)$. Then there exist two different allocations such that $\forall a \in Prec(a_i) : t(a) = t'(a)$ and $t(a_i) \neq t'(a_i)$. From $(\diamond.0)$ in Chapter 2 (Definition of Extensive Form$^+$) follows that $a_i \in A^{nontrivial}$. Therefore, $A^{nontrivial} \neq \varnothing$. ↯ ■

So let t be that single allocation for G. That means that the payoffs are uniquely defined for each terminal node. Let $v \in V_T$, $p \in P$ and $\tau_p(v) = (\sum_{a'\in A} c_{a'}a') + c_{rest}$. Then the unique payoff for p in v is defined as $\tau'_p(v) := \tau^t_p(v) = (\sum_{a'\in A} c_{a'}t(a')) + c_{rest}$.

As we have the payoffs uniquely defined, we don't need constraints or linear payoff functions anymore. If we replace $(\tau_p)_{p\in P}$ with $(\tau'_p)_{p\in P}$ and erase constraint structures, we will get an Extensive Form game, that essentially describes the same game as G does.

Definition 20. *$G' = (P, A, V, E, I, \lambda, \sigma_N, (\tau'_p)_{p\in P}) \in \Gamma^{EF}$ is an 'equal' representation of G in Extensive Form.*

Let η^Γ denote the transformation that transforms any trivial game $G \in \Gamma^{EF^+}$ to its 'equal' representation $G' \in \Gamma^{EF}$. For simplicity reasons, we will write G^{η^Γ} instead of G'.

Lemma 1. *G^{η^Γ} is an Extensive Form game that:*

- *is finite,*
- *satisfies perfect recall.*

Proof. V is finite so from Remark 1 follows that G^{η^Γ} is finite. And as V, E, I, A, λ remain the same as in G, G^{η^Γ} also satisfies perfect recall. ∎

5.2 Transforming Nontrivial Games

Constructing an 'equal' Extensive Form game for a nontrivial game is a bit trickier to do. In order to do it, we will first reduce it to a trivial EF$^+$ game by recursively 'unrolling'[1] nontrivial actions one-by-one and then apply η^Γ:

For this section let $G = (P, A, V, E, I, \lambda, \sigma_N, (C_a)_{a\in A}, (\tau_p)_{p\in P}) \in \Gamma^{EF^+}$ be a finite Extensive Form$^+$ game satisfying perfect recall, $a' \in A_{p'}$ a nontrivial action belonging to some player $p' \in P$, $a' \in A^{nontrivial}$, $v_1, v_2 \in V$ such that $e' = (v_1, v_2) \in E$ and $\lambda(e') = a'$. Moreover, we will need following structures, which copy a subgame in G^{v_2} (note that v_1 belongs to a singleton information set (follows from ($\diamond$.1) in Chapter 2, definition of Extensive Form$^+$) - so e' is the only edge that is assigned action a' (follows from ($*$.5) in Chapter 2, definition of Extensive Form) in G and so there is only one subtree that follows an action a'):

$A^v := \{\lambda(e) | e \in (E^v \backslash E_N)\}$ is the set of actions used in $(V, E)^v$.

- V^* is a set of new nodes (that aren't included in V) such that there exists a bijection $\delta_V : V^{v_2} \to V^*$. Define
  - $\forall p \in P : V^*_p := \{\delta_V(v) | v \in (V^{v_2} \cap V_p)\} \subseteq V^*$,
  - $V^*_N := \{\delta_V(v) | v \in (V^{v_2} \cap V_N)\} \subseteq V^*$,
  - $V^*_T := \{\delta_V(v) | v \in (V^{v_2} \cap V_T)\} \subseteq V^*$.
- $E^* := \{(\delta_V(v), \delta_V(v')) | v, v' \in V^{v_2}, (v, v') \in E\}$ is a set of new edges (that aren't included in E) that copies the edges from E^{v_2} and connects nodes in V^* in the exact same way. $\delta_E : E^{v_2} \to E^*, (v, v') \mapsto (\delta_V(v), \delta_V(v'))$ is also a bijection (because δ_V is). Define
  - $\forall p \in P : E^*_p := \{\delta_E(e) | e \in (E^{v_2} \cap E_p)\} \subseteq E^*$,
  - $E^*_N := \{\delta_E(e) | e \in (E^{v_2} \cap E_N)\} \subseteq E^*$,

[1] Even though the formal notation of 'unrolling' may seem somewhat cumbersome (which is due to relatively long formal description of both Extensive Form and Extensive Form$^+$) - what it actually does can be easily grasped when first looking on a drawn example of how an example game is 'unrolled'. So we advise the reader to look at the Figures 7.1, 7.2, 7.3 at the end of this section while reading formal definitions in this section.

- A^* is a set of new actions (that aren't included in A) such that there exists a bijection $\delta_A : A^{v_2} \to A^*$. This set copies actions used in $(V, E)^{v_2}$. Define
  - $\forall p \in P : A^*_p := \{\delta_A(a) | a \in (A^{v_2} \cap A_p)\} \subseteq A^*$,
- I^* is a set of new information sets (that aren't included in I) that copies the structure of information sets in $(V, E)^{v_2}$. We can copy their structure because of the constraint we laid on EF$^+$ games (follows from ($\diamond$.2) in Chapter 2) which guarantees that no information set of a subtree following a nontrivial action would intersect with information sets outside the subtree. Let $s \in I$ be some information set with $s \subseteq V^{v_2}$. With $s^* := \{\delta_V(v) | v \in s\}$ we denote a copy of that information set. We define $\forall p \in P : I^*_p = \{s^* | s \in I_p, s \subseteq V^{v_2}\}$, $I^*_N = \{s^* | s \in I_N, s \subseteq V^{v_2}\}$ and $I^* = \{s^* | s \in I, s \subseteq V^{v_2}\}$.
- $(C^*_a)_{a \in A^*}$ are constraint structures for new actions. They are constructed as follows: first, let's define a bijection $\delta_C : 2^{Span(A \cup \{1\})} \to 2^{Span((A \setminus A^{v_2} \cup A^*) \cup \{1\})}, c = (\sum_{a \in A} c_a a) + c_{rest} = (\sum_{a \in (A \setminus A^{v_2})} c_a a) + (\sum_{a \in A^{v_2}} c_a a) + c_{rest} \mapsto (\sum_{a \in (A \setminus A^{v_2})} c_a a) + (\sum_{a \in A^{v_2}} c_a \delta_A(a)) + c_{rest}$. In that way each constraint that was laid on some action $a \in A^{v_2}$ will be copied for $\delta_A(a)$ with all the actions from A^{v_2} changed to their copies in A^*.

  Let $a \in A^*$. Then $C^*_a := \{\delta_C(c) | c \in C_{\delta_A^{-1}(a)}\}$.
- $(\tau^*_p)_{p \in P}$ are the payoff functions that are defined for new terminal nodes in V^*. Let $p \in P$. Then $\tau^*_p : \{\delta_V(v) | v \in (V^{v_2} \cap V_T)\} \to 2^{Span((A \setminus A^{v_2} \cup A^*) \cup \{1\})}, v \mapsto \delta_C(\tau_p(\delta_V^{-1}(v)))$.

Using these definitions we define two transformations, *add* and *delete*. After we do it, we will, similar to Lemma 1 prove that these operations preserve properties of finiteness and of perfect recall (in that way, we can later run a sequence of such operations on G and be sure that the resulting game will still satisfy perfect recall and be finite like the original game G).

Let $z \in \mathcal{Z}$ be some integer.

Definition 21. *Operation add(G, a, z) outputs an Extensive Form$^+$ game* $G' = (P, A', V', E', I', \lambda', \sigma_N, (C'_a)_{a \in A'}, (\tau'_p)_{p \in P})$ *where*

- *The player set remains to be the same both in G and G'.*
- *Actions A^{v_2} are copied and then added to A. Finally another new action that we will call a^z will be assigned to the edge that connects the copied subtree with G:*

$$A' = A \cup (A^* \cup \{a^z\}),$$

$$\forall p \in (P \setminus \{p'\}) : A'_p = A_p \cup A^*_p,$$

$$A'_{p'} = A_{p'} \cup A^*_{p'} \cup \{a^z\}.$$

- *Then we add the copied nodes:*

$$V' = V \cup V^*,$$

$$\forall p \in P : V'_p = V_p \cup V^*_p,$$

$$V'_N = V_N \cup V^*_N,$$

$$V'_T = V_T \cup V^*_T.$$

- *The same thing is done with edges, except that an additional edge* $e^z = (v_1, \delta_V(v_2))$ *is added (it connects* (V,E) *with* (V^*,E^*)*):*

$$E' = E \cup (E^* \cup \{e^z\}),$$

$$E'_{p'} = E_{p'} \cup E^*_{p'} \cup \{e_z\},$$
$$\forall p \in (P \backslash \{p'\}) : E'_p = E_p \cup E^*_p,$$
$$E'_N = E_N \cup E^*_N.$$

- *The structure of information sets is also copied:*

$$I' = I \cup I^*,$$

$$\forall p \in P : I'_p = I_p \cup I^*_p,$$
$$I'_N = I_N \cup I^*_N.$$

- λ *is extended to* $\lambda' : (E' \backslash E'_N) \to A'$ *so that:*

$$\forall e \in E \backslash E_N : \lambda'(e) = \lambda(e),$$

$$\forall (v,v') \in E^* \backslash E^*_N : \lambda'((v,v')) = \delta_A((\delta_V^{-1}(v), \delta_V^{-1}(v'))),$$
$$\lambda'(e^z) = a^z.$$

- σ_N *is extended to* $\sigma'_N : E'_N \to (0,1]$ *so that:*

$$\forall e \in E_N : \sigma'_N(e) = \sigma_N(e),$$

$$\forall (v,v') \in E^*_N : \sigma'_N((v,v')) = \sigma_N((\delta_V^{-1}(v), \delta_V^{-1}(v'))).$$

- *Constraints are updated in such a way that new constraint structures satisfy all the properties of constraints of Extensive Form*$^+$ *games. Note that action* a^z *is constrained to a single value of* z *which makes this action trivial:*

$$\forall a \in A : C'_a = C_a,$$

$$\forall a \in A^* : C'_a = C^*_a,$$
$$C'_{a^z} = \{a^z - z, -a^z + z\}.$$

- *Finally, payoff functions are extended:*

$$\forall p \in P \forall v \in V_T : \tau'_p(v) = \tau_p(v),$$

$$\forall p \in P \forall v \in V^*_T : \tau'_p(v) = \tau^*_p(v).$$

Definition 22. *Operation delete*(G,a) *outputs an Extensive Form*$^+$ *game* $G' = (P, A', V', E', I', \lambda', \sigma'_N, (C'_a)_{a \in A'}, (\tau'_p)_{p \in P})$ *where*

- *The player set remains to be the same both in* G *and* G'*.*

- *Actions used in* $(V,E)^{v_2}$ *are deleted:*

$$A' = A \backslash (A^{v_2} \cup \{a'\}),$$

$$A'_{p'} = A_{p'} \backslash (A^{v_2} \cup \{a'\}),$$
$$\forall p \in (P \backslash \{p'\}) : A'_p = A_p \backslash A^{v_2}.$$

- *Then we delete the nodes contained in* V^{v_2}*:*

$$V' = V \backslash V^{v_2},$$

$$\forall p \in P : V'_p = V_p \backslash V^{v_2},$$
$$V'_N = V_N \backslash V^{v_2},$$
$$V'_T = V_T \backslash V^{v_2}.$$

- *Edges contained in* E^{v_2} *are deleted too:*

$$E' = E \backslash (E^{v_2} \cup \{e'\}),$$

$$E'_{p'} = E_{p'} \backslash (E^{v_2} \cup \{e'\}),$$
$$\forall p \in (P \backslash \{p'\}) : E'_p = E_p \backslash E^{v_2},$$
$$E'_N = E_N \backslash E^{v_2}.$$

- *Information sets in* $(V,E)^{v_2}$ *are deleted:*

$$I' = I \backslash \{s \in I | s \subseteq V^{v_2}\},$$

$$\forall p \in P : I'_p = I_p \backslash \{s \in I_p | s \subseteq V^{v_2}\},$$
$$I'_N = I_N \backslash \{s \in I_N | s \subseteq V^{v_2}\}.$$

- λ *is reduced to* $\lambda' : (E' \backslash E'_N) \to A'$ *so that:*

$$\forall e \in (E' \backslash E'_N) : \lambda'(e) = \lambda(e).$$

- σ_N *is reduced to* $\sigma'_N : E'_N \to (0,1]$*:*

$$\forall e \in E'_N : \sigma'_N(e) = \sigma_N(e).$$

- *Constraints from* G^{v_2} *are deleted - in that way new constraint structures satisfy all the properties of constraints of Extensive Form*$^+$ *games:*

$$\forall a \in A' : C'_a = C_a.$$

- *Payoff functions are also corrected:*

$$\forall p \in P \forall v \in V'_T : \tau'_p(v) = \tau_p(v).$$

Lemma 2. *Both $add(G, a, z)$ and $delete(G, a)$ are finite Extensive Form$^+$ games satisfying perfect recall.*

Proof. As both operations only finitely extend/shorten V and T, from Remark 2 directly follows the finiteness of both games.

$delete(G, a)$ satisfies perfect recall because *delete* only deletes structure in G.

Now in $add(G, a, z)$ all game structure from G is preserved so for any information set in G the property of perfect recall preserves. The only place where it can be broken is in I^*. Because all information sets in I^* lay in a subgame, for any $v \in s$ for some $s \in I_p^*$ holds:

$$Path(\delta_V(v_2)) \subseteq Path(v). \quad (\diamond)$$

Assume there exist $v, v' \in s : (Path(v) \cap A'_p) \neq (Path(v') \cap A'_p)$. Let's say there exists $a_0 \in (Path(v) \cap A'_p)$ such that $a_0 \notin (Path(v') \cap A'_p)$. From $(\diamond)$ follows that $a_0 \in A^*_p$. From definition of I^*_p follows that $\exists s' \in I : \delta_V^{-1}(v), \delta_V^{-1}(v') \in s'$ and from definition of V^*, E^*, λ' follows that

$$\delta_A^{-1}(a_0) \in (Path(\delta_V^{-1}(v)) \cap A_p),$$
$$\delta_A^{-1}(a_0) \notin (Path(\delta_V^{-1}(v')) \cap A_p).$$

From both equations follows:

$$\exists s'' \in I_p \exists v'', v''' \in s'' : (Path(v'') \cap A_p) \neq (Path(v''') \cap A_p).$$

Thus, G doesn't satisfy perfect recall. ↯ ■

5.2.1 Unrolling a Single Nontrivial Action

Let $a \in A^{nontrivial}$ be such that $\forall a' \in Prec(a) : a' \in A^{trivial}$. That means we can replace all $a' \in Prec(a)$ in C_a through unique value they can be assigned. It means that a is numerically restricted and we can define exact set of value it could be assigned. Let $z_1, ..., z_k \in \mathcal{Z}$ be all the values that could be assigned to a. Then a Single Unroll function η_a^Γ is defined so that given the G and a it outputs G' (we will also write $G^{\eta_a^\Gamma}$ instead of G'):

Definition 23. *Function η_a^Γ is defined as follows:*

η_a^Γ : *for* ($i = 1$ *to* k) {
    $G = add(G, a, z_i)$;
}
$G = delete(G, a)$;
return G;

5.2.2 Unrolling a Nontrivial Game

Define $find$ to be a function that takes G as input and outputs first nontrivial action it finds (using either depth-first search or breadth-first search) in G. It will then be such an action a that $\forall a' \in Prec(a) : a' \in A^{trivial}$. Then we'll use a η_a^Γ on G - and we will do the same thing recursively, until all nontrivial actions are unrolled. In the end, we will get a trivial EF$^+$ game, which we transform to Extensive Form using η^Γ and output it as a result.

Definition 24. *The output of function Φ is called an equivalent Extensive Form representation of G (also called G^{Φ} or unrolled G) where Φ is defined as follows:*

Φ : *while*$(A^{nontrivial} \neq \varnothing)$
do {
$a = find(G)$;
$G = G^{\eta_a^{\Gamma}}$;
}
return $G^{\eta^{\Gamma}}$;

Theorem 3. *Let $G \in \Gamma^{EF^+}$ be a finite Extensive Form$^+$ game satisfying perfect recall. Then G^{Φ} is a finite Extensive Form game satisfying perfect recall.*

Proof. As η_a^{Γ} essentially consists only from operations *add* and *delete* and both of them, as well as η^{Γ}, preserve properties of perfect recall and finiteness of the game (Lemma 1 and Lemma 2), it follows that if we do only finitely many Single Unrolls in Φ then G^{Φ} is a finite Extensive Form game satisfying perfect recall. So the only thing we have to prove is that there are finitely many Single Unroll operations made in Φ. In order to omit too cumbersome notation, we will prove it in an informal style:

Note that unrolling a single nontrivial action doesn't increase the depth of the game tree. Also note that with each Single Unroll the nontrivial action a that is being unrolled is replaced with trivial actions and if any nontrivial actions are added then they are added deeper in the game tree than action a was. So by doing Single Unrolls in Φ we gradually eliminate nontrivial actions starting from the top of the tree and constantly go deeper into the game tree to find new nontrivial actions to unroll. That being said, we will have to do only finitely many Single Unrolls.

Thus G^{Φ} is also finite and satisfies perfect recall. ∎

Example: Unrolling Dice Game

As an example of unrolling a nontrivial game (Figures 7.1, 7.2, 7.3) we will unroll Dice Game with $B = 3$. Note that independent of how much money Bob begins the game with (be it $B = 0$, $B = 3$ or $B = 1000$) both natural description and Extensive Form$^+$ description (almost) don't change its size. In contrast to the size of 11 nodes that the Extensive Form$^+$ has for any value of B, equal Extensive Form description size varies considerably[1].

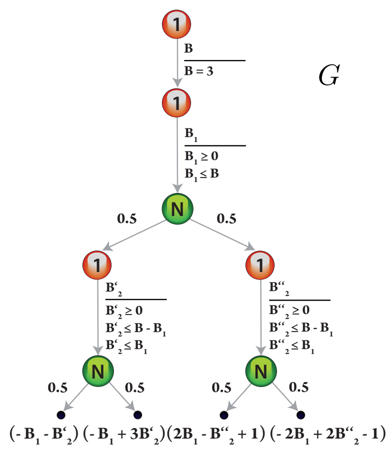

Figure 7.1: Original description of Dice Game in Extensive Form$^+$.

[1] In this example, equal Extensive Form description has a size of 54 nodes for $B = 3$, 300 nodes for the case of $B = 10$ and 2005003 nodes for the case $B = 1000$.

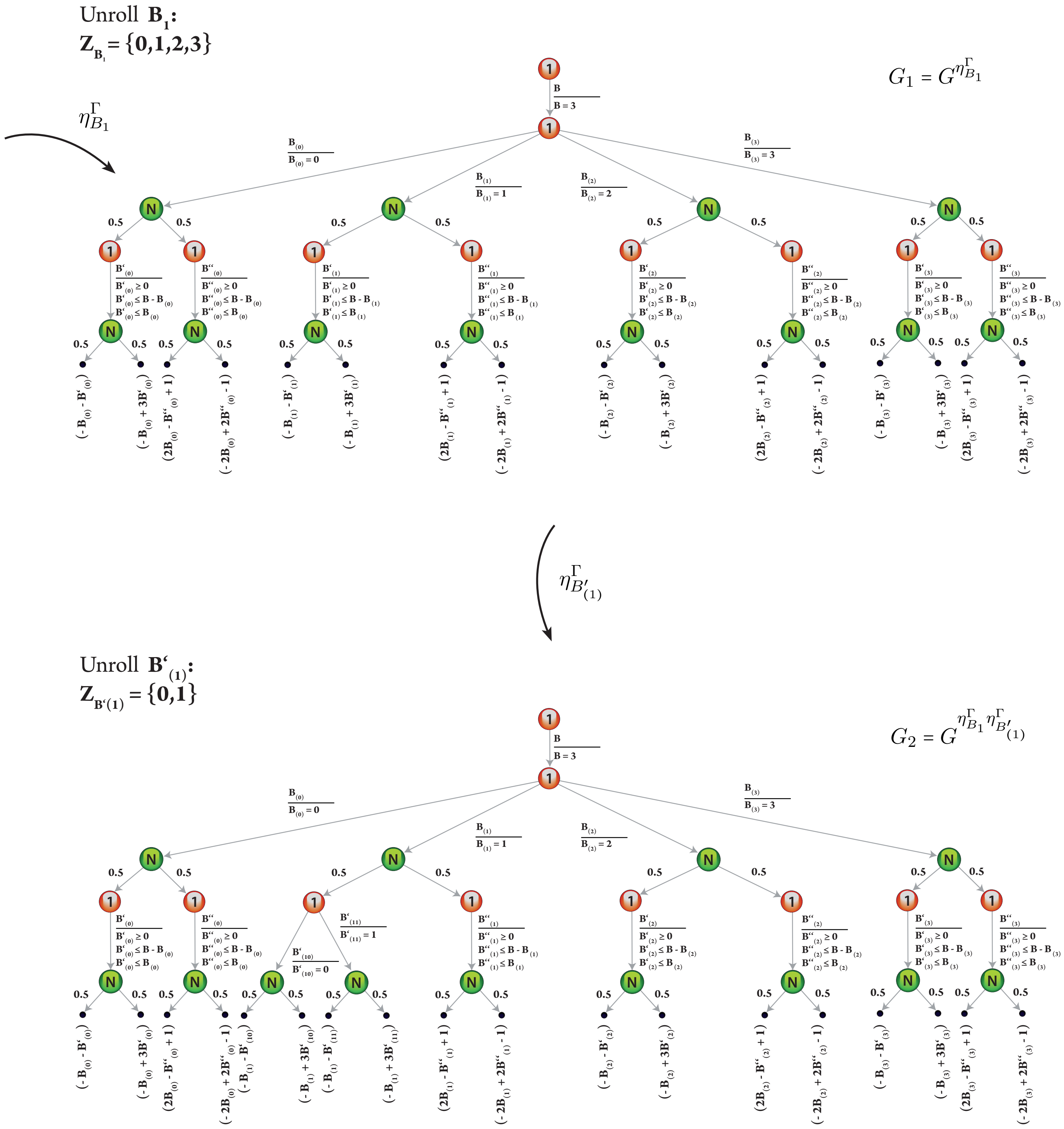

Figure 7.2: Description of Dice Game in Extensive Form+ after unrolling nontrivial action B and then new nontrivial action $B'_{(1)}$.

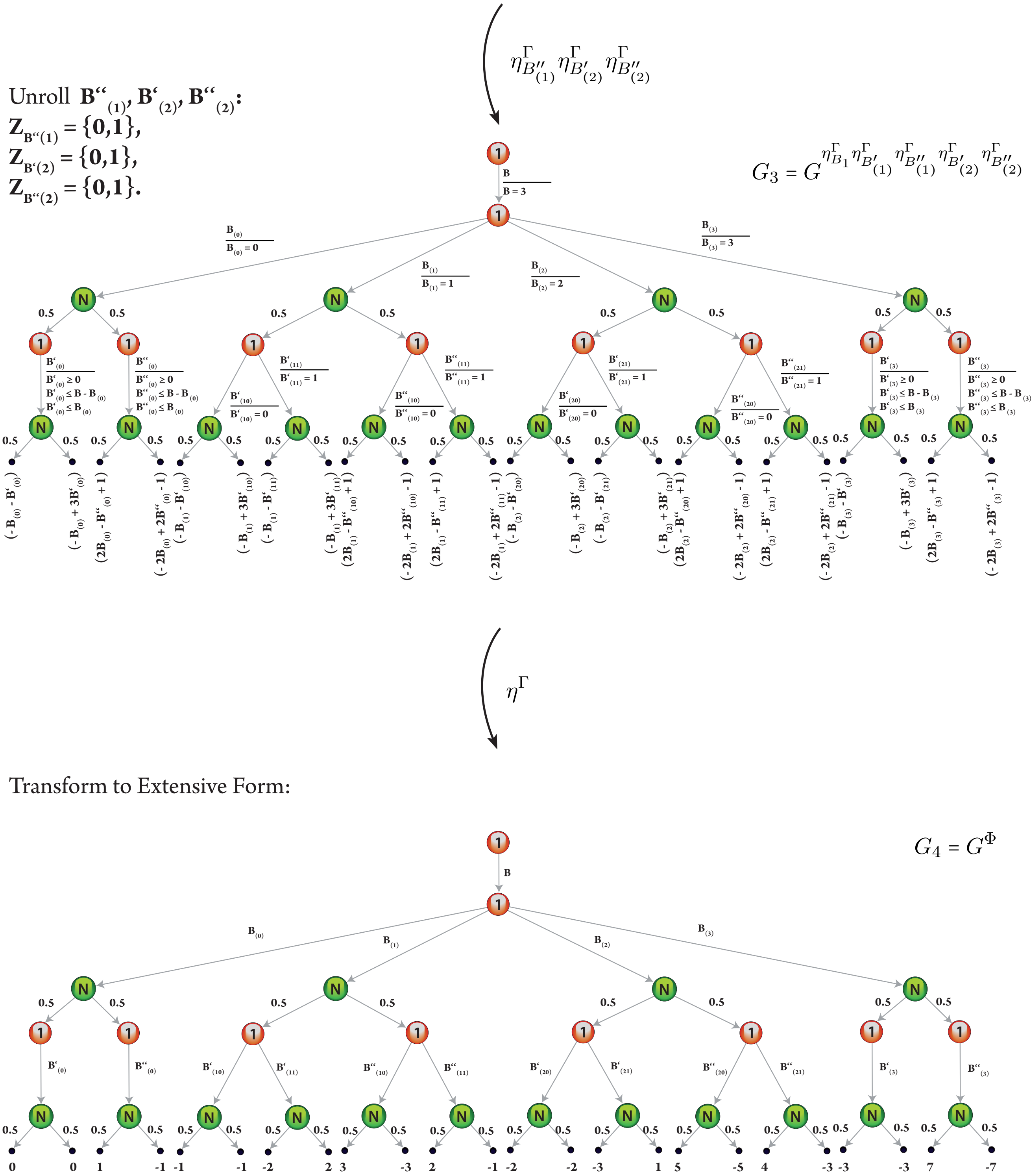

Figure 7.3: Unrolling last three nontrivial actions and converting the description of Dice Game to its equal description in Extensive Form.

6

Notion of Strategy - Extensive Form$^+$

In this chapter we propose a definition of a strategy for Extensive Form$^+$. For this chapter let $G = (P, A, V, E, I, \lambda, \sigma_N, (C_a)_{a\in A}, (\tau_p)_{p\in P}) \in \Gamma^{EF^+}$ be some finite Extensive Form$^+$ game.

6.1 Partial Assessment with Pure Allocation (PAPA)

Let t be some allocation for G and (σ, β) a Bayesian consistent assessment, defined exactly as it was defined in Chapter four for Extensive Form games.

Definition 25. $(\sigma, \beta, t) \in (\times_{p\in P} \times_{s\in I_p} \Delta(A(s))) \times (\times_{s\in I} \Delta(s)) \times T$ *is a Partial Assessment with Pure Allocation (or shortly, PAPA).*

The game value of (σ, β, t) let be defined similarly to game value of behavioral strategies:

$$u_p^{\sigma,t}(v_0) := \sum_{v_g\in V_T} ((\Pi_{a'\in Path(v_g)}\sigma(a'))(\Pi_{e\in Path_N(v_g)}\sigma_N(e))\tau_p^t(v_g)).$$

Even though PAPA is defined on all information sets of G, (σ, β, t) will be later constructed from a Bayesian consistent assessment built for an unrolled game G^Φ, (σ, β, t) depicting only part of that assessment. That is why it is called *Partial* Assessment with Pure Allocation. And because we use a single allocation in that strategy, we call it Partial Assessment with *Pure* Allocation.

6.1.1 Sufficient PAPA

Imagine we started playing G according to some PAPA (σ, β, t), and then some player changes his mind in the middle of the play and commits himself to another strategy. Should we continue playing according to (σ, β, t) or should we calculate a new strategy? It is a good question because if some player deviates from playing according to the allocation t then it could change payoffs at the end of the game and our initial strategy may lead us to an outcome dramatically different from our expectations. Moreover, it can make it impossible for us to assign some action a the value $t(a)$ which is prescribed by (σ, β, t) which makes playing according to that strategy impossible. But probably the worst thing of players deviating from playing with allocation t is that we cannot really compare this PAPA with other PAPA's - because we don't really know what will the value of the game be if someone deviates from t. In other words, if players don't commit to play with certain allocation, we cannot use the criteria of best response and, consequently, cannot define a Nash Equilibrium in PAPA's.

So it seems to be reasonable at least to demand that participants commit to playing with allocation t if we want to use PAPA as a strategy.

Definition 26. *(σ, β, t) is called sufficient in G under certain conditions if participants can't deviate from the allocation t under these conditions.*

We have intentionally defined sufficiency in such a way that we don't demand directly that all participants commit to playing with an allocation t. The reader might still think, that the assumption that no player will deviate from playing with t is too strong. In the following theorem we show an example how this assumption can be reformulated in a way that seems to be much more plausible:

Let $P' \subseteq P$ be the set of players such that $A^{nontrivial} \subseteq \dot{\cup}_{p' \in P'} A_{p'}$.

Theorem 4. *If all players $p' \in P'$ commit to play with allocation t in G, then (σ, β, t) is sufficient.*

Proof. We will prove that if players $p' \in P'$ commit to play with allocation t in G then there is a single allocation players can commit to, namely t:

Let's assume there be another allocation $t' \in T, t' \neq t$ the players can commit to play with. Then there exists some play in G, $\nu = (v_0, ..., v_g)$ with actions $Path(v_g)$ assigned along it, such that at least one of these actions would be assigned different values in both allocations. Let $a \in Path(v_g)$ be an action assigned to some edge (v_{i-1}, v_i) (with $1 \leq i \leq g$) that is chosen in such a way that it holds $(\forall a' \in Path(v_{i-1}) : t(a') = t'(a')) \wedge (t(a) \neq t'(a))$. Then there are only two cases:

Case 1 $a \in A^{nontrivial}$.

Then the action belongs to one of the players in P'. But they commit to play such actions with the allocation t which means a cannot be assigned a different value $t'(a) \neq t(a)$ and therefore players cannot commit to playing with t'. ↯

Case 2 $a \in A^{trivial}$.

From definitions of functions $Prec$ and $Path$ follows:

$(\diamond)$

$$Prec(a) \subseteq \{a | a \in Path(v_i)\},$$

But from this statement and from the fact that we have two different allocations that assign the same values to the action from $\{a | a \in Path(v_i)\}$ follows:

$$\exists t_0, t_1 \in T : (\forall a' \in Path(a) : t_0(a') = t_1(a')) \wedge (t_0(a) \neq t_1(a)) \Rightarrow^{(\diamond)}$$

$$\exists t_0, t_1 \in T : (\forall a' \in Prec(a) : t_0(a') = t_1(a')) \wedge (t_0(a) \neq t_1(a))$$

which means $a \in A^{nontrivial}$ (follows from $(\diamond.0)$ in Chapter 2). ↯

Now that we have proven that all players commit to play with t, from Theorem 4 follows directly that (σ, β, t) is sufficient. ∎

Because we do not demand anything from players who don't have nontrivial actions, they can try to change their strategies whenever they want during the play (as we have proven, however, they wouldn't be able to change the allocation). We even allow player(s) $p' \in P'$ to deviate from their strategy parts $(\sigma_{p'}, \beta_{p'})$ during the play. If we formulate a game in such a way that only, say, our player has nontrivial actions then if we hold on to playing with allocation t then σ remains being sufficient without demanding anything from our opponents.

Summing it up, the privilege of using nontrivial actions lays an obligation on the player of being able to agree with other 'privileged' players on an allocation if he wants to use PAPA as a strategy.

7

Solution Concept - Extensive Form$^+$

In this chapter we propose the following idea as a solution concept: Let $G \in \Gamma^{EF^+}$ be a finite Extensive Form$^+$ game satisfying perfect recall and $G^\Phi \in \Gamma^{EF}$ be its equal game in Extensive Form. We will define a way to reconstruct certain assessments (σ', β') in G^Φ into a PAPA (σ, β, t) for the original game G so that they both have the same game value. We will call such PAPA a Partial Sequential Equilibrium (shortly PSE) if (σ', β') is a Sequential Equilibrium and, as a main result of this thesis, we will prove that such a solution exists for every finite Extensive Form$^+$ game satisfying perfect recall.

But before we formally define Partial Sequential Equilibrium, we will first show how for every finite Extensive Form$^+$ game G, certain assessments defined for G^Φ can be converted into PAPA for G so that both the assessment and PAPA have the same game value:

7.1 Simple Behavioral Strategy

Let $G = (P, A, V, E, I, \lambda, \sigma_N, (\tau_p)_{p \in P}) \in \Gamma^{EF}$ be some game in Extensive Form. Let $v \in V$ be some game state in G that belongs to a singleton information set $s = \{v\} \in I$. Let's denote with $V_v^{sub} := \{v' | v' \in V^+(v), G^{v'}$ is a subgame $\}$ a set[1] of successors of v that lead to a subgame in G. Also, $A_s^{sub} := \{\lambda((v, v')) | v' \in V_v^{sub}\} \subseteq A(s)$ denotes actions leading to some subgame right after v.

Definition 27. *A behavioral strategy profile σ is called simple if for each $v \in V$ which happens to belong to a singleton information set, holds:*

$$\forall v' \in V_v^{sub} : \sigma(\lambda(v, v')) > 0 \Rightarrow (\forall v'' \in (V_v^{sub} \backslash \{v'\}) : \sigma(\lambda(v, v'')) = 0).$$

Simply put, a simple behavioral strategy is such a strategy that for each node v in the game, if it belongs to a singleton information set $s = \{v\}$, at most one of the actions from $A(s)$ that lead to a subgame, is assigned a nonzero probability. This property may seem strange at first but it is exactly the property that allows us to construct PAPA in Extensive Form$^+$ games from such simple strategy profiles (defined for the equal Extensive Form representation of the game) in such a way that PAPA will have the same game value as this simple behavioral strategy.

[1] Note that this definition is similar to the restriction we layed on finite Extensive Form$^+$ games in ($\diamond$.1) and ($\diamond$.2) in Chapter 2.

7.2 Transforming an Assessment with Simple Behavioral Strategy to PAPA

7.2.1 Transforming for Trivial Games

Let $G = (P, A, V, E, I, \lambda, \sigma_N, (C_a)_{a \in A}, (\tau_p)_{p \in P}) \in \Gamma^{EF^+}$ be some trivial finite Extensive Form$^+$ game. Let t be its only allocation and $G' = G^{\eta^\Gamma} = (P, A, V, E, I, \lambda, \sigma_N, (\tau'_p)_{p \in P}) \in \Gamma^{EF}$ be it's equal Extensive Form representation.

Let (σ, β) be a Bayesian consistent assessment for G' with σ being simple. We define (σ, β, t) as a PAPA for G and prove that

Lemma 3. (σ, β, t) *satisfies following properties:*

- σ *is simple,*
- (σ, β) *is Bayesian consistent,*
- (σ, β, t) *has the same game value as* (σ, β).

Proof. First two points follow directly from definition of (σ, β). The last point is also trivial (follows directly from definition of η^Γ. Let $p \in P$:

$$u_p^{\sigma,t}(v_0) =$$

$$\sum_{v_g \in V_T} ((\Pi_{a' \in Path(v_g)} \sigma(a'))(\Pi_{e \in Path_N(v_g)} \sigma_N(e)) \tau_p^t(v_g)) =$$

$$\sum_{v_g \in V_T} ((\Pi_{a' \in Path(v_g)} \sigma(a'))(\Pi_{e \in Path_N(v_g)} \sigma_N(e)) \tau'_p(v_g)) =$$

$$u_p^\sigma(v_0).$$

■

We will denote with η^Ω the transformation transforming (σ, β) to (σ, β, t). We will also write $(\sigma, \beta, t) = (\sigma, \beta)^{\eta^\Omega}$

7.2.2 Transforming for Nontrivial Games

Let $G = (P, A, V, E, I, \lambda, \sigma_N, (C_a)_{a \in A}, (\tau_p)_{p \in P}) \in \Gamma^{EF^+}$ be some finite Extensive Form$^+$ game and $G^{\eta_a^\Gamma} = (P, A', V', E', I', \lambda', \sigma'_N, (C'_a)_{a \in A'}, (\tau'_p)_{p \in P}) \in \Gamma^{EF^+}$ be G with an unrolled action $a \in A^{nontrivial}$.

Because of the restriction that nontrivial actions are assigned only to edges that connect two nodes, each belonging to a singleton set, there exists only one unique edge $(v, v^*) = e \in E$ such that $\lambda(e) = a$. From the definition of η_a^Γ follows that there is some finite set $Z = \{z_1, z_2, ..., z_l\}, \forall i \le l : z_i \in \mathcal{Z}$ of all values that could be assigned to action a (that is, values, for which exists an allocation that assigns that value to a) in G. Define a collection of subgame structures $(G_i)_{i \le l}$ in $G^{\eta_a^\Gamma}$ such that G_i denotes a subgame structure added during operation $add(G, a, z_i)$ during η_a^Γ. G^{v^*} is the original subgame structure in G, from which all the G_i were copied. A^{v^*} is a set of actions in G^{v^*}. Let also $v_i \in V'^+(v)$ denote a starting node of a subgame G_i in $G^{\eta_a^\Gamma}$ and $a_i := \lambda'((v, v_i))$ the action assigned to the edge $(v, v_i) \in E'$.

Now let's assume that there exists a PAPA (σ', β', t') in $G^{\eta_a^\Gamma}$ which has following properties:

- σ' is simple,
- (σ', β') is Bayesian consistent.

Now let's construct a PAPA for G which, supposedly, will also consist of a simple behavioral strategy profile, Bayesian consistent assessment, and have the same game value for each player, that (σ', β', t') has:

v belongs to a singleton information set and from definition of $(v_i)_{i \leq l}$ follows $\forall i \leq l : v_i \in V_v^{\prime sub}$. Then from the fact that σ' is simple follows that at most one action from $(a_i)_{i \leq l}$ is assigned a nonzero probability. Let a_j be that action (for the case when $\forall i \leq l : \sigma'(a_i) = 0$ let a_j be any action out of $(a_i)_{i \leq l}$.

What we essentially want is to copy strategy and belief system from G_j to G^{v^*}. Take functions δ_V, δ_A from the definition of the operation *add* for the case of $add(G, a, z_j)$. Now we define behavioral strategy σ, belief system β and an allocation t for the game G:

- Allocation t is defined as follows:
  - $\forall a' \in (A \backslash (A^{v^*} \cup \{a\})) : t(a') = t'(a')$,
  - $t(a) = t'(a_j)$,
  - $\forall a' \in A^{v^*} : t(a') = t'(\delta_A(a'))$,
- Behavioral strategy σ is defined as follows:
  - $\forall a' \in (A \backslash (A^{v^*} \cup \{a\})) : \sigma(a') = \sigma'(a')$,
  - $\sigma(a) = \sigma'(a_j)$,
  - $\forall a' \in A^{v^*} : \sigma(a') = \sigma'(\delta_A(a'))$,
- Belief system β is defined as follows:
  - $\forall v' \in (V \backslash V^{v^*}) : \beta(v') = \beta'(v')$,
  - $\forall v' \in V^{v^*} : \beta(v') = \beta'(\delta_V(v'))$,

Before we proceed to our Lemma, we will need following two equations:

Let $p \in P$ be some player. Let's take any path in $G^{\eta_a^\Gamma}$, starting at the root node v_0 and ending at some terminal node $v_g \in V_T'$. Let $i \leq l, i \neq j$.

If $v_i \in (v_0, v_g) - path$ then also $v \in (v_0, v_g) - path$ and $a_i \in Path(v_g)$. Thus according to σ' it holds $\sigma'(a_i) = 0$. Thus v_g will be reached with 0 probability:

$(\diamond.1)$

$$(\Pi_{a' \in Path(v_g)} \sigma'(a'))(\Pi_{e \in Path_N(v_g)} \sigma_N'(e)) =$$
$$\sigma'(a_i) \cdot (\Pi_{a' \in (Path(v_g) \backslash \{a_i\})} \sigma'(a'))(\Pi_{e \in Path_N(v_g)} \sigma_N'(e)) =$$
$$0 \cdot (\Pi_{a' \in (Path(v_g) \backslash \{a_i\})} \sigma'(a'))(\Pi_{e \in Path_N(v_g)} \sigma_N'(e)) = 0.$$

From that equation it follows:

$(\diamond)$

$$\sum_{\substack{v_g \in V_T' \\ \exists i \leq l, i \neq j : v_i \in (v_0, v_g) - path}} ((\Pi_{a' \in Path(v_g)} \sigma'(a'))(\Pi_{e \in Path_N(v_g)} \sigma_N'(e)) \tau_p^{\prime t'}(v_g)) =^{(\diamond.1)}$$
$$\sum_{\substack{v_g \in V_T' \\ \exists i \leq l, i \neq j : v_i \in (v_0, v_g) - path}} (0 \cdot \tau_p^{\prime t'}(v_g)) = 0.$$

Lemma 4. *We will prove that the constructed PAPA has the same qualities as (σ', β', t') and the same game value:*

- *σ is simple,*
- *(σ, β) is Bayesian consistent.*
- *(σ, β, t) has the same game value as (σ', β', t').*

Proof. We prove these properties in the order they are listed in the Lemma:

- Let's assume σ is not simple and there exists a node $v_{nsimple} \in V$, belonging to a singleton inormation set, such that σ doesn't satisfy the condition of simplicity in $v_{nsimple}$. Then there are two cases:

  Case 1 $v_{nsimple} \in V^{v^*}$.

  Then it follows from definition of σ, δ_V and δ_A that $\delta_V(v_{nsimple})$ doesn't satisfy the property of simplicity for σ' either. Thus, σ' is not simple. ↯

  Case 2 $v_{nsimple} \in (V \backslash V^{v^*})$.

  Then it follows from the fact that game structure and strategies of both games outside the subgame G^{v^*} are the same, that $v_{nsimple}$ doesn't satisfy the property of simplicity for σ' either. Thus, σ' is not simple. ↯

- For $s' \in I$ that are outside the subgame G^{v^*}, the Bayesian consistency of β automatically follows from the Bayesian consistency of β' for these information sets because the game structures of both G and $G^{\eta_a^\Gamma}$, and both σ and σ', are identical there. For any information set in the subgame G^{v^*} a Bayesian consistency property for β in any node $v' \in V^{v^*}$ follows from Bayesian consistency property for β' in $\lambda_V(v')$.

-
$$u_p^{\sigma,t}(v_0) =$$
$$\sum_{v_g \in V_T} ((\Pi_{a' \in Path(v_g)} \sigma(a'))(\Pi_{e \in Path_N(v_g)} \sigma_N(e)) \tau_p^t(v_g)) =$$

$$\sum_{\substack{v_g \in V_T \\ v^* \in (v_0, v_g)-path}} ((\Pi_{a' \in Path(v_g)} \sigma(a'))(\Pi_{e \in Path_N(v_g)} \sigma_N(e)) \tau_p^t(v_g)) +$$
$$\sum_{\substack{v_g \in V_T \\ v^* \notin (v_0, v_g)-path}} ((\Pi_{a' \in Path(v_g)} \sigma(a'))(\Pi_{e \in Path_N(v_g)} \sigma_N(e)) \tau_p^t(v_g)) =$$

$$\sum_{\substack{v_g \in V_T \\ v^* \in (v_0, v_g)-path}} ((\Pi_{a' \in Path(v_g)} \sigma(a'))(\Pi_{e \in Path_N(v_g)} \sigma_N(e)) \tau_p^t(v_g)) +$$
$$\sum_{\substack{v_g \in V'_T \\ \forall i \leq l : v_i \notin (v_0, v_g)-path}} ((\Pi_{a' \in Path(v_g)} \sigma'(a'))(\Pi_{e \in Path_N(v_g)} \sigma'_N(e)) \tau_p'^{t'}(v_g)) =$$

$$\sum_{\substack{v_g \in V'_T \\ v_j \in (v_0, v_g)-path}} ((\Pi_{a' \in Path(v_g)} \sigma'(a'))(\Pi_{e \in Path_N(v_g)} \sigma'_N(e)) \tau'^{t'}_p(v_g)) +$$
$$\sum_{\substack{v_g \in V'_T \\ \forall i \le l: v_i \notin (v_0, v_g)-path}} ((\Pi_{a' \in Path(v_g)} \sigma'(a'))(\Pi_{e \in Path_N(v_g)} \sigma'_N(e)) \tau'^{t'}_p(v_g)) =^{(\diamond)}$$

$$\sum_{\substack{v_g \in V'_T \\ \exists i \le l, i \neq j: v_i \in (v_0, v_g)-path}} ((\Pi_{a' \in Path(v_g)} \sigma'(a'))(\Pi_{e \in Path_N(v_g)} \sigma'_N(e)) \tau'^{t'}_p(v_g)) +$$
$$\sum_{\substack{v_g \in V'_T \\ v_j \in (v_0, v_g)-path}} ((\Pi_{a' \in Path(v_g)} \sigma'(a'))(\Pi_{e \in Path_N(v_g)} \sigma'_N(e)) \tau'^{t'}_p(v_g)) +$$
$$\sum_{\substack{v_g \in V'_T \\ \forall i \le l: v_i \notin (v_0, v_g)-path}} ((\Pi_{a' \in Path(v_g)} \sigma'(a'))(\Pi_{e \in Path_N(v_g)} \sigma'_N(e)) \tau'^{t'}_p(v_g)) =$$

$$\sum_{v_g \in V'_T} ((\Pi_{a' \in Path(v_g)} \sigma'(a'))(\Pi_{e \in Path_N(v_g)} \sigma'_N(e)) \tau'^{t'}_p(v_g)) =$$
$$u_p^{\sigma', t'}(v_0).$$

■

We will denote with η_a^{Ω} the transformation transforming (σ', β', t') to (σ, β, t). We will also write $(\sigma, \beta, t) = (\sigma', \beta', t')^{\eta_a^{\Omega}}$.

Let $G \in \Gamma^{EF^+}$ be some finite Extensive Form$^+$ game and $G^{\Phi} = G^{\eta_{a_1}^{\Gamma} \dots \eta_{a_l}^{\Gamma} \eta^{\Gamma}} \in \Gamma^{EF^+}$ be its equal Extensive Form representation. Let (σ', β') be some assessment in G^{Φ} that satisfies following two conditions:

- σ' is simple,
- (σ', β') is Bayesian consistent.

We define $\phi = \eta^{\Omega} \eta_{a_l}^{\Omega} \dots \eta_{a_1}^{\Omega}$ and a PAPA $(\sigma, \beta, t) = (\sigma', \beta')^{\phi}$ in G.

Theorem 5. *(σ, β, t) satisfies following three conditions:*

- *σ is simple,*
- *(σ, β) is Bayesian consistent.*
- *(σ, β, t) has the same game value as (σ', β').*

Proof. Follows from Lemma 3 and Lemma 4. ■

7.3 Partial Sequential Equilibrium

Let $G \in \Gamma^{EF^+}$. Let (σ', β') be some Assessment for G^Φ with σ' being simple. Define a PAPA $(\sigma, \beta, t) := (\sigma', \beta')^\phi$ for G.

Definition 28. *(σ, β, t) is a Partial Sequential Equilibrium if (σ', β') is a Sequential Equilibrium and (σ, β, t) has the same game value as (σ', β').*

7.3.1 Existence of Partial Sequential Equilibrium

In order to prove that there exists PSE for every finite Extensive Form$^+$ game G satisfying perfect recall it would suffice to show that for every finite Extensive Form game satisfying perfect recall exist a Sequential Equilibrium with a simple behavioral strategy: From Theorem 3 follows that G^Φ is a finite Extensive Form game satisfying perfect recall. From Theorem 1 follows that there exists a Sequential Equilibrium (σ, β) for G^Φ. (σ, β) is Bayesian consistent and if σ is simple then $(\sigma, \beta)^\phi$ would be SPE in G that has the same game value as (σ, β) has.

But in order to prove that there always exists a Sequential Equilibrium with a simple behavioral strategy, we will need to prove two Lemmas. First one is more of a trick that we will use to prove the second Lemma:

Preparations

Let D be some finite set, $(b^k)_{k \in \mathcal{N}}$ a sequence of probability distributions over D satisfying

(1) $\forall k \in \mathcal{N} \forall d \in D : b^k(d) \in (0, 1]$,

(2) $\forall k \in \mathcal{N} : b^k \in \Delta(D)$,

and a probability distribution b satisfying

(3) $lim_{k \to \infty}(b^k) = b$.

(4) $b \in \Delta(D)$,

Let $D' \subseteq D, d' \in D', k' \in K$. We define a new sequence $b^{k'}_{d'} : D \to R$ as follows:

$$(*) \qquad \begin{aligned} \forall d \in (D' \backslash \{d'\}) : \quad & b^{k'}_{d'}(d) := \tfrac{1}{k'} b^{k'}(d), \\ & b^{k'}_{d'}(d') := b^{k'}(d') + \tfrac{k'-1}{k'} \textstyle\sum_{d \in (D' \backslash \{d'\})} (b^{k'}(d)), \\ \forall d \in (D \backslash D') : \quad & b^{k'}_{d'}(d) := b^{k'}(d). \end{aligned}$$

and $b_{d'} : D \to R$ as follows:

$$(**) \qquad \begin{aligned} \forall d \in (D' \backslash \{d'\}) : \quad & b_{d'}(d) := 0, \\ & b_{d'}(d') := \textstyle\sum_{d \in D'} (b(d)), \\ \forall d \in (D \backslash D') : \quad & b_{d'}(d) := b(d). \end{aligned}$$

Lemma 5. $b_{d'}^{k'}$ *satisfies*

- $\forall d \in D : b_{d'}^{k'}(d) \in (0,1]$,
- $b_{d'}^{k'} \in \Delta(D)$,

and $b_{d'}$ *satisfies*

- $b_{d'} \in \Delta(D)$,
- $lim_{k\to\infty}(b_{d'}^{k}) = b_{d'}$.

Proof. We prove these four properties in the order they are listed in the lemma:

- As we have defined $b_{d'}^{k'}$ on three intervals (namely, on $(D'\backslash\{d'\})$, $\{d'\}$ and $(D\backslash D')$), we will also prove the fact that $\forall d \in D : b_{d'}^{k'}(d) \in (0,1]$ for each interval independently:
  - Let $d \in (D'\backslash\{d'\})$:
$$0 = \frac{1}{k'} \cdot 0 \overset{(1)}{<} \frac{1}{k'} b^{k'}(d) \overset{(*)}{=} b_{d'}^{k'}(d),$$
$$1 \geq \frac{1}{k'} \cdot 1 \overset{(1)}{\geq} \frac{1}{k'} b^{k'}(d) \overset{(*)}{=} b_{d'}^{k'}(d).$$
  - Now we prove that for $\{d'\}$:
$$0 = 0 + \frac{k'-1}{k'} \sum_{d\in(D'\backslash\{d'\})} 0 \overset{(1)}{<} b^{k'}(d') + \frac{k'-1}{k'} \sum_{d\in(D'\backslash\{d'\})} (b^{k'}(d)) \overset{(*)}{=} b_{d'}^{k'}(d'),$$
$$1 \overset{(2)}{\geq} b^{k'}(d') + \sum_{d\in(D'\backslash\{d'\})} (b^{k'}(d)) > b^{k'}(d') + \frac{k'-1}{k'} \sum_{d\in(D'\backslash\{d'\})} (b^{k'}(d)) \overset{(*)}{=} b_{d'}^{k'}(d'),$$
  - Let $d \in (D\backslash D')$:
$$0 \overset{(1)}{<} b^{k'}(d) \overset{(*)}{=} b_{d'}^{k'}(d).$$
$$1 \overset{(1)}{\geq} b^{k'}(d) \overset{(*)}{=} b_{d'}^{k'}(d).$$
- Now we prove that $b_{d'}^{k'} \in \Delta(D)$. We have already proven that $\forall d \in D : b_{d'}^{k'}(d) \in (0,1]$. Now all we have to prove is that $\sum_{d\in D}(b_{d'}^{k'}(d)) = 1$:

$$\sum_{d\in D}(b_{d'}^{k'}(d)) = \sum_{d\in(D'\backslash\{d'\})} (b_{d'}^{k'}(d)) + \sum_{d\in\{d'\}} (b_{d'}^{k'}(d)) + \sum_{d\in(D\backslash D')} (b_{d'}^{k'}(d)) \overset{(*)}{=}$$
$$\frac{1}{k'} \sum_{d\in(D'\backslash\{d'\})} b^{k'}(d) + b^{k'}(d') + \frac{k'-1}{k'} \sum_{d\in(D'\backslash\{d'\})} (b^{k'}(d)) + \sum_{d\in(D\backslash D')} (b^{k'}(d)) =$$
$$(\frac{1}{k'} + \frac{k'-1}{k'}) \sum_{d\in(D'\backslash\{d'\})} b^{k'}(d) + \sum_{d\in(D\backslash(D'\backslash\{d'\}))} (b^{k'}(d)) =$$
$$\sum_{d\in((D'\backslash\{d'\})\cup(D\backslash(D'\backslash\{d'\})))} b^{k'}(d) = \sum_{d\in D} b^{k'}(d) \overset{(2)}{=} 1.$$

- Now we prove that $b_{d'} \in \Delta(D)$:
  - Let $d \in (D' \backslash \{d'\})$:
$$0 \le 0 \overset{(**)}{=} b_{d'}(d) \overset{(**)}{=} 0 \le 1,$$
  - Now we prove it for d' :
$$0 = \sum_{d \in D'} 0 \overset{(4)}{\le} \sum_{d \in D'} b(d) \overset{(**)}{=} b_{d'}(d') \overset{(**)}{=} \sum_{d \in D'} b(d) \overset{(4)}{\le} \sum_{d \in D} b(d) = 1,$$
  - Let $d \in (D \backslash D')$:
$$0 \overset{(4)}{\le} b(d) \overset{(**)}{=} b_{d'}(d) \overset{(**)}{=} b(d) \overset{(4)}{\le} 1.$$

  Now that we have already proven that $\forall d \in D : b_{d'}(d) \in [0,1]$, we need to prove that $\sum_{d \in D}(b_{d'}(d)) = 1$:
$$\sum_{d \in D}(b_{d'}(d)) = \sum_{d \in (D' \backslash \{d'\})}(b_{d'}(d)) + \sum_{d \in \{d'\}}(b_{d'}(d)) + \sum_{d \in (D \backslash D')}(b_{d'}(d)) \overset{(**)}{=}$$
$$\sum_{d \in (D' \backslash \{d'\})} 0 + \sum_{d \in D'}(b(d)) + \sum_{d \in (D \backslash D')}(b(d)) = \sum_{d \in D}(b(d)) \overset{(4)}{=} 1.$$

- Finally, we prove that $lim_{k \to \infty}(b^k_{d'}) = b_{d'}$. We will rephrase that and prove that $\forall d \in D$: $lim_{k \to \infty}(b^k_{d'}(d)) = b_{d'}(d)$.
  - Let $d \in (D' \backslash \{d'\})$:
$$lim_{k \to \infty}(b^k_{d'}(d)) \overset{(*)}{=} lim_{k \to \infty}(\frac{1}{k} b^k(d)) = lim_{k \to \infty}(\frac{1}{k}) lim_{k \to \infty}(b^k(d)) =$$
$$0 \cdot lim_{k \to \infty}(b^k(d)) = 0 \overset{(**)}{=} b_{d'}(d).$$
  - Now we prove that $lim_{k \to \infty}(b^k_{d'}(d')) = b_{d'}(d')$:
$$lim_{k \to \infty}(b^k_{d'}(d')) \overset{(*)}{=} lim_{k \to \infty}(b^k(d') + \frac{k-1}{k} \sum_{d \in (D' \backslash \{d'\})}(b^k(d))) =$$
$$lim_{k \to \infty}(b^k(d')) + lim_{k \to \infty}(\frac{k-1}{k}) \cdot \sum_{d \in (D' \backslash \{d'\})}(lim_{k \to \infty}(b^k(d))) \overset{(3)}{=}$$
$$b(d') + 1 \cdot \sum_{d \in (D' \backslash \{d'\})}(b(d)) = \sum_{d \in D'}(b(d)) \overset{(**)}{=} b_{d'}(d')$$
  - Let $d \in (D \backslash D')$:
$$lim_{k \to \infty}(b^k_{d'}(d)) \overset{(*)}{=} lim_{k \to \infty}(b^k(d)) \overset{(3)}{=} b(d) \overset{(**)}{=} b_{d'}(d).$$

  Now that we have proven $\forall d \in D : lim_{k \to \infty}(b^k_{d'}(d)) = b_{d'}(d)$, it follows $lim_{k \to \infty}(b^k_{d'}) = b_{d'}$.

■

Let $G = (P, A, V, E, I, \lambda, \sigma_N, (\tau_p)_{p \in P}) \in \Gamma^{EF}$ be some finite Extensive Form game satisfying perfect recall and (σ, β) be a Sequential Equilibrium in G. Let's assume there exists some game state $v_1 \in V_p$ belonging to a singleton information set $s := \{v_1\} \in I_p$ for some player $p \in P$.

We remind that $V_{v_1}^{sub}$ denotes a set of successors of v_1 that lead to a subgame in G and A_s^{sub} denotes actions leading to some subgame right after v_1.

What we will try to do now, is to construct another assessment for which it holds that *at most* one of the actions from A_s^{sub} is assigned a nonzero probability.

($\diamond$) In case it holds $\forall a \in A_s^{sub} : \sigma(a) = 0$, we don't have to do anything. So let's assume there exists a $v_2 \in V_{v_1}^{sub}$ such that $\sigma(\lambda((v_1, v_2))) > 0)$. Define $a := \lambda((v_1, v_2)) \in A_s^{sub}$.

Now we will construct the desired assessment and prove that it is also a Sequential Equilibrium. According to definition of Sequential Equilibrium, there exists a sequence of Bayesian consistent assessments $(\sigma^k, \beta^k)_{k \in \mathcal{N}}$ with σ^k being completely mixed, for which holds $lim_{k \to \infty}(\sigma^k, \beta^k) = (\sigma, \beta)$. So let $(\sigma^k, \beta^k)_{k \in \mathcal{N}}$ be such a sequence.

Now there are several properties that $(\sigma^k)_{k \in \mathcal{N}}$ and σ hold:

- It holds $\forall k \in \mathcal{N} \sigma_{p,s}^k \in \Delta(A(s))$
- and because σ^k are completely mixed, it holds: $\forall k \in \mathcal{N} \forall a \in A(s) : \sigma_{p,s}^k(a) \in (0, 1]$.

And as for σ, it holds:

- $\sigma_{p,s} \in \Delta(A(s))$
- and because of $lim_{k \to \infty}(\sigma^k, \beta^k) = (\sigma, \beta)$, it holds: $lim_{k \to \infty}(\sigma_{p,s}^k) = \sigma_{p,s}$

Define a new sequence of behavioral strategies[1] $(\varsigma^k)_{k \in \mathcal{N}}$ such that (for each $k \in \mathcal{N}$):

- $\forall p' \in P \forall s' \in (I_{p'} \backslash \{s\}) : \varsigma_{p',s'}^k := \sigma_{p',s'}^k$,
- $\forall a' \in (A_s^{sub} \backslash \{a\}) : \varsigma_{p,s}^k(a') := \frac{1}{k} \sigma_{p,s}^k(a')$,
- $\varsigma_{p,s}^k(a) := \sigma_{p,s}^k(a) + \frac{k-1}{k} \sum_{a' \in (A_s^{sub} \backslash \{a\})} (\sigma_{p,s}^k(a'))$,
- $\forall a' \in (A(s) \backslash A_s^{sub}) : \varsigma_{p,s}^k(a') := \sigma_{p,s}^k(a')$,

and a new behavioral strategy ς such that:

($*$.1) $\forall p' \in P \forall s' \in (I_{p'} \backslash \{s\}) : \varsigma_{p',s'} := \sigma_{p',s'}$,

($*$.2) $\forall a' \in (A_s^{sub} \backslash \{a\}) : \varsigma_{p,s}(a') := 0$,

($*$.3) $\varsigma_{p,s}(a) := \sum_{a' \in A_s^{sub}} (\sigma_{p,s}(a'))$,

($*$.4) $\forall a' \in (A(s) \backslash A_s^{sub}) : \varsigma_{p,s}(a') := \sigma_{p,s}(a')$.

[1] Unfortunately, we will have to use another letter for a strategy in this case, as the notation gets too complicated. Luckily, it's the only place in this work we will have to do this.

Lemma 6. (ς, β) *is a Sequential Equilibrium.*

Proof. In order to prove that we need to prove that (ς, β) is a consistent and sequentially rational assessment:

- (**Consistency**)

  Using Lemma 5 it holds[1]:

  - $\forall k \in \mathcal{N} : \varsigma^k_{p,s}$ is completely mixed,
  - $lim_{k\to\infty}(\varsigma^k_{p,s}) = \varsigma_{p,s}$,

  and because we didn't change behavioral strategies in other information sets, it also holds that $(\varsigma^k)_{k\in\mathcal{N}}$ is a sequence of completely mixed strategies with $lim_{k\to\infty}(\varsigma^k) = \varsigma$. As it also holds $lim_{k\to\infty}(\beta^k) = \beta$. Thus it holds:

  $$lim_{k\to\infty}(\varsigma^k, \beta^k) = (\varsigma, \beta).$$

  Now the last thing to prove is that for each $k \in \mathcal{N}$ the belief system β^k is Bayesian consistent with ς^k. Let's fix some $k \in \mathcal{N}$ and let $s' \in I$ be some information set and $v' \in s'$ some node belonging to this information set:

  - Let's assume v' belong to some subgame G^v with $v \in (V^{sub}_{v_1} \backslash \{v_2\})$. That would mean that s' is contained in G^v. That also means that $\forall v'' \in s' : v_1, v \in (v_0, v'') - path$ which means that

    (1) $\forall v'' \in s' : \mathcal{P}_{\varsigma^k}(v'') = \Pi_{a' \in (v_0,v'')-path}(\varsigma(a')) \overset{(**.1),(**.2)}{=} \frac{1}{k}\Pi_{a' \in (v_0,v'')-path}(\sigma(a')) = \frac{1}{k}\mathcal{P}_{\sigma^k}(v'')$.

    which also means that

    (2) $\mathcal{P}_{\varsigma^k}(s') = \sum_{v'' \in s'}(\mathcal{P}_{\varsigma^k}(v'')) \overset{(1)}{=} \frac{1}{k}\sum_{v'' \in s'}(\mathcal{P}_{\sigma^k}(v'')) = \frac{1}{k}\mathcal{P}_{\sigma^k}(s')$.

    From both equations follows:

    $$\beta^k(v') = \frac{\mathcal{P}_{\sigma^k}(v')}{\mathcal{P}_{\sigma^k}(s')} \overset{(1),(2)}{=} \frac{\frac{1}{k}\mathcal{P}_{\sigma^k}(v')}{\frac{1}{k}\mathcal{P}_{\sigma^k}(s')} = \frac{\mathcal{P}_{\varsigma^k}(v')}{\mathcal{P}_{\varsigma^k}(v')}.$$

  - Similarly we prove Bayesian consistency if v' belongs to the subgame G^{v_2}.
  - If v' doesn't belong to any of the subgames G^v with $v \in V^{sub}_{v_1}$ then that means that no node from $V^{sub}_{v_1}$ occurs on the path from v_0 to any $v'' \in s'$ which means that probabilities assigned to actions on this path remain the same. Thus,

    (3) $\forall v'' \in s' : \mathcal{P}_{\varsigma^k}(v'') \overset{(**.1),(**.4)}{=} \mathcal{P}_{\sigma^k}(v'')$,

    (4) $\mathcal{P}_{\varsigma^k}(s') \overset{(3)}{=} \mathcal{P}_{\sigma^k}(s')$

    and it holds:

    $$\beta^k(v') = \frac{\mathcal{P}_{\sigma^k}(v')}{\mathcal{P}_{\sigma^k}(s')} \overset{(3),(4)}{=} \frac{\mathcal{P}_{\varsigma^k}(v')}{\mathcal{P}_{\varsigma^k}(v')}.$$

  Thus it holds:

  $$\forall s'' \in I \forall v'' \in s'' : \beta^k(v'') = \frac{\mathcal{P}_{\varsigma^k}(v'')}{\mathcal{P}_{\varsigma^k}(s'')},$$

  which means that β^k is Bayesian consistent with ς^k.

[1] Replace D with $A(s)$, D' with A^{sub}_s, d' with a, b^k with $\sigma^k_{p,s}$, b with $\sigma_{p,s}$, $b^k_{d'}$ with $\varsigma^k_{p,s}$ and $b_{d'}$ with $\varsigma_{p,s}$.

- (**Sequential Rationality**)
  - First we prove the property of sequential rationality for a:

    Because of the fact that (σ, β) is sequentially rational and $\sigma_{p,s}(a) \overset{(\diamond)}{>} 0$, it follows from the definition of sequential rationality:

    $$(I)\ \ \sum_{v' \in s}(\beta(v')u^{\sigma}_{p,s}(v', a)) = max_{a' \in A(s)} \sum_{v' \in s}(\beta(v')u^{\sigma}_{p,s}(v', a')).$$

    But because s is a singleton and because σ and ς coincide in the subgame G^{v_2}, it also holds:

    $$(I.a)\ \ \sum_{v' \in s}(\beta(v')u^{\sigma}_{p,s}(v', a)) \overset{s=\{v_1\}}{=} \beta(v_1)u^{\sigma}_{p,s}(v_1, a) =$$

    $$1 \cdot u^{\sigma}_{p}(v_2) \overset{(**.1)}{=} u^{\varsigma}_{p}(v_2) = \sum_{v' \in s}(\beta(v')u^{\varsigma}_{p,s}(v', a)).$$

    and because σ and ς coincide in every subtree $G^{v'}$ for every $v' \in V^{+}(v_1)$, it also holds:

    $$(I.b)\ \ max_{a' \in A(s)} \sum_{v' \in s}(\beta(v')u^{\sigma}_{p,s}(v', a')) \overset{s=\{v_1\}}{=} max_{a' \in A(s)}(\beta(v_1)u^{\sigma}_{p,s}(v_1, a')) =$$

    $$max_{a' \in A(s)}(u^{\sigma}_{p,s}(v_1, a')) \overset{(**.4)}{=} max_{a' \in A(s)}(u^{\varsigma}_{p,s}(v_1, a')) =$$

    $$max_{a' \in A(s)} \sum_{v' \in s}(\beta(v')u^{\varsigma}_{p,s}(v', a')).$$

    From adding $(I.a)$ and $(I.b)$ into (I) follows:

    $$\sum_{v' \in s}(\beta(v')u^{\varsigma}_{p,s}(v', a)) = max_{a' \in A(s)} \sum_{v' \in s}(\beta(v')u^{\varsigma}_{p,s}(v', a')).$$

  - ς assigns 0 probability to all actions $a' \in (A^{sub}_{s} \backslash \{a\})$ so we don't have to prove the property from (I) for them.
  - The property of sequential rationality relies only on the value of utility functions in each node of the game and the system of belief. The belief system wasn't changed when constructing (ς, β). And if we show that the value of the utility function for each player in v_1 didn't change then it would follow that the values of utility functions in other nodes didn't change either (because we changed behavioral strategy profile only on actions from $A^{sub}_{s} \subseteq A(s)$). So then it would follow that the property of sequential rationality of (ς, β) automatically holds for all other actions $a' \in (A \backslash A^{sub}_{s})$.

    So now we just show that $u^{\sigma}_{p}(v_1) = u^{\varsigma}_{p}(v_1)$:

    $$u^{\sigma}_{p}(v_1) = \sum_{a' \in A(s)} (\sigma(a')u^{\sigma}_{p,s}(v_1, a')) =$$

    $$\sum_{a' \in A^{sub}_{s}} (\sigma(a')u^{\sigma}_{p,s}(v_1, a')) + \sum_{a' \in (A(s) \backslash A^{sub}_{s})} (\sigma(a')u^{\sigma}_{p,s}(v_1, a')) =^{1}$$

[1]Because $(u^{\sigma}_{p,s}(v_1, a') \neq max_{a'' \in A(s)} \sum_{v' \in s}(\beta(v')u^{\sigma}_{p,s}(v', a''))) \Rightarrow \sigma(a') = 0$.

$$\sum_{a'\in A_s^{sub}}(\sigma(a')(max_{a''\in A(s)}\sum_{v'\in s}(\beta(v')u_{p,s}^{\sigma}(v',a'')))) + \sum_{a'\in(A(s)\setminus A_s^{sub})}(\sigma(a')u_{p,s}^{\sigma}(v_1,a')) =$$

$$(\sum_{a'\in A_s^{sub}}\sigma(a'))(max_{a''\in A(s)}\sum_{v'\in s}(\beta(v')u_{p,s}^{\sigma}(v',a'')))+\sum_{a'\in(A(s)\setminus A_s^{sub})}(\sigma(a')u_{p,s}^{\sigma}(v_1,a')) \overset{(**.3),(I.a)}{=}$$

$$(\varsigma(a))(u_p^{\varsigma}(v_2)) + \sum_{a'\in(A(s)\setminus A_s^{sub})}(\varsigma(a')u_{p,s}^{\varsigma}(v_1,a')) \overset{u_p^{\varsigma}(v_2)=u_{p,s}^{\varsigma}(v_1,a)}{=}$$

$$0 + (\varsigma(a))(u_{p,s}^{\varsigma}(v_1,a)) + \sum_{a'\in(A(s)\setminus A_s^{sub})}(\varsigma(a')u_{p,s}^{\varsigma}(v_1,a')) =$$

$$\sum_{a'\in(A_s^{sub}\setminus\{a\})}(0\cdot u_{p,s}^{\varsigma}(v_1,a')) + (\varsigma(a))(u_{p,s}^{\varsigma}(v_1,a)) + \sum_{a'\in(A(s)\setminus A_s^{sub})}(\varsigma(a')u_{p,s}^{\varsigma}(v_1,a')) \overset{(**.2)}{=}$$

$$\sum_{a'\in(A_s^{sub}\setminus\{a\})}(\varsigma(a')u_{p,s}^{\varsigma}(v_1,a')) + (\varsigma(a))(u_{p,s}^{\varsigma}(v_1,a))) + \sum_{a'\in(A(s)\setminus A_s^{sub})}(\varsigma(a')u_{p,s}^{\varsigma}(v_1,a')) =$$

$$\sum_{a'\in A(s)}(\varsigma(a')u_{p,s}^{\varsigma}(v_1,a')) = u_p^{\varsigma}(v_1).$$

Though the proof of sequential rationality may seem a bit cumbersome, the idea is simple: in the original Sequential Equilibrium actions from A_s^{sub} that were assigned nonzero probabilities, led to the subgames which guaranteed maximum payoff for the player p. So because of the *maximum* criterium those 'chosen' subgames must all guarantee *equal* payoffs for p (otherwise those that guarantee less would be assigned 0 probability by σ). That means that p is theoretically indifferent to which one of those subgames it chooses. So what we did, essentially, by constructing (ς, β) is that we just demanded that p would choose only one of those equally great subgames when being at the game state v_1.

■

Remark 4. *Note that the (ς, β) constructed in this Lemma doesn't change probabilities assigned to actions not belonging to A_s^{sub}. That means that (ς, β) can be constructed in such a way that if σ was not simple, ς would have one less such node $v \in V$ with $s_v = \{v\} \in I$ that assigns more than one action from $A_{s_v}^{sub}$ a nonzero probability than (σ, β).*

Now we are ready to prove a following beautiful result:

Theorem 6. *There exists a Sequential Equilibrium (σ', β') in G with σ' being simple.*

Proof. As G is a finite Extensive Form game satisfying perfect recall, from Theorem 1 follows that there exists a Sequential Equilibrium in G. Assume there exists no Sequential Equilibrium with a simple behavioral strategy profile for G. Then let's take such a Sequential Equilibrium that has a minimum amount of such nodes $v \in V$ for which the simplicity property doesn't hold (let's assume there are l such nodes). Then from Lemma 6 and Remark 4 follows that there exists a Sequential Equilibrium with $l-1$ such nodes for which that equation doesn't hold. $\lightning$ ■

Theorem 7. *(Main Result) For every finite Extensive Form$^+$ game G satisfying perfect recall exists PAPA that is a Partial Sequential Equilibrium.*

Proof. From Theorem 3 follows that G^{Φ} is a finite Extensive Form game satisfying perfect recall. From Theorem 6 follows that there exists a Sequential Equilibrium (σ', β') for G^{Φ} with σ being simple. From Theorem 5 follows that $(\sigma, \beta, t) := (\sigma', \beta')^{\phi}$ is a PAPA for G that has the same game value as (σ', β'). Thus (σ, β, t) is a Partial Sequential Equilibrium. ■

Example: Constructing PSE in the Dice Game

In conclusion, we will illustrate how we construct a PSE on a Dice Game (for $B = 3$). On Figure 8.1 we show two Sequential Equilibria, $(\sigma_{SE}, \beta_{SE})$ and $(\sigma'_{SE}, \beta'_{SE})$ for the unrolled version of Dice Game we constructed in Chapter 5. In $(\sigma_{SE}, \beta_{SE})$, σ_{SE} is not simple. $(\sigma'_{SE}, \beta'_{SE})$ is constructed from $(\sigma_{SE}, \beta_{SE})$ using the construction from Lemma 6. Then, using the construction from Lemma 3 and Lemma 4, we gradually (Figures 8.2, 8.3) construct a PSE for the original Extensive Form^{+} description of the game, presented in the end of Chapter 2. All assessments have the game value of 0.25 which means that Bob has, how gamblers call it, an 'Edge'[1] in this game. Unfortunately we aren't able to fully show what role the concept of sufficiency, presented in Chapter 4, plays. That is mostly connected with the fact that the example game we have chosen is played by one player.

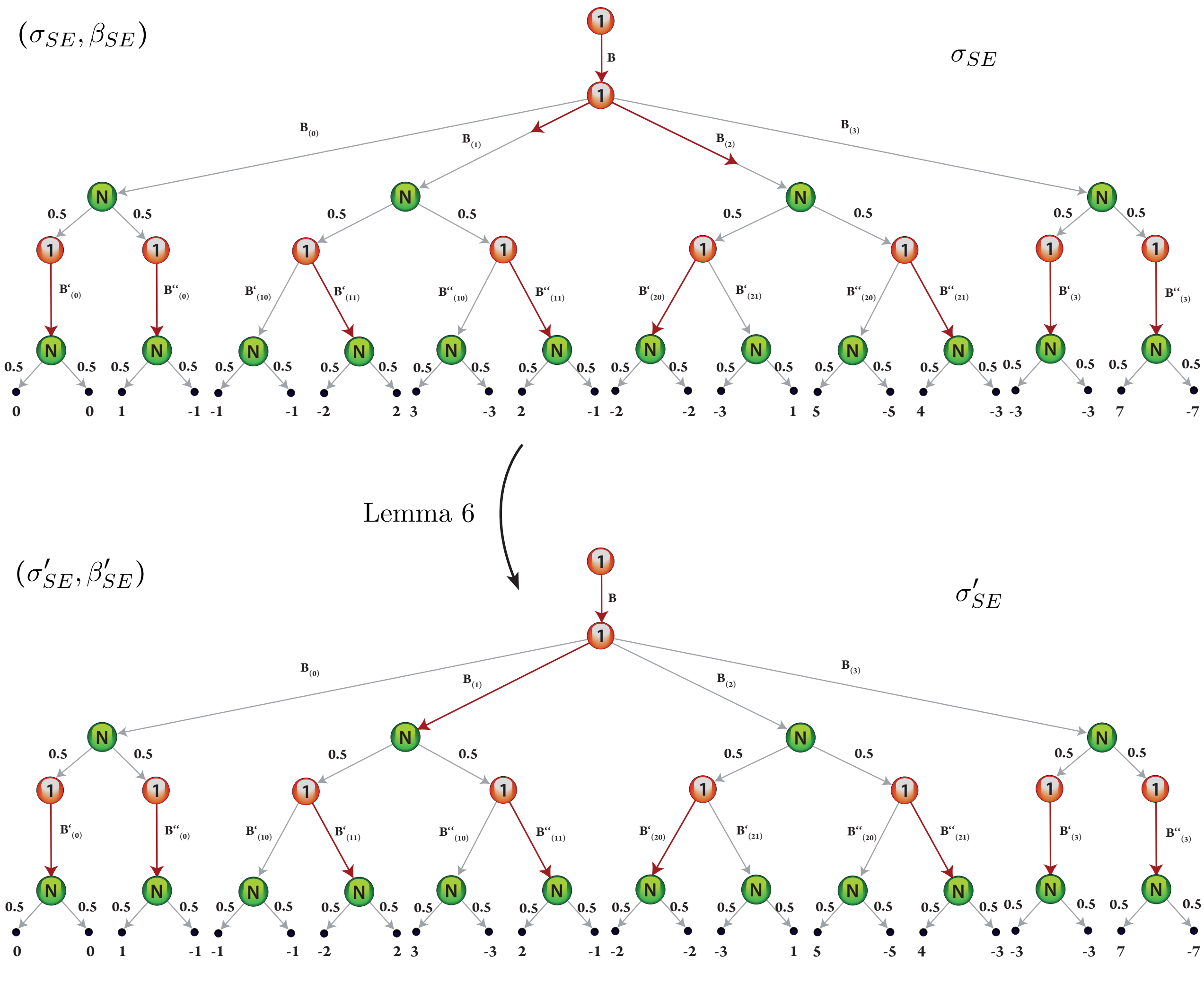

Figure 8.1: Two Sequential Equilibria for the unrolled version of Dice Game, one with simple behavioral strategy (lower one) and one without.

[1]Which means that in a long term, if Bob would play this game according to defined PSE, he would profit from this game.

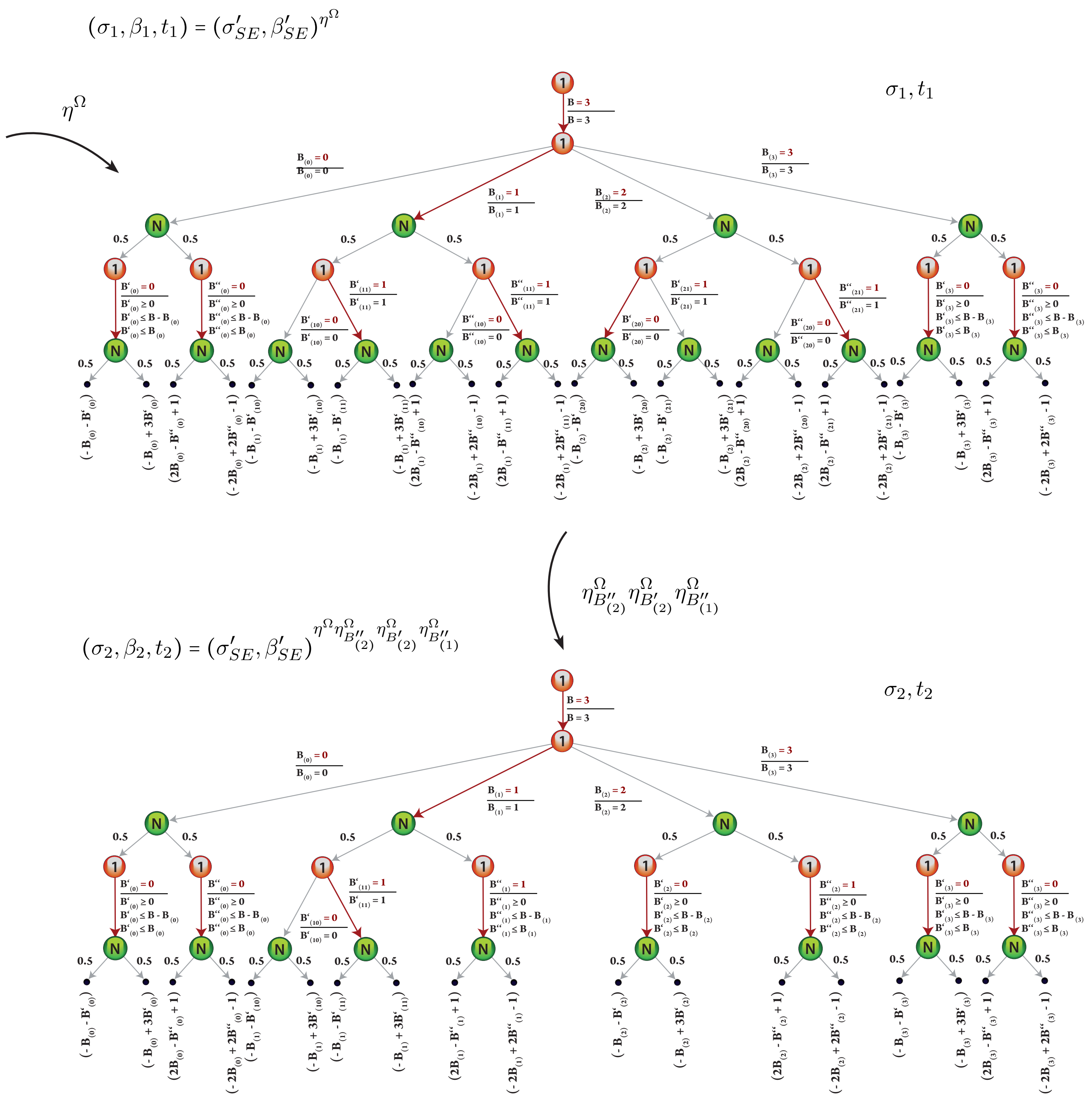

Figure 8.2: Constructing PAPA for the stages of unrolling the Dice Game that were used in the example of Chapter 5.

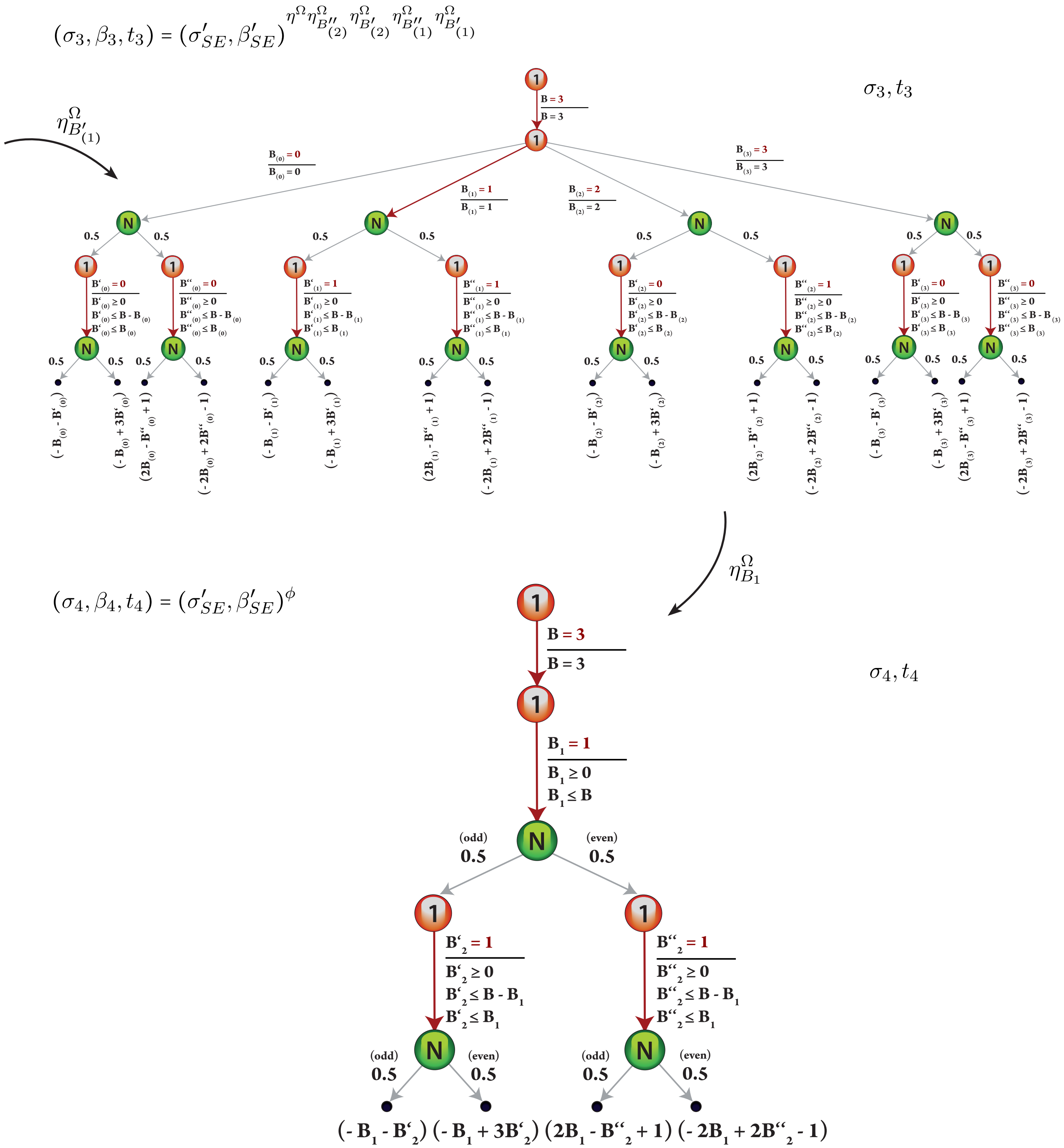

Figure 8.3: Constructing PSE that has the same value as the Sequential Equilibria on Figure 8.1, for the original description of the Dice Game in Extensive Form⁺.

8

Future Research

In this chapter I shortly present some ideas I have gathered during the writing period for this thesis but didn't find enough time to explore. I believe that some of them could lead to interesting results and if anybody would find any of these ideas interesting, I would only appreciate any further research.

- **Finding PSE For 2-Player Extensive Form$^+$ Games**

  In (17) it was shown how to find a Nash Equilibrium in two-player extensive form games satisfying perfect recall, using linear programming techniques. The result was further refined in (7) where the idea of how to calculate a Sequential Equilibrium for such games was described.

  If we would, in the same way, construct such a program for a two-player Extensive Form$^+$ game with perfect recall, we would get a similar-looking program where real numbers in the constraint matrix are replaced by linear combinations of values assigned to actions. Then we could also add all the constraints we've set on actions to that same matrix. As a result, we would get a Mixed Integer Quadratically Constrained Program (MIQCP) to solve[1]. One can hope that solving such a program would result in a PSE.

- **Remove the Constraint on Information Sets**

  In the second chapter we have laid a restriction[2] on information sets in Extensive Form$^+$. Exactly that restriction doesn't give us possibility to use the expressive power of Extensive Form$^+$ in describing such games as Poker.

  If we would allow to include nodes, that belong to edges with nontrivial actions, in more complex information sets then we would also have to redefine Φ and find a new notion of strategy, as the proposed notion of PAPA may not suffice to prove the existence of PSE in these games.

[1] For a curious reader, there are quite a few (commercial) software packages out there that are able to solve MIQCP and, more general, MINLP problems: Matlab, ALPHAECP, BARON 9.0, DICOPT, KNITRO 6.0, LINDOGLOBAL 6.0, OQNLP, SBB, AMPLwrap, Kestrel, GUROBI. CPLEX 12.1 and MOSEK 6 can solve MIQCP problems as well.

[2] ($\diamond$.1) and ($\diamond$.2) in section of Chapter 2, where we formally defined finite Extensive Form$^+$ games.

- **Extensive Form$^+$ as a Lossless Abstraction Technique**

  Nowadays one of the biggest problems is not how to solve games but how to solve BIG games. In other words, real world problems formulated in Extensive Form often have too big description (Poker).

  So the idea is to view Extensive Form$^+$ not as a new way of describing and designing games, but as a more compact way to describe Extensive Form games. One could develop an abstraction technique that tries to find the smallest Extensive Form$^+$ representation for an Extensive Form game. I believe that would result in a much more compact description for some games which in turn could allow much bigger problems formulated in Extensive Form to be solved.

- **Generalization of Extensive Form$^+$**

  In this thesis we have never used the fact that the values assigned to actions are linearly constrained. We have only used the fact that the amount of possible different allocations is finite. That means that even though linear constraints seem to be a natural way of defining an interconnected structure that forms the payoffs, they don't have to be necessarily linear.

- **Alternative way to calculate Sequential Equilibrium in Extensive Form Games**

  As it was proven in the main result, for each Extensive Form$^+$ game there exists a PAPA which is a PSE. If we could calculate such a PAPA, we could project this strategy on an unrolled Extensive Form - and get a part of some Sequential Equilibrium. If we then change constraints of the original Extensive Form$^+$ game a bit, calculate a SPE for it and project it again on the same unrolled Extensive Form game - we would probably get another part of the Sequential Equilibrium. In such a way we could recursively define a Sequential Equilibrium for an unrolled Extensive Form game - not by solving the whole game - but by solving many (probably much) smaller Extensive Form$^+$ games.

9

Related Work

In this chapter some research related with the topic of this thesis is presented. It is not known to me if anybody has approached Extensive Form in a similar manner it is approached in this work. Here are gathered the works I found being related with the subject of trying to describe certain structure of the game in an adequate manner:

- **Stackelberg Games**

  Stackelberg games, first introduced by Stackelberg ('Market Structure and Equilibrium', 1934), are originally two-player games where each of the players has only one action. There is a 'leader' of the game, a player who acts first; and a 'follower', the other player that acts after the 'leader'. Both players commit a certain amount of resources when performing their action. In the end each player gets a payoff dependant on the amount of resources committed during the game. These games also have a restriction which is quite similar to the restriction we have laid on information sets in Etensive Form^{+} games: 'the leader' must act publicly. That is, the second player observes first player's actions. The problem is, most of these games, even if defined for n players, are one-round games meaning that each player acts once.

- **Lossless Abstraction of Extensive Form Games**

  Another interesting class of works related with the subject of this thesis are lossless abstractions. Beginning with a classical publication of Thompson (15) in 1952, continuing with refinements of his results (1) and modern techniques like 'Gameshark' (3) used when trying to abstract huge games like Poker to smaller games. This makes the task of solving big games much easier as we can solve a (sometimes much) smaller game in order to get a solution for a bigger game, instead of solving the big game.

- **Polynomial Games**

  Yet another interesting approach (introduced in 1950 by Dresher, Karlin and Shapley (2)) to games is to assign every possible action some value (or enumerate these strategies) and then define resulting payoffs as a polynomial function of the values of actions that were made. (12)

10

Conclusion

In this work I propose Extensive Form$^+$, an extension of Extensive Form that allows us to describe structures that form payoffs in games more adequately than Extensive Form does. I also propose a notion of strategy for that extension and arguments for when and why this notion can be considered to be acceptable. Finally, I propose a solution concept for games described in the new form and prove that such a solution exists for every game in Extensive Form$^+$.

As it was already mentioned in the introduction, this thesis was mostly inspired by a game of Poker. Even though the cute result of this work doesn't allow us to get any advantage in solving this beautiful game yet, I hope it is a step in the right direction.

References

[1] Giacomo Bonanno. **Set-theoretic equivalence of extensive-form games**. *Internat. J. Game Theory*, **20**(4):429–447, 1992. 50

[2] M. Dresher, S. Karlin, and L. S. Shapley. **Polynomial games**. In *Contributions to the Theory of Games*, Annals of Mathematics Studies, no. 24, pages 161–180. Princeton University Press, Princeton, N. J., 1950. 50

[3] Andrew Gilpin and Tuomas Sandholm. **Lossless abstraction of imperfect information games**. *J. ACM*, **54**(5):Art. 25, 30 pp. (electronic), 2007. 50

[4] John C. Harsanyi. **Games with incomplete information played by "Bayesian" players. I. The basic model**. *Management Sci.*, **14**:159–182, 1967. 7

[5] Harold W. Kuhn. **Extensive games and the problem of information**. In *Contributions to the theory of games, vol. 2*, Annals of Mathematics Studies, no. 28, pages 193–216. Princeton University Press, Princeton, N. J., 1953. 4, 15, 17

[6] Peter Bro Miltersen. **A near-optimal strategy for a heads-up no-limit Texas Holdem poker tournament**. In *In International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, 2007. 2

[7] Peter Bro Miltersen and Troels Bjerre Sørensen. **Computing sequential equilibria for two-player games (extended abstract)**. In *Proceedings of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 107–116, New York, 2006. ACM. 48

[8] Roger B. Myerson. *Game Theory: Analysis of Conflict*. Harvard University Press, Cambridge, MA, 1991. 13

[9] John F. Nash, Jr. **Equilibrium points in n-person games**. *Proc. Nat. Acad. Sci. U. S. A.*, **36**:48–49, 1950. 16, 17

[10] John F. Nash, Jr. **Non-cooperative games**. *Annals of Mathematics*, **2**(54):286–295, 1951. 16, 17

[11] Noam Nisan, Tim Roughgarden, Éva Tardos, and Vijay V. Vazirani, editors. *Algorithmic game theory*. Cambridge University Press, Cambridge, 2007. 8

[12] Pablo A. Parrilo. **Polynomial games and sum of squares optimization**. In *Proceedings of the 45th IEEE Conference on Decision and Control, San Diego, pp. 2855-2860*, 2006. 50

[13] Andres Perea, editor. *Rationality in extensive form games*. Springer, 2001. 20

[14] Hans Peters. *Game Theory: A Multi-Leveled Approach*. Springer-Verlag, Berlin, 2008. 16

[15] Frederick B. Thompson. **Equivalence of games in extensive form**. **14**, 1952. 50

[16] John von Neumann and Oskar Morgenstern. *Theory of Games and Economic Behavior*. Princeton University Press, Princeton, New Jersey, 1944. 2, 4

[17] Bernhard von Stengel. **Efficient computation of behavior strategies**. *Games Econom. Behav.*, **14**(2):220–246, 1996. 48