Master's thesis

Czech
Technical
University
in Prague

F3

Faculty of Electrical Engineering
Department of Computer Science

Stackelberg Extensive-Form Correlated Equilibrium with Multiple Followers

Jakub Černý

Program: Open Informatics
Field: Artificial Intelligence

May 2016
Supervisor: Mgr. Branislav Bošanský, Ph.D.

Acknowledgement / Declaration

I would like to thank my supervisor, Mgr. Branislav Bošanský, PhD., for giving me the opportunity to work on this project, for his patient guidance, much appreciated encouragement and frequent helpful critiques, as he guided me through the process of writing this thesis.

I would especially wish to thank my family for the love, support, and continual encouragement I have gotten throughout my study. In particular, I would like to thank my parents and my sister. I would not be able to stay focused without the constant reminders of Naďa – the reality is always one step ahead of the model.

I also wish to acknowledge the help provided by Sven Bednář, Hayk Divotyan, Tomáš Flek, Adam Klsák, Tomáš Slach and Míša Šubrová, my friends who I continuously asked for opinions on individual chapters and they contributed by many valuable insights.

I hereby declare that I have written the submitted thesis myself and I quoted all used sources of information in accord with Methodical instructions about ethical principles for writing academic theses.

....
Jakub Černý

In Prague, May 27, 2016

Prohlašuji, že jsem předloženou práci vypracoval samostatně, a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

....
Jakub Černý

V Praze, 27. května, 2016

Abstrakt / Abstract

Tato práce formalizuje Stackelbergovo korelované ekvilibrium s vícero následovníky v hrách reprezentovaných v extenzivní formě. V tomto ekvilibriu se vedoucí hráč zavazuje, že bude hrát strategii, kterou veřejně oznámí. Tato strategie je reflektována dalšími hráči, kteří na ni reagují, jak nejlépe mohou. Navíc je vedoucí hráč schopen koordinovat hru pomocí série doporučených akcí zaslaných ostatním hráčům. Každá taková akce je ale následovníkovi skryta, dokud nedosáhne stavu, ve kterém může být vykonána. Práce ukazuje, že v extenzivních hrách s vícero hráči lze toto ekvilibrium popsat lineárním programem s polynomiálně mnoha nerovnostmi, avšak exponenciálně mnoha proměnnými. Navíc tato práce dokazuje, že v případě, že P $\neq$ NP, není nalezení takového ekvilibria v polynomiálním čase možné, kdykoli se hry účastní více jak dva následovníci, či pokud hra obsahuje náhodné události. Algoritmus počítající Stackelbergovo korelované ekvilibrium byl naimplementován a odtestován na náhodně generovaných hrách. Výsledky experimentů ukazují, že výpočetní čas roste exponenciálně ve velikosti herního stromu.

Klíčová slova: Teorie her; Multiagentní systémy; Extenzivní forma; Korelované ekvilibrium; Stackelbergovo ekvilibrium; Stackelbergovo korelované ekvilibrium.

Překlad titulu: Korelované Stackelbergovo equilibrium v sekvenčních hrách s vícero následovníky

This thesis formalizes the Stackelberg extensive form correlated equilibrium (SEFCE) with multiple followers for extensive games with perfect recall. In this scenario, the leader commits to a strategy that is observed by the followers that play a best response. Moreover, he is able to coordinate the course of the game through a series of moves recommended to the players. Each move is revealed to a follower when he reaches the information set where he can take that move. The thesis shows that in multi-player extensive games with perfect recall, the linear program describing SEFCE has polynomial number of constraints, but number of variables is exponential. Moreover, it proves that unless P = NP, the equilibrium can never be found in polynomial time in games with more than three players, or if the game contains chance nodes. The algorithm computing SEFCE was implemented and evaluated on randomly generated games. The experimental results show that the computation time grows exponentially in a size of a game tree.

Keywords: Game theory; Multiagent systems; Extensive form; Correlated equilibrium; Stackelberg equilibrium; Stackelberg extensive-form correlated equilibrium.

/ Contents

Tables / Figures

Tables

Figures

Chapter 1
Introduction

Games and puzzles have fascinated the human society since its early origins. Might it be the immersive spirit of competition, the eternal circle of winning and losing, which encourages humans across all civilizations to spend endless time striving to become the masters of various games. In fact, games accompanied the human kind throughout its own evolution, being an integral part of social interaction between individuals in cultures separated in time and space. And it is not just the humanity, which could not resist the charm of challenging the uncertainty through the physical and mental skill. Also other animals were observed to play games like hide and seek. For some reason, games seem to be written deeply in the genes of all living beings.

Together with the number of different games grew also the curiosity of mankind how to solve the games. The fundamental challenge became to recognize rational responses in various scenarios. The search for the ways to win led to the mathematical formalism of *noncooperative game theory*. This theory tries to formally define games and identify reasonable behavior optimizing the probability of victory. However, the solution concepts may differ greatly, based on the disparity of information provided to participating players. The foundations were laid when the *Nash Equilibrium (NE)* [38] was defined, implying that every game contains a set of strategies for each player so that no one profits from altering his strategy. The Nash equilibrium was later refined several times to overcome the perceived flaws, but all these concepts are closely related. The more general solution concept called *Correlated Equilibrium (CE)* [2] introduced the external events which can help the players to cooperate. Nevertheless, the roles of the players are still considered symmetric. On the other hand, a completely different concept was proposed in *Stackelberg Equilibrium (SE)* [51], when one of the players has an advantage to publicly announce his strategy, while the other players observe this strategy and play as best as they can.

The advent of computers changed the game theory from a mathematical concept to a computable solution. In late 40', the academic computer scientists began designing simple games either as a part of their research or solely for their own amusement. For them, computers provided a computational power for computing the equilibria even in large games – a process which could be hardly ever done manually. This breakthrough encouraged the effort to mathematically formalize many processes in the human society as games. The main applications remain in computational economics, however, any multiagent system with limited resources, where the agents are forced to interact, can be modeled as game. Such systems can include, but are not restricted to, the examples like:

- *Security of dynamic traffic light control* [32]. The complex large-scale transport systems with many remotely controllable intersections present in modern urban centers require effective design of its traffic control systems. Their deployment is fundamental, as they offer significant gains in traffic flow especially during rush hours. Even though dynamic and adaptive in nature, their dependence on external sensors can be exploited to substantially increase congestion. The optimal placement of sensors can ensure robustness against data loss while maintaining system control.

- *Threat detection in computer networks* [57]. In large computer networks with multiple valuable computers of differing importance, the defender has to optimally allocate resources to select and inspect packets to detect potential threats. Meanwhile, an attacker tries to evade the detection and infiltrate the targets by sending malicious packets from several entry points of the network.
- *Optimal honeypot selection* [44]. A decoy computer systems called honeypots are commonly used in network security to waste the time and resources of attackers and to analyze their behavior. The designer of the network has to decide how to design the honeypot systems and also how to locate the honeypots strategically in the network defense. A mathematical models provide important insights into how should honeypots look like to maximize the probability that a rational attacker will attempt to target it.
- *Patrolling and property protection* [54]. A learning attacker tries to identify the patterns in security and then chooses how to attack the facility being protected. A defender schedules randomized patrols to minimize the risk of being infiltrated.
- *Schedules of ticket inspections* [54]. The provider of public transport assigns officers on patrols to search for fare evaders. He has to take into account the information about the places with high concentration of people and the predicted behavior of evaders. The evaders try to remain undetected.
- *Vaccine design to maximize evasion difficulty* [40]. The modern medicine uses vaccination therapies as a very important part of the prevention of spreading infectious diseases such as HIV or influenza. However, most viruses are capable to build themselves a specific immunity throughout sequence of mutations and therefore escape the effect of vaccination. A model-based interaction enables to design vaccines that make such evasion difficult.
- *Analysis of re-identification risk of anonymous data* [58]. Nowadays, many companies can profit greatly from analyzing personal data in large databases. The organizations seek a way to trade their data while protecting privacy by sharing de-identified data. The concerns persist as various methods demonstrated that such data can be re-identified. It is possible to balance re-identification risk with the value of traded data, taking into account that a recipient will try to re-identify it if its potential gains outweigh the incurred costs.

Many of these real-world applications have already been deployed in their domain. However, the scalability and robustness of created algorithms vary. The hardness of computation can cause significant delays when facing a large amount of data. One of the proposed approaches in order to design faster and more effective domain-independent algorithms is to model the complex situations more precisely. In past years, various specialized solution concepts emerged to describe distinct multiagent systems.

This thesis focuses on computation of an equilibrium called *Stackelberg Extensive-Form Correlated Equilibrium (SEFCE)*. In this scenario, a leading agent commits to a publicly known strategy and coordinates the other players. The coordination is realized through the recommendations to the players, which are the moves that are generated before the game starts. However, each recommended move is assumed to be in a sealed envelope and is only revealed to a player when he reaches the point where he can make that move. The applications of this concept may include for example the situations where a commission of union of states strives to coordinate several independent law enforcement forces, even though the law disallows it to directly control the services of any sovereign state. Such situations can be found in economics, politics or national security. Concrete applications are analyzed in detail in Appendix B. Last but not least, this equilibrium can be also used as highly effective heuristic for computing original Stackelberg equilibrium [11].

1.1 Related work

The Stackelberg Extensive-Form Correlated Equilibrium generalizes the Stackelberg equilibrium in extensive-form games by introducing recommendations to the players. Even though the variant with multiple followers has not yet been researched, the concept is closely related to several other works, especially the original correlated equilibrium.

The correlated equilibrium was first introduced by Aumann as a solution concept in single step games [2]. The significant advantage of this concept is that it is computationally more tractable when compared with Nash equilibrium, which is caused by the convexity of the set of equilibria [17]. The property holds also for multiplayer single step games [41, 21]. The concept was later extended to more general classes of sequential games. The best known is perhaps the *Extensive-Form Correlated Equilibrium (EFCE)*. This equilibrium describes a correlation device which recommends moves to every participating player at the very moment they reach a situation when they can take that move. Algorithms for computing EFCE were first introduced in two-player games [52] and consequently also in games with multiple players and random events [19]. The theoretical results achieved in analysis of EFCE are essential for this thesis. Even more promising property is that the problem of finding one equilibrium is still polynomial even in sequential games.

Another proposed extension of correlated equilibrium is the *Agent-Form Correlated Equilibrium (AFCE)* [13]. This concept is very similar to EFCE. In fact, every EFCE is an AFCE [52]. However, in the agent form of the game, moves are chosen by a separate agent for each information set of the player. Moreover, the set of possible outcomes of AFCE can be generally larger than the set of EFCE outcomes. The computation of one AFCE also remains polynomial.

The first extension of the correlated equilibrium which applies to multistage games is the *communication equilibrium* [37, 14]. In this concept, all players are capable to send inputs to the device, which the author calls communication device. This is the main difference when compared with EFCE, in which the device is considered strictly separated and unreachable by the players.

The second extension of CE in multistage games is the *autonomous correlated equilibrium* [14]. It differs from the communication equilibrium by disallowing the players to make any inputs to the device. However, they still receive the signals at every stage of the game. In the canonical form of autonomous correlated equilibrium, the device recommends at every stage to every player a mapping, which based on the relevant part of his strategy for the appropriate stage selects a suitable move. Each player is therefore aware of the optimal behavior throughout the entire stage and not only locally in every information set as in EFCE or SEFCE.

In [22, 59], the authors introduce a specific structure of extensive game with single initial chance node. A separate disinterested player called „the maven“ replaces the correlation device and his role is to reveal a partial information about the initial chance move to every player. The concept is similar to EFCE and as the authors of EFCE remark, the resulting set of obtainable payoffs can be almost identical [52]. Anyway, the maven is a much stronger correlating factor than the correlation device, as he is able to observe and intervene during the entire gameplay.

In both the Nash equilibrium or correlated equilibrium concepts the strategies are analyzed under the assumption that all players will choose their strategies simultaneously, independently on others. However, in some cases the roles of player can become asymmetric. The solution concept called Stackelberg equilibrium (also known as leader-follower or commitment equilibrium) [51] captures a unique situation when one of the players has some sort of advantage, which enables him to move first. The possible advantage of commitment power is a result well-known in both economic and game-theoretic fields [46, 15, 18]. It was demonstrated that the leadership position can be extremely profitable for the player who can acquire it. The reason is that if one player (called the *leader*) commits to a strategy, the only rational reaction of other players (called the *followers*) is to play their best responses. The expected utility of the leader is hence always at least the same as in the Nash solution. The equilibrium is well-studied and proved to be effectively scalable in many classes of single step games [28, 25]. Moreover, this model is more realistic and algorithms based on Stackelberg solution concept are already deployed in systems closely related to security. The successful applications include airport security [43], coastal patrolling [47] and many others [54]. However, computing Stackelberg equilibrium is often NP-hard when sequential interaction is allowed [30].

The idea of correlated equilibrium with one leading player was first mentioned in the field of simple matrix games [7, 9], but this concept was soon informally extended to sequential games [31]. However, the first formal definition and computing algorithms for extensive-form games were presented in [11, 4]. The authors formulate the equilibrium as a Stackelberg analogue of EFCE. The *Stackelberg correlated equilibrium (SCE)* hence has a specific property – while Stackelberg equilibrium pushes the computational demands up, the convexity of correlated equilibrium suggests the possibility of efficient calculation. In the algorithms presented so far, the more deciding property among these conflicting characteristics proved to be the convexity, as they are polynomial.

1.2 Computational motivation

The main motivation for introducing the SEFCE concept with multiple followers is computational.

- **Knowing that most relevant solution concepts are polynomial, is it possible to compute a Stackelberg Extensive-Form Correlated Equilibrium with multiple followers in polynomial time?**

Unfortunately, an answer to this question is *no* (unless $\mathrm{P} = \mathrm{NP}$). This result is proved in this thesis. Note that the complexity is actually closely related to the size of the game description. The input of any solving algorithm is typically some representation of the sequential game. This representation has to be complete in the sense that it entirely encodes the whole structure including the game tree, the information of players, their possible actions, the chance probabilities (if present) and outcomes. Polynomial (or linear or exponential) complexity always refers to the size of this description. The general non-compact form of the sequential games has typically exponential size. However, the compression of the description is possible in many cases. Unfortunately, the computation of correlated Stackelberg equilibrium with multiple followers does not allow the structure to be represented compactly. The thesis characterizes the complexity of SEFCE in two most general classes of games.

Main result I. *For a two-player, perfect-recall extensive game with imperfect information and with chance moves, the problem of finding SEFCE is NP-hard.*

The existence of chance nodes therefore marks the transition from polynomial complexity to NP-hardness. Interestingly, the same holds for multiple followers.

Main result II. *For an imperfect-information perfect-recall extensive game with more than three players and no chance nodes, the problem of finding SEFCE is NP-hard.*

The second result can not be directly derived from the properties of EFCE, since the complexity of finding maximum-payoff EFCE in games with two correlated players is still polynomial. While the related solution concepts can be computationally easier than Stackelberg correlated equilibrium or even Nash equilibrium for simple matrix games, the situation becomes much more ambiguous when considering sequential games. The reason is their representation can be exponentially large. This thesis confirms that even though the set of SEFCE is a polytope defined by polynomially many inequalities, finding a SEFCE can never (unless P = NP) be done in polynomial time.

1.3 Approach of this thesis

This thesis aims at designing a domain-independent algorithm for the problem of finding Stackelberg Correlated equilibrium in extensive-form games. First, the related concepts are introduced, explained and the algorithms for their computation are described. Second, the SEFCE is formally defined, emphasizing the properties shared with other solution concepts. An already existing algorithm for computing this equilibrium in a relatively narrow class of two-player games is shown to not apply to games with chance nodes. Third, a more general algorithm in games with multiple followers is presented. The existence of the equilibrium in every game is formally proved and the complexity of finding SEFCE in different classes of games is analyzed.

Furthermore, the scalability of the designed algorithm is then examined in various games, including general approaches into solving linear programs.

1.3.1 Overview

The thesis is organized in the following structure:

- The game theory provides the necessary theoretic background for playing games. The basics of its mathematical formalism are formulated in Chapter 2.
- Chapter 3 presents various related concepts and algorithms for finding equilibria in games; together with their properties.
- The existence and complexity of finding SEFCE are proven in Chapter 4. This chapter contains also the description of the designed algorithm.
- Chapter 5 contains the experimental results achieved with the algorithm.
- Finally, Chapter 6 concludes this thesis. It discusses both the theoretical and experimental results and proposes the possible directions for future research.

Chapter 2
Introduction to Game Theory

This chapter formally introduces the fundamentals of game theory. At the beginning it formalizes the central notion – utility, and proceeds with the definitions of basic game descriptions. The structure and the technical background are inspired by [48].

Game theory was originally proposed as a mathematical study of winning strategies in games, but was further developed into a theory describing interaction among independent and self-interested agents. In contrast to the cooperative theory, the basic unit of the non-cooperative game theory is an individual, not a group. The interest of each agent is effectively quantified using *utility functions*. The utility functions are a central concept of utility theory, which studies the measures of preferences over a set of possible outcomes of a given interaction. In game theory, this interaction is the very game.

Every utility function is a mapping from states of the game to real numbers and represents the satisfaction of each player of being located in this state. The utility functions can be regarded as ordinal, when only the preference relation is meaningful; or cardinal, when the increments to satisfaction can be compared across different states. The rational models of game theory assumes the utility functions to be cardinal.

Moreover, the correlation of utility functions of different players is essential. In so-called *zero-sum* (or more generally, constant-sum) games the sum over utilities of the players in each terminal state of the game is always equal to zero (or any given constant). These games are usually easier to analyze than *general-sum* games, where the property of a constant sum is violated in at least one terminal state.

Before the games are formally defined, note, that identifying rational behavior is much easier when considering only one player (which is a subject matter of decision theory), but becomes significantly more complex once more agents interact.

2.1 Normal form

Normal (or strategic) form is a basic type of game representation in single step games. Each player moves only once and actions are chosen simultaneously. This makes the model simpler than other forms at a cost of neglecting sequential decision making.

Definition 2.1. Normal-form game. Every normal-form game (NFG) is a tuple $\mathrm{G} = (\mathrm{N}, \mathrm{A}, \mathrm{u})$, where

- $N = \{1, \ldots, n\}$ *is a set of players;*
- $A = \{A_1, \ldots, A_n\}$ *is a set of sets of actions for each player; and*
- *u is a utility function for each player,* $u_i : A_1 \times \ldots \times A_n \to \mathbb{R}$.

The utility functions in normal-form games are usually visualized as a payoff matrix. The number of dimensions of this matrix is equal to the number of players participating in the game. The elements of the payoff matrix are the tuples of utility values, indexed by the respective actions available to each player.

Strategies can be seen as plans contingency or policy for playing the game. In every situation, player's reaction is defined by his strategy. One option is to choose a *pure* *strategy* $\pi_i \in \Pi_i$, which assigns exactly one action from A_i to player i. On the other hand, a *mixed strategy* $\delta_i \in \Delta_i$ is a probability distribution over Π_i. From the player's perspective, randomizing the decisions can be seen as a belief that he can profit from playing such action. A strategy profile is a tuple of pure strategies $\pi = (\pi_1, ..., \pi_n) \in \Pi$ or mixed strategies $\delta = (\delta_1, ..., \delta_n) \in \Delta$, which completely defines how the game will progress.

		Player 2	
		Cooperate	Defect
Player 1	Cooperate	-1,-1	-4,0
	Defect	0,-4	-3,-3

Figure 2.1. An example of normal-form game [48].

Example Consider a two-player game in Figure 2.1. The depicted payoff matrix describes the *prisoner's dilemma*, a standard game modeling the situation when two members of a criminal gang are arrested and kept in isolated confinements. Both are given an opportunity to betray the other prisoner in exchange for a lesser charge. However, none of them is aware of the choice of his colleague. If they both decide to remain silent and *Cooperate*, each of them serves only an year in prison. On the other hand, if they *Defect*, the charge is 3 years. The combined choices lead to a situation when the traitor is freed and the betrayed prisoner serves 4 years.

2.2 Extensive form

Extensive-form games (EFGs) represent sequential interactions between the players. The structure of EFGs can be visually represented as game trees, with each node representing a different state of the game. Every game-state is uniquely determined by a sequence of moves executed by all players during the gameplay. In every node of a game tree exactly one player acts. An edge from a node corresponds to an action that can be performed by the player who acts in this node. All actions are deterministic, so that they are always correctly executed. EFGs model limited observations of the players by grouping certain nodes into information sets; a player cannot distinguish between nodes that belong to the same information set. In perfect-information games, each information set contains exactly one node, which makes their existence redundant. A special case are concurrent-move games (also called simultaneous), where players act all at once during one round. The EFG model also represents uncertainty about the environment and stochastic events by introducing chance moves of a special Nature player.

Definition 2.2. Extensive-form game. Every EFG is a tuple G = (N, H, Z, A, ρ, u, C, I), where

- *N is a set of players;*
- *H is a set of nodes;*
- *$Z \subseteq H$ is a set of terminal nodes;*
- *A is a set of all actions, $A(h)$ are the possible actions in node $h \in H$;*

- *ρ is a player function, ρ: $H \rightarrow N$;*
- *u is a utility function for each player, $u_i : Z \rightarrow \mathbb{R}$;*
- *C is a probability function for performing a chance action, $C : A \rightarrow [0, 1]$; and*
- *I is a set of information sets for each player.*

The function $\rho : H \rightarrow N \cup c$ assigns each node to a player who is on a move in the node, where c implies that the Nature player chooses an action based on a fixed probability distribution. This probability is denoted as function $C : A \rightarrow [0, 1]$ and is known to all players in advance. Consequently, the probability of reaching node h due to Nature (i.e., assuming that all players play all actions required to reach node h) is defined to be the product of the probabilities of all actions taken by the Nature player in history of h. The function C can be overloaded to denote this product as $C(h)$.

As already mentioned, in some cases the players have only limited information about their true state in the game tree. This so-called imperfect observation of player i is expressed by information sets I_i that form a partition over the nodes $\{h \in H : \rho(h) = i\}$ belonging to player i. Every information set contains at least one node (and singular information sets carry a perfect information) and each node is assigned to exactly one information set. All nodes in an information set of a player are not distinguishable to that player. Therefore, the nodes h in a single information set $I \in I_i$ have the same set of possible actions $A(h)$. Any action a from $A(h)$ uniquely identifies information set I and there cannot exist any other node $h' \in H$ belonging to information set different from I in which a can be performed (i.e., $a \in A(h')$). The overloaded notation $A(I)$ is also used to denote the set of actions which can be played in this information set. The game is said to be a game of *perfect recall* if the players remember the history of their own actions and all information gained during the gameplay.

Similar to the strategic form, players can play pure strategies, which assign one action from $A(I)$ to each information set $I \in I_i$ for player i. Unlike in NFGs, the set of all pure strategies can be reduced by using only relevant pure strategies in each information set. This set of *reduced pure strategies* is denoted Π^*. For example in Figure 2.2, the choice of action in the second information set of player 1 is irrelevant, when assuming the player decided to take action R in his first information set. However, representing an EFG using pure strategies is highly inefficient, because the number of actions in the equivalent normal-form game is exponential in the size of the game tree. This exponential blowup is caused by the inevitability to consider all combinations of actions in every information set for each player.

The mixed strategies in EFGs are again the probability distributions over the set of pure strategies. Moreover, games in extensive form can be also played according to another kind of strategy. The *behavioral strategy* $\beta_i \in B_i$ is similar to a mixed strategy in a sense of repeated one-turn games. But instead of randomizing over the set of pure strategies, behavioral strategy randomizes independently over actions in each information set with preset probability distribution. In games of perfect recall, behavioral strategies allowed their compact representation called *sequence form* [27].

A sequence of actions for player i is a list of actions $\sigma_i \in \Sigma_i$ that lie on the path from root state r to any state $h \in H$. $\emptyset$ is an empty sequence and all sequences leading to an information set I or a node h are denoted $seq_i(I)$ and $seq_i(h)$, respectively. The function $inf_i(\sigma_i)$ is used to obtain the information set in which the last action of the sequence σ_i is taken. For an empty sequence, function $inf_i(\emptyset)$ returns the information set of the root node r. Sequences can be extended by finding feasible actions in the information set to which the particular sequence leads. Formally, for every sequence $\sigma_i \in seq_i(I)$, a set of its extensions is a set $\mathrm{Ext}(\sigma_i) = \{\sigma_i a_j | \ \mathrm{a}_j \in A(I)\}$. The sequence σ_i'

is a prefix of σ_i ($\sigma_i' \sqsubseteq \sigma_i$) if σ_i is obtained by finite number of extensions of σ_i'. The game tree is constructed inductively in the way that for every information set the sequences which lead to it are taken and by their extensions is found a subsequent information set. Every state in every information set is clearly characterized by a combination of sequences of all players, which lead to this state.

Given a behavioral strategy, it is obvious that some sequences will be preferred over others in sense of their likelihood to be played. This concept is called a realization plan of β_i and for each player i it is a function $r_i : \Sigma_i \to [0, 1]$ defined as $r_i(\sigma_i) = \prod_{a \in \sigma_i} \beta_i(a)$. Intuitively, realization plans compute the conditional probability of playing a sequence σ_i when considering a behavioral strategy β_i. However, this realization probability cannot be arbitrary. The following network-flow linear constraints have to be met:

$$\begin{aligned} r_i(\emptyset) &= 1 & \\ \sum_{\sigma_i' \in Ext_i(I)} r_i(\sigma_i') &= r_i(seq_i(I)) \qquad & \forall I \in I_i \\ r_i(\sigma_i) &\geq 0 & \forall \sigma_i \in \Sigma_i \end{aligned} \tag{2.1}$$

The first constraint says that the conditional probability of playing an empty sequence when considering any behavioral strategy is always 1. The second constraint ensures that the realization plans of sequences leading to the states reachable by one action from information set I sum up to the realization plan of reaching set I. This also allows the original behavioral strategy to be possibly recovered afterwards, just from these equations. Finally, the third constraint demands the realizations of all sequences to be nonnegative. It is quite natural, since the realization plans are the probabilities, which are by definition at worst zero.

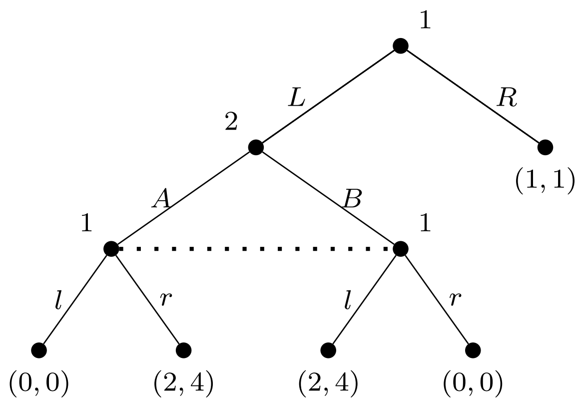

Figure 2.2. An example of extensive-form game, inspired by [48].

The sequence form is much smaller than the normal form, or even reduced normal form. The reason is that every sequence contains only moves of one player along the path from the root. The maximum number of sequences is therefore bounded by the number of nodes in the tree. The realization plan is described by a polynomial number of constraints (one equation for each information set), which uniquely represents any EFG.

In this representation, the *extended utility function* $g_i : \Sigma_1 \times ... \times \Sigma_n \to R$ is defined for each player i as $g_i(\sigma_1 \times ... \times \sigma_n) = \sum_{z \in Z | \forall \sigma_i \in \sigma : seq_i(z) = \sigma_i} u_i(z)C(z)$. If no leaf is reachable with a tuple of sequences σ, a value of g_i is 0.

Finally, the set of opponents of player i is often denoted as $-i$. This notation is frequently used for tuples of strategies restricted to the opponents, such as $\delta_{-i} = \delta \backslash \delta_i$

or $\pi_{-i} = \pi \backslash \pi_i$. In games with only two players, $-i$ is the only opponent of player i. Therefore, his set of sequences is referred as Σ_{-i} and the same notation is used also in functions seq_{-i} or inf_{-i}.

Example Consider the two-player imperfect-information extensive game in Figure 2.2. In this game, player 1 has two information sets – the set including only the top root state, and the set which contains two nodes in the bottom part of the game tree. A dotted line denotes that the states are indistinguishable for player 1. Note that the sets of actions which can be performed in these states belonging to the second information set are identical. Player 1 can be regarded as not informed about an action player 2 took in his information set. On the other hand the second player is always aware where in the game tree he finds himself. This is because his information set contains only a single state. To reach one of the leaves, the players can choose from the following maximal sequences – R, Ll and Lr for player 1 and A or B for player 2. In contrast, the corresponding normal-form equivalent is exponential. The strategies of player 1 are Ll, Lr, Rl and Rr; while player 2 chooses from A or B.

Chapter 3
Solution Concepts

This chapter describes the algorithms for computing both the correlated and Stackelberg equilibrium in finite imperfect-information games with perfect recall and finite utilities, which is assumed to be a *canonical class* of games. Anytime throughout this thesis a *game* is mentioned in the text, it refers to this class, unless expressively stated otherwise. However, the individual subclasses may differ in the number of players and the existence of chance nodes.

Game equilibria are the central concepts of game theory, describing optimal strategy profiles. First, equilibria are successfully used to predict and describe what will happen in strategic interactions between multiple rational agents. Second, every equilibrium is stable. When looking for an optimal reaction, an equilibrium provides a strategy to which (by definition) there is no better response than the equilibrium. An opponent who changes his strategy from the equilibrium is now playing a worse strategy.

The chapter starts with definitions formalizing an optimal gamepley. A player that plays a mixed strategy can gain a various range of outcomes. To evaluate different strategies, he can use an *expected payoff*. An expected payoff for player i is defined as $\mathrm{u}_i(\delta) = \sum_{a \in A_1 \times \ldots \times A_n} \mathrm{u}_i(\mathrm{a}) \prod_{j=1}^{n} \delta_j(\mathrm{a}_j)$. By $\delta_j(\mathrm{a}_j)$ is denoted a probability of player j taking j^{th} action from a. Now it is possible to ask which strategy is the best. Agent's strategy δ_i^* in game G = (N, A, u) is a best response to strategies δ_{-i} if and only if $\forall \delta_i \in \Delta_i$: $\mathrm{u}_i(\delta_i, \delta_{-i}) \leq \mathrm{u}_i(\delta_i^*, \delta_{-i})$. As every player intends to do his best to maximize his utility and considers the decision-making of his opponents, behavior of all agents playing the same game over and over again is evolving. At a certain point of finding their best responses, players realize that changing their strategy would not lead to earn more than with their current decision plan. This concept of balance is called an equilibrium.

Definition 3.1. Nash equilibrium (NE) *Given a game G = (N, A, u) and strategy profile $\delta_{Nash} = (\delta_1, ..., \delta_n) \in \Delta$, players N are in Nash equilibrium if and only if for each player i it holds that δ_i is a best response to δ_{-i}.*

If the current strategy profile allows no one to benefit from changing his strategy, the situation remains stable. It has been proved, that in every game with finitely many players and with finite set of pure strategies, there is at least one Nash equilibrium profile, although it might consist of mixed strategies [38].

Not all equilibria can be computed efficiently. For example, Nash equilibrium in general-sum games of two players can be found by solving a sequence form linear complementarity problem. However, several sets of equilibria in this chapter are represented using linear programs (LPs). The complexity of solving the game is dependent on the size of the LP describing the desired equilibrium. Every LP can be solved in time polynomial in its size [24, 23], even though the most well-known simplex algorithm is faster, but with worst-case exponential time [26].

Proofs of all stated theorems can be found in the respective cited literature.

3.1 Correlated equilibrium

Correlated equilibrium describes the situation when players are given a chance to coordinate according to an external event. This so-called *correlation device* can be imagined as a signaling device (e.g., the traffic lights) helping the players to synchronize. In the canonical representation of correlated equilibrium, the recommendations to the players are the moves, not arbitrarily signals. It was shown that this can be assumed without loss of generality [14]. The very important aspect is that even if the probability distribution that correlation device uses to generate signals is known, each player is not aware of the recommendations given to the other players. All he knows is that they are proposed the best move with respect to the others.

Finding a correlated equilibrium is less difficult than finding a Nash equilibrium, because the sets of correlated equilibrium distributions and payoffs are convex. This property enables the description to be more compact and thus computationally less demanding.

3.1.1 Normal-form CE

In matrix games, players are in correlated equilibrium when given a move according to the correlation device λ, they do not have an intention to unilaterally deviate from the recommended strategy, given his posterior on the recommendations to the other players.

Definition 3.2. Normal-form correlated equilibrium *The distribution* λ *on* Π *is a correlation device of a correlated equilibrium if and only if for all* i*, every* $\pi_i \in \Pi_i$ *with* $\lambda(\pi_i) > 0$ *and every* $\pi_i' \in \Pi_i$*,*

$$\sum_{\pi_{-i} \in \Pi_{-i}} u_i(\pi_i, \pi_{-i}) \lambda(\pi_{-i}|\pi_i) \geq \sum_{\pi_{-i} \in \Pi_{-i}} u_i(\pi_i', \pi_{-i}) \lambda(\pi_{-i}|\pi_i) \tag{3.1}$$

A correlation device λ makes recommendations to the players by randomly picking a strategy profile π^* according to its distribution λ. Then it privately recommends the component π_i of π^* to each player i before the game starts. The equilibrium can be found in polynomial time even in multiplayer games [41, 21].

		Husband	
		LW	WL
Wife	LW	2,1	0,0
	WL	0,0	1,2

Figure 3.1. An example of correlated equilibrium in normal-form game [48].

Example Consider the Battle of Sexes game in Figure 3.1. The game has a unique mixed-strategy Nash equilibrium δ_{Nash} =((1/3, 2/3), (2/3, 1/3)) which guarantees each player an expected payoff of 2/3. However, imagine the players can observe an external event and correlate their actions according to this event. Their strategies are therefore conditioned on this event. For example, if they decide their strategies based on flipping a fair coin, their strategies are extended by the possibility of "head" or "tail". A pair of strategies "WL if heads, LW if tails" forms an equilibrium in this richer strategy space, because whenever one player adopts this strategy, the other one can only lose if he decides to play another. Moreover, the expected payoff of each player is $0.5 \cdot 2 + 0.5 \cdot 1 = 1.5$, which is strictly more than they receive in the mixed-strategy equilibrium in the original game.

3.1.2 Extensive-form CE

The notion of a commitment equilibrium where the leader can adopt a correlated strategy in a sequential game (i.e. SEFCE) is the Stackelberg analogue of the Extensive-Form Correlated Equilibrium (EFCE) introduced in [52]. The definition was shown to be relevant also technically, enabling using techniques of finding EFCE for computing SEFCE [4]. The properties of EFCE are therefore important for better understanding of SEFCE and are explained in this section in detail.

The behavior of correlation device in extensive games differs from the definition of correlation device in normal-form games. In EFGs, the device recommends a move just at the moment an information set is reached. For this reason the recommendations become local and players are less aware of the intended progress of the game. Consequently, also the set of EFCE in an extensive-form game is larger than the set of CE of the equivalent normal-form game. Again, the distribution λ describes a correlation device of EFCE if any rational player behaves according to his recommendations while assuming that

- all other players also follow their recommendations – this is a standard assumption for any equilibrium; and
- when any player decides to deviate from the recommended move, he gets no further information. Consequently, the posterior of the player at all following information sets is equal to that at the last information set before he deviates.

This assumption can be made without loss of generality because any EFCE can be defined using reduced strategies only [52].

Example Consider an extensive-form game with two players depicted in Figure 3.2. This game is described in [52] as a costless variant of a signaling game presented in [50]. In this game a professor (player 2) decides whether to accept a student (player 1) applying for a summer research job. With the same probability, the student is either well-educated (type G) or inexperienced (type B). He sends a costless signal X or Y. Consistently with the structure of the game shown in the Figure, the professor is able to distinguish the signals X and Y, but not the education level of the student. The student is hence capable to impersonate an experienced person and get hired even if his education is poor. In case the professor lets the student to work with him, the payoffs are either $(4, 10)$ for G or $(6, 0)$ for B. Refusal leads to the utility $(0, 6)$ in both cases.

In [52] the authors analyze the correlated equilibria of this game and show that in every equilibrium the professor plays $r_X r_Y$ with a probability of 1. Therefore, he never allows the student to work with him and both players always receive the payoff $(0, 6)$. On the other hand, the situation changes significantly when considering EFCE. The signal for the well-educated student is now hidden from the student with insufficient skills, which effectively prevents any bad student to wittingly overvalue his education. The professor is thus able to evaluate student's knowledge more precisely. Therefore, this situation is advantageous for both of them. For example, a distribution λ_{EFCE} which with the equal probability picks one of the following strategies – $\{(X_G X_B, l_X r_Y), (X_G Y_B, l_X r_Y), (Y_G X_B, r_X l_Y), (Y_G Y_B, r_X l_Y)\}$ – is an EFCE. Note that to prohibit any insidious student from impersonating a good student, λ_{EFCE} recommends the correct signal exactly in one half of cases. Otherwise, he could do exactly the opposite of what he is given and λ_{EFCE} would not be an EFCE. The expected payoff of this equilibrium is $(3.5, 6.5)$, which is strictly more than each of them obtains in the correlated equilibrium.

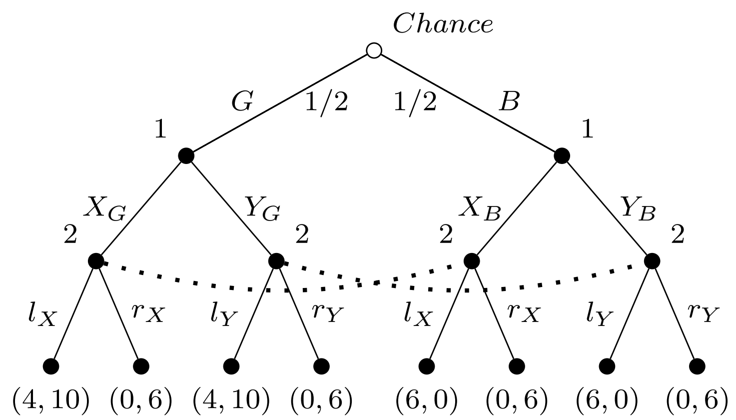

Figure 3.2. Signaling game with costless signals (X or Y) for player 1 [52].

3.1.2.1 Two-player games

When a game has only two players and no chance moves, for every selected reference sequence, the information set is uniquely determined by the player's own history path and the reference sequence. This is the reason why sequence form can be used to compactly describe EFCE in this class of games.

Definition 3.3. Relevant sequences [52] *A pair of sequences* (σ_1, σ_2) *is termed relevant if and only if* $\exists i \in \{1, 2\}$ *either* $\sigma_i = \emptyset$ *or* $\exists h, h' \in H, h' \sqsubseteq h; \sigma_i = seq_i(h) \wedge \sigma_{-i} = seq_{-i}(h')$.

The set of sequences of $-i$ which form a relevant pair with σ_i is denoted $rel(\sigma_i)$. Informally, the sequences are relevant when decisions at the information sets reachable by one of the sequences can affect the decisions at the information sets reachable by the other sequence. This happens exactly when the information sets are connected. In two-player games, if one information set precedes another, the opposite can not hold. The relation of "preceding" is hence antisymmetric even for information sets of different players. This is not true when considering games with chance nodes or strictly more than two players. Now it is possible to extend the constraints for realization plans (2.1) to consistency constraints for joint probabilities of pairs of sequences, which define what is called a correlation plan. These constraints apply only to mutually relevant information sets, where the consistency of recommendations can be violated.

Definition 3.4. A correlation plan [52] *is a partial function* $p : \Sigma_1 \times \Sigma_2 \to \mathbb{R}$ *so that there is a probability distribution* λ *on the set of reduced strategy profiles* Π^* *so that for each relevant sequence pair* (σ_1, σ_2)*, the term* $p(\sigma_1, \sigma_2)$ *is defined and fulfills* $p(\sigma_1, \sigma_2) = \sum_{(\pi_1, \pi_2) \in \Pi^*} \lambda(\sigma_1, \sigma_2)$ *where* π_1, π_2 *prescribe playing all of the actions in* σ_1 *and* σ_2*, respectively.*

The correlation plans describe a joint realization probability $p(\sigma_1, \sigma_2)$ that a pair of sequences (σ_1, σ_2) is recommended to the two players. In [52] the authors proved that correlation plans are sufficient to characterize the set of EFCE in two-player games with no chance nodes.

Theorem 3.5. Extensive-form correlated equilibrium in two-player game without chance moves [52] *The distribution λ on Π^* is a correlation device of an EFCE if and only if the respective correlation plan p for all players $i \in \{1, 2\}$ satisfies*

$$
\begin{aligned}
p(\emptyset, \emptyset) &= 1; \quad 0 \le p(\sigma_1, \sigma_2) \le 1 \\
p(seq_i(I), \sigma_{-i}) &= \sum_{a \in A(I)} p(seq_i(I)a, \sigma_{-i}) \quad \forall I \in I_i, \forall \sigma_{-i} \in rel(\sigma_i) \\
v(\sigma_i) &= \sum_{\sigma_{-i} \in rel(\sigma_i)} p(\sigma_{-i}, \sigma_i) g_i(\sigma_{-i}, \sigma_i) + \\
&\quad + \sum_{I \in I_i;\, seq_i(I) = \sigma_i} \ \sum_{a \in A(I)} v(\sigma_i a) \quad \forall \sigma_i \in \Sigma_i \qquad (3.2) \\
v(I, \sigma_i) &\ge \sum_{\sigma_{-i} \in rel(\sigma_i)} p(\sigma_{-i}, \sigma_i) g_i(\sigma_{-i}, seq_i(I)a) \ + \sum_{I' \in I_i;\, seq_i(I') = seq_i(I)a} v(I', \sigma_i) \\
&\quad \forall I \in I_i, \forall \sigma_i \in \bigcup\nolimits_{h \in I} rel(seq_{-i}(h)), \forall a \in A(I) \\
v(seq_i(I)a) &= v(I, seq_i(I)a) \quad \forall I \in I_i, \forall a \in A(I)
\end{aligned}
$$

First two constraints are based on the formulation of realization plan constraints (2.1). The following constraint ensures that v_{σ_i} is a representation of an expected payoff of the player when he plays σ_i, assuming he follows his recommendations. The constraint consists of two parts – the first sum computes the expected utility of the leaves reached by playing according to σ_{-i} and σ_i, the second sum adds the contribution of the expected utility of information sets reachable by all the extensions of σ_i. The next constraint guarantees that the expected payoff $v(I, \sigma_i)$ is the maximum over all possible sequences leaving the information set I (denoted as $seq_i(I)a$ for all possible actions $a \in A(I_i)$) after the player is recommended to play σ_i. Finally, the last constraint forces the move which is recommended to player i in the information set I to be optimal.

Both the number of constraints and the number of variables in Theorem 3.5 are polynomial in the size of the tree, so any EFCE in this class of games can be computed in polynomial time. However, the authors remark that the problem of finding maximum-payoff EFCE seems to be an example of a game-theoretic solution concept where the introduction of chance moves marks the transition from polynomial-time solvability to NP-hardness. Consequently, finding EFCE in more general classes of games is much harder. In [52], the authors demonstrate it on a real example of a constructed game tree with chance nodes. They show that the consistency constraints (2.1) of realization or correlation plans used also in the linear program (3.2) are not sufficient to completely characterize the convex hull of pure (and reduced) strategy profiles. It means that there exists a distribution on sequence pairs $\hat{p}$ that satisfies the conditions (2.1), but is not a convex combination of pure strategy pairs. The moves which are recommended to players according to $\hat{p}$ are not consistent and therefore cannot be a part of any EFCE. The sequence form is hence not suitable for any game with chance nodes or strictly more than two players.

3.1.2.2 Multi-player games

In [19], the author points out that not only a sequence form, but generally no compact representation like sequence form is about to be expected when aiming to compute an EFCE in more general classes of games. In perfect-recall games of two players, the structure of information sets significantly differs from games with chance nodes or multiple players. The imposed restrictions enable the recommendations to be generated uniquely for each information set, while maintaining the compact description. However, in more general classes of games, the consistency constraints satisfied by a sequence form become only a necessary condition. Consequently, the size of the game representation used for describing EFCE is always asymptotically equal to the size of an equivalent normal-form game. There is also no correlation plan.

On the other hand, the compact representation of games with two players require to compare every pair of preceding information sets, regardless on whom they belong to. In the multi-player case, it is sufficient to compare only those sets where the same player acts. The preceding relation is altered to suit the game representation based on pure strategies, which is introduced in [19] to describe the set of EFCE.

Definition 3.6. Agreeing strategy [52] *A strategy $\pi_i \in \Pi_i$ agrees with a sequence σ if and only if $\forall a \in \sigma\ \exists I \in I_i;\ \pi_i(I) = a$. A partial strategy profile π_J where $J \subseteq \{1,..,n\}$ agrees with a node $h \in H$ if and only if every $\pi_i \in \pi_J$ agrees with $seq_i(h)$.*

Furthermore, the set of all agreeing strategies for sequence σ or (possibly partial) strategy profiles for node h is denoted $agr(\sigma)$ and $agr(h)$, respectively.

Similarly to the linear program for computing EFCE in games with two players, the conditions for EFCE in multiplayer games can be expressed using inequalities. First, in the equilibrium no player has an intention to deviate from his received recommendations. To consider a deviation, a player i calculates the expected payoff contribution of $a \in A_i$ as a sum of expected utilities from all leaves reachable by playing a.

$$u(a) = \sum_{\substack{t \in Z: \\ a \in seq_i(t)}} u_i(t)C(t) \sum_{\pi \in agr(t)} \lambda(\pi) \tag{3.3}$$

The second constrain then compares the expected payoff contribution of action $a \in A(I')$ with the potential payoff the player is able to obtain in case he deviates from his recommendation at this information set I'. The deviation will affect the expected utilities in all subsequent information sets. The optimal expected payoff at information set I (under the assumption the player is recommended move $a \in A(I')$ where I' precedes I) is the maximum of the utilities the player expects for actions $b \in A(I)$.

$$\begin{aligned} v(I,a) \geq & \sum_{\pi_i \in agr(seq_i(I')a)} \sum_{\substack{t \in Z: \\ seq_i(I)d = seq_i(t)}} \sum_{\pi_{-i} \in agr(t)} u_i(t)C(t)\lambda(\pi_i, \pi_{-i}) \\ & + \sum_{\hat{l}: seq(\hat{l}) = seq_i(I)b} v(\hat{l}, a) \end{aligned} \tag{3.4}$$

Note that by the definition, once the player decides not to follow the recommendations, he starts to ignore every further signal. Equivalently, he may as well not receive the recommendations any more. Finally, the last equation makes sure that the expected payoff contribution and the optimal expected payoff of every move $a \in A(I)$ is equal.

$$u(a) = v(I,a) \tag{3.5}$$

If there exists a probability distribution λ on pure strategy profiles that fulfills the presented constraints, it clearly represents an EFCE. Before the game starts, the correlation device picks one of the strategy profiles according to the equilibrium distribution and privately recommends the appropriate moves to every player once they reach a state where they can take that move. Based on the assumption and the presented inequalities, no player has an intention to deviate and hence the distribution λ describes an EFCE. Formally, this fact is summarized in the following theorem.

Theorem 3.7. Extensive-form correlated equilibrium in multi-player game [19] *A probability distribution λ on $\Pi*$ is a correlation device defining an EFCE if and only if it satisfies for all players $i \in \{1, ..., n\}$ the incentive constraints*

$$
\begin{aligned}
u(a) =& \sum_{\substack{t \in Z: \\ a \in seq_i(t)}} u_i(t) C(t) \sum_{\pi \in agr(t)} \lambda(\pi) \qquad \forall I \in I_i, \forall a \in A(I) \\
v(I, a) \geq & \sum_{\pi_i \in agr(seq_i(I')a)} \sum_{\substack{t \in Z: \\ seq_i(I)d = seq_i(t)}} \sum_{\pi_{-i} \in agr(t)} u_i(t) C(t) \lambda(\pi_i, \pi_{-i}) + \sum_{\hat{I}: seq(\hat{I}) = seq_i(I)b} v(\hat{I}, a) \\
& \forall I, I' \in I_i; seq_i(I') \sqsubseteq seq_i(I) \forall a \in A(I') \forall b \in A(I) \\
u(a) =& v(I, a) \qquad \forall I \in I_i, \forall a \in A(I)
\end{aligned}
\tag{3.6}
$$

The number of these constraints that describe the set of EFCE is polynomial in the size of the game tree.

The linear program specified in Theorem 3.7 has an exponential number of variables, since every pure strategy profile is described by exactly one variable; but only a polynomial number of constraints. The respective dual program has therefore only a polynomial number of variables. In [20], the authors exploited the fact that the solution, which is obtainable in polynomial time, is polynomial reducible to a behavioral strategy satisfying the conditions of the primal program. Such reduction is generally not realizable when considering a general linear program. A direct consequence is that despite the exponential representation of the game, one equilibrium can always be found in polynomial time.

3.2 Stackelberg equilibrium

Stackelberg equilibrium [51] models a situation when the roles of players are asymmetric. In this scenario, one of the players (the leader) moves first, while the other players (the followers) observe his strategy and then move sequentially. The expressive power of this concept is suitable for modeling existing real-world situations, because it often occurs in economy, e.g. when the market leader has the power to set the price for items or service; or in security, e.g. when the defender allocates a number of road checkpoints to protect a urban road network.

Despite being the thoroughly useful solution concept, maybe the most burdensome drawback of Stackelberg equilibrium is its high computational complexity in extensive games. The reason is that in contrast with NE or CE, the Stackelberg concept is optimal not even locally, which means that no player has an intention to deviate; but also globally. The leading player strives to adopt a strategy which optimizes his expected utility while knowing the followers will observe it and adapt their own strategies.

In two-player games, it is possible to assume that the follower will adopt a strategy maximizing the payoff of the leader if he does not strictly prefer one of the possibilities.

This property of the follower to act in leader's favor when indifferent is called *compliance* and it is the basis of the commitment theory [33]. In [53], the authors demonstrated that the rational reason for the follower to be compliant is that his outcome is in this case always better or at least equal to that obtained in the Nash equilibrium. Formally, this situation is referred as *Strong Stackelberg equilibrium (SSE)*.

Definition 3.8. Strong Stackelberg equilibrium *The distribution p on Π_l is an optimal leader's strategy of Strong Stackelberg equilibrium if and only if p is a leader's strategy of Stackelberg equilibrium and the followers will break ties in leader's favor.*

However, in the same article they also show that this assumption generally no longer holds for games with more than two players. If the followers are able to coordinate, they can cooperate to achieve the outcome which is best for them. In this case the expected payoff of the leader can be even worse than in the best Nash equilibrium and hence publicly announcing his strategy can be considered as a disadvantage. On the other hand, in the scenarios where the followers cannot coordinate, the leader can still make the followers to adopt the equilibrium which is best for him. Note that in SEFCE, the leader is able to coordinate the followers using the correlation signals, which effectively solves the problem of inconvenienced leader. Once every follower is aware that all other followers are suggested a move consistent with an equilibrium, no one would prefer to deviate from this strategy.

3.2.1 Normal-form SE

For games in normal form [8, 53], computing the Stackelberg equilibrium is straightforward when the class is restricted to two-player games. In this case the follower reacts to the leader's strategy exclusively. When more followers are involved, the situation becomes more complex. Intuitively, the players are in the equilibrium when the leader's expected payoff is maximal possible and the followers cannot obtain higher payoffs by changing their strategy. In the following definition, the set of pure strategies of the leader is denoted as Π_l. If there is only one follower, his set of pure strategies is referred as Π_f.

Definition 3.9. Stackelberg equilibrium *The distribution p on Π_l is an optimal leader's strategy of Stackelberg equilibrium if and only if p maximizes leader's utility and all the followers play their best responses. In 2-player games* [8]*, this can be expressed as*

$$\max_{p,\pi_f} \sum_{\pi_l \in \Pi_l} u_l(\pi_l, \pi_f) p(\pi_l) \tag{3.7}$$

and one inequality for every $\pi'_f \in \Pi_f$

$$\sum_{\pi_l \in \Pi_l} u_f(\pi_l, \pi_f) p(\pi_l) \geq \sum_{\pi_l \in \Pi_l} u_f(\pi_l, \pi'_f) p(\pi_l). \tag{3.8}$$

In [8], the authors proved that the equilibrium can be effectively found in games with two players using the linear program from the definition. However, the problem of finding Stackelberg equilibrium in matrix games with more than three players is NP-hard.

Player 1 \ Player 2	A	B
l	2,1	4,0
r	1,0	3,1

Figure 3.3. An example of Stackelberg equilibrium in normal-form game [8].

Example Consider the matrix game in Figure 3.3. Let player 1 be a leader in this game. First, imagine a situation in which the leader is capable to commit only to a pure strategy. In the Nash concept, he would always try to avoid playing r, because no matter what the second player does, the first player will obtain a higher utility. In contrast, when considering a Stackelberg concept, playing the bottom strategy is exactly what the leader is supposed to do to maximize his expected outcome. The follower is now forced to prefer an action B, which leads to a utility 3 for the leader. Conversely, if the leader decides to commit to the top strategy, the follower would choose A, leaving only a utility 2 for the leader.

Moreover, if the leader commits to a mixed strategy, he is able to obtain even higher payoff. If he decides to play r with higher probability than l, the follower will still prefer B and the expected payoff of the leader is the appropriate proportion of utility 4 and 3. However, in case he commits to placing the equal probability on both his strategies, the follower is indifferent between his strategies. By definition of SSE, the follower adopts a strategy that maximizes the payoff of the leader. The expected utility in this equilibrium is $(3.5, 0.5)$, which is more than the first player is able to obtain in both CE and NE.

3.2.2 Extensive-form SE

Although the security applications of the Stackelberg concept proved to be suitable for many domains, the existing works typically focus on the situations where the players cannot interact in a sequential manner. The first algorithm for computing SSE in extensive games was presented in [8] and its implementation is based on solving multiple linear programs. Specifically, for every pure strategy of the follower it computes the strategy of the leader and then picks the one maximizing his outcome. Number of LPs solved is therefore exponential. Moreover, their implementation uses normal-form games only, thus making it computationally impossible for large EFGs, since the transformation between the forms is also exponential.

The second approach primarily focuses on the Bayesian games, where the leader is playing against one of possible followers with different preferences. The algorithm of [42] formulates the problem as a mixed-integer linear program (MILP). The main advantage of the MILP formulation is in avoiding the exponential Harsanyi transformation of a Bayesian game into a normal-form game.

Even more general algorithm for computing Strong Stackelberg equilibrium in imperfect-information games with two players is the formulation introduced in [3]. The authors exploit both the compact sequence form representation of EFGs and the MILP-based algorithm of [42].

Theorem 3.10. Strong Stackelberg equilibrium in two-player game [3] *The strategies represented as the realization plans r_1 and r_2 describe a SSE if and only if they maximize the leader's expected utility*

$$\max_{p,r,v,s} \sum_{z \in Z} p(z) u_1(z) C(z) \tag{3.9}$$

and satisfy the constraints

$$\begin{aligned}
v(inf_f(\sigma_f)) &= s_{\sigma_f} + \sum_{I' \in I_f : seq_f(I') = \sigma_f} v(I') \\
&\quad + \sum_{\sigma_l \in \Sigma_l} r_l(\sigma_l) g_f(\sigma_l, \sigma_f) \quad \forall \sigma_f \in \Sigma_f \\
r_i(\emptyset) &= 1 \quad \forall i \in N \\
r_i(\sigma_i) &= \sum_{a \in A_i(I_i)} r_i(\sigma_i a) \quad \forall i \in N, \forall I_i \in I_i, \sigma_i = seq_i(I_i) \\
0 &\le s_{\sigma_f} \le (1 - r_f(\sigma_f))M \quad \forall \sigma_f \in \Sigma_f \\
0 &\le p(z) \le r_f(seq_f(z)) \quad \forall z \in Z \\
0 &\le p(z) \le r_l(seq_l(z)) \quad \forall z \in Z \\
1 &= \sum_{z \in Z} p(z) C(z) \\
r_f(\sigma_f) &\in \{0, 1\} \quad \forall \sigma_f \in \Sigma_f \\
0 &\le r_l(\sigma_l) \le 1 \quad \forall \sigma_l \in \Sigma_l
\end{aligned} \tag{3.10}$$

The first constraint in this linear program ensures the follower plays a best response in each information set. As always, the following network flow constraints restricts the realization plans and hence guarantee their well-formedness. Because the best response of the follower in Stackelberg equilibrium is limited to pure strategies, his realization plan is binary. The next constraint effectively enforces the slack variables s_{σ_f} to be zero for the sequences contained in the realization plan of the follower. This is achieved by using large constant M. Finally, the linear program introduces a variable p, which semantically corresponds to the probability of reaching a particular leaf, assuming the players behave according to their realization plans. This probability is equal to a multiplication of the realization plans r_l and r_f, which cannot be in a linear program realized otherwise. The use of p ensures the objective function is formulated as a linear expression.

The analysis of computational complexity of SSE in [30] confirms that in most classes of games (except the perfect-information games with two players and no chance) the problem of finding Strong Stackelberg equilibrium is NP-hard. The formulation used in Theorem 3.10 remains consistent with the analysis, since solving a mixed-integer linear program is also known to be NP-complete. However, in [11], the authors demonstrated that SSE can be calculated using an iterative branch-and-bound algorithm with heuristic based on solving for SEFCE. Even though the problem is still NP-hard, their approach proved to significantly outperform the previous state-of-the-art formulation (3.10) in terms of computational time. Their work was one of the main motivations for analyzing SEFCE with multiple followers.

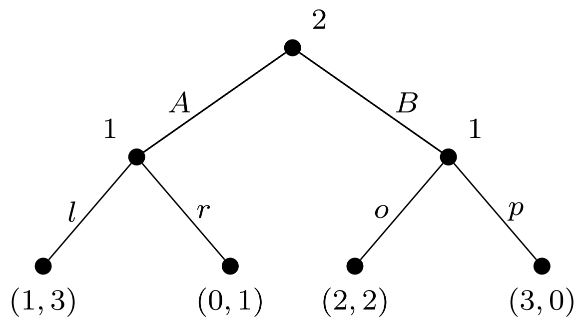

Figure 3.4. An example of Stackelberg equilibrium in extensive-form game [30].

Example Consider an extensive-form game depicted in Figure 3.4. Assume that player 1 acts like a leader in this game, which means that according to the game tree, he moves as a second. In case the roles of the players are symmetric, the solution is a Nash equilibrium where player 1 prefers the moves l and p, which induces player 2 to choose A. The payoffs of the players in this equilibrium are $(1, 3)$. However, once player 1 exploits the power of commitment to pure strategy, he can force player 2 to move to right at the top state by committing to play r and o. The resulting utilities are now $(2, 2)$, which is strictly more than the leader would get in NE. Moreover, suppose that the leader can commit to a behavioral strategy. If he announces that he prefers r in his left state and plays o and p with equal probability in his right state, the follower is indifferent between A and B. By the assumption of Strong Stackelberg equilibrium he therefore takes B, which guarantees the players the expected payoff $(2.5, 1)$.

Chapter 4
Computing Stackelberg Extensive-Form Correlated Equilibrium

This chapter presents the main theoretic results of this thesis. At the beginning it formally defines Stackelberg extensive-form correlated equilibrium in imperfect-information extensive games with perfect recall. It introduces an analysis of algorithm of [11] for computing SEFCE in the elementary class of two-player games without chance moves. The key elements of the transition from a linear program describing EFCE to a linear program describing SEFCE are emphasized. It is shown that the algorithm of [11] cannot be extended any further. In fact, just computing SEFCE in games with two players and chance nodes is by Theorem 4.3 NP-hard. The chapter proceeds by presenting a general linear program for computing SEFCE in games with more than two players and also with the possibility of chance. In addition, the proof is provided to show that SEFCE always exists in every extensive game with perfect recall.

Players	Information	Chance	Complexity	Source
2	perfect	No	$O(\lvert S\rvert\lvert Z\rvert)$	[4]
2	imperfect	No	poly	[11]
2	CM	Yes	poly	[4]
2^+	imperfect	Yes	NP-hard	Theorem 4.3
3^+	imperfect	No	NP-hard	Theorem 4.8
4^+	imperfect	No	NP-hard	Theorem 4.6

Table 4.1. The complexity of finding SEFCE in games with perfect information, concurrent moves (CM) or imperfect information.

The subsequent hardness result proved in Theorem 4.8 claims that the problem of finding SEFCE in games with more than two followers is NP-hard, independently on whether a game contains the chance moves. The obtained complexity results rely on the inspection of proofs of complexity of EFCE presented in [52] and their consecutive reformulation into the formalism of the correlated Stackelberg solution concept. Moreover, a method of pseudo-chance nodes from [30] is adapted to conclude the proof of NP-hardness in three-player games.

Definition 4.1. Stackelberg Extensive-Form Correlated Equilibrium [4] *A probability distribution λ on reduced pure strategy profiles Π^* is called a Stackelberg Extensive-Form Correlated Equilibrium if it maximizes the leader's utility subject to the constraint that whenever play reaches an information set I where any follower can act, this follower is recommended an action a according to λ such that he cannot gain by unilaterally deviating from a in I and possibly in all succeeding information sets given the posterior on the probability distribution of the strategies of other players, defined by the actions taken by the leader and other followers so far.*

The definition suggests that the correlated Stackelberg equilibrium is in fact the "Stackelberg analogue" of extensive-form correlated equilibrium. However, in contrast to the correlated equilibrium, where the signals were recommended by an external correlation device, the signals in SEFCE are the moves which the leader suggests the followers should play. As mentioned in Section 3.2.2, no rational follower would intentionally prefer to deviate from this signal knowing that all other followers are also recommended a move which describes an equilibrium. The leader is hence able to choose the recommendations so that his utility is maximized, while the expected payoffs of the followers are still ensured. The probability distribution λ encodes this (possibly) mixed behavioral strategy of the leader in the equilibrium. In every information set of the leader, he chooses his actions according to λ. This action identifies the strategy profile π^* with non-zero probability $\lambda(\pi^*)$ from which the actions are given to the followers in the subsequent information sets. Similarly to correlated equilibrium, the correlation of the followers gives the leader an opportunity to expect even higher payoff than in strong Stackelberg equilibrium. Consider the following example, which demonstrates the difference between the behavior of leader in SSE and SEFCE.

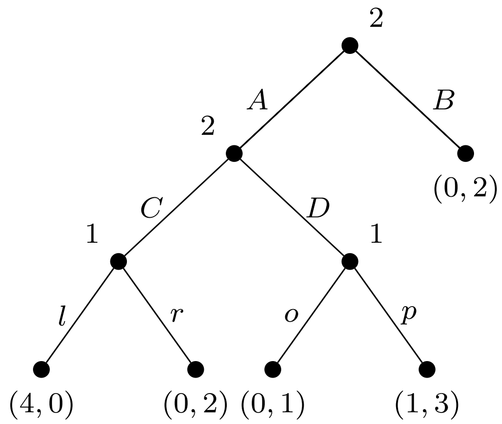

Figure 4.1. An example of correlated Stackelberg equilibrium in extensive-form game [4].

Example Consider an extensive-form game shown in Figure 4.1 and assume that player 1 acts like a leader in this game. In case he is not able to correlate the follower, the solution is a strong Stackelberg equilibrium in which player 1 commits to play a pure strategy l and p in his information sets. The follower is therefore induced to play A and D, resulting in the final payoffs $(1, 3)$. Note that even if the leader plays a strictly mixed strategy, he cannot improve this outcome. The reason is that he has to guarantee the follower the expected utility at least 2 to make him choose action A instead of B. However, the commitment to correlated strategies enables the leader to change it. Consider the distribution over strategy profiles λ_{SEFCE} where the leader with probability 0.5 picks profile (lp, AD) and with the equal remaining probability chooses profiles $\{(ro, AC), (lo, AC)\}$. The probability distribution λ_{SEFCE} is interpreted so that with the equal probability the leader sends either signal C or signal D to the follower. In case the follower receives C, the leader commits to play a uniform strategy in his first bottom information set and o in the second information set. Otherwise the follower receives D and the leader is about to take actions l and p. Note that λ_{SEFCE} is indeed an equilibrium, because the follower does not intend to deviate from his recommendations. Using the signals according to this equilibrium, the leader is able to increase his expected payoff to 1.5 while maintaining the expected utility equal to 2 for the follower.

4.1 Two-player games

As it was shown in the work about EFCE [52], the structure of any two-player perfect-recall extensive game without chance moves can be effectively described using sequence form. This representation leads to the linear program (3.2), which is able to describe and compute extensive-form correlated equilibrium in polynomial time. In [7], the authors discussed the formulation of Stackelberg counterpart of correlated equilibrium in normal-form games. They argue that the Stackelberg correlated equilibrium can be described by dropping the incentive constraints for leader in the linear program and instead adding an objective of maximizing leader's expected utility. It is important to realize that if the incentive constraints for the leading player are added back in, their program just finds the correlated equilibrium that maximizes leader's expected utility. This concept differs from SEFCE significantly. In [11], the authors combined the formulation of EFCE and the approach of [7] to formulate the following linear program computing SEFCE in two-player games without chance moves. They remark (but not prove) that the same approach cannot be used for games with chance nodes.

Theorem 4.2. Stackelberg extensive-form correlated equilibrium in two-player game without chance moves [11] *The distribution λ on Π^* defines a SEFCE if and only if λ is a solution of the following linear program that maximizes leader's expected utility*

$$\max_{p,v} \sum_{\sigma_l \in \Sigma_l} \sum_{\sigma_f \in \Sigma_f} p(\sigma_l, \sigma_f) g_l(\sigma_l, \sigma_f) \tag{4.1}$$

and the respective correlation plan p satisfies

$$\begin{aligned}
p(\emptyset, \emptyset) =&\ 1; \quad 0 \leq p(\sigma_l, \sigma_f) \leq 1 \\
p(seq_l(I), \sigma_f) =& \sum_{a \in A(I)} p(seq_l(I)a, \sigma_f) \quad \forall I \in I_l, \forall \sigma_f \in rel(\sigma_l) \\
p(\sigma_l, seq_f(I)) =& \sum_{a \in A(I)} p(\sigma_l, seq_f(I)a) \quad \forall I \in I_f, \forall \sigma_l \in rel(\sigma_f) \\
v(\sigma_f) =& \sum_{\sigma_l \in rel(\sigma_f)} p(\sigma_l, \sigma_f) g_f(\sigma_l, \sigma_f) + \\
& + \sum_{I \in I_f;\ seq_f(I) = \sigma_f} \ \sum_{a \in A_f(I)} v(\sigma_f a) \quad \forall \sigma_f \in \Sigma_f \\
v(I, \sigma_f) \geq& \sum_{\sigma_l \in rel(\sigma_f)} p(\sigma_l, \sigma_f) g_f(\sigma_l, seq_f(I)a) \ + \sum_{I' \in I_f;\ seq_f(I') = seq_f(I)a} v(I', \sigma_f) \\
& \forall I \in I_f, \forall \sigma_f \in \bigcup_{h \in I} rel(seq_l(h)), \forall a \in A(I) \\
v(seq_f(I)a) =&\ v(I, seq_f(I)a) \qquad \forall I \in I_f, \forall a \in A(I)
\end{aligned} \tag{4.2}$$

The close similarity to the linear program (3.2) is clearly noticeable. Consistent with the assumptions, the constraints remain almost the same (except that the leader's are omitted), only the objective function maximizing the expected payoff is added. The compact sequence representation ensures that the number of variables and constraints of the linear program is polynomial in the size of the tree. Taking into account that any linear program can be solved in polynomial time, finding SEFCE in this class of games is also polynomial. Interestingly, adding chance moves to the game tree dramatically affects the complexity. The reason is the same as in EFCE – the presence of chance nodes causes the recommendations to affect not only the expected contribution of action in a particular sequence, but also across the tree. Even though the corresponding correlation plan is locally consistent (which means it satisfies the network flow constraints), it might not be a convex combination of pure strategy pairs.

Theorem 4.3. *For a two-player, perfect-recall extensive game with imperfect information and with chance moves, the problem of finding SEFCE is NP-hard.*

The proof is given as a reduction from Boolean Satisfiability Problem, in which a formula in conjunctive normal form has a size of each clause limited to at most three literals (also known as 3-SAT). This problem is commonly known to be NP-hard [10]. The similar approach is used to prove hardness of either computing EFCE in [52] or finding optimal plays in possible-worlds models in [6].

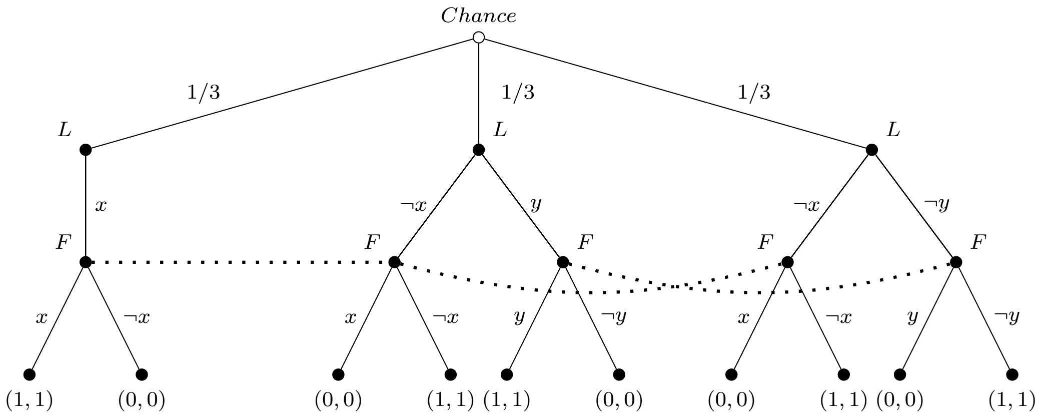

Figure 4.2. A reduction from a 3-SAT instance $\phi = x \wedge (\neg x \vee y) \wedge (\neg x \vee \neg y)$ to a two-player imperfect-information game with perfect recall. The first player acts like a leader, the second player follows. Inspired by a reduction from [52].

Proof. Let ϕ be a boolean formula in conjunctive normal form with n clauses, where each clause has at most three literals. A two-player game $\Gamma(\phi)$ is constructed so that in the root state there is a chance node which chooses each clause of ϕ with uniform probability. Every choice leads to a singleton information set of the follower, who is acknowledged which clause was chosen. His action is to choose one literal of the clause. The leader then decides the boolean value of each variable without knowing if the follower picked the positive or negated variant. If he chooses it correctly, both players get utility 1, 0 otherwise. Apparently, the number of actions in the game tree is at most $10n$ and the number of information sets is at most $4n$, so the reduction is linear in the size of the formula. ϕ is satisfiable if and only if the expected utility of the leader in SEFCE of $\Gamma(\phi)$ is 1.

→ If ϕ is satisfiable, then there exists a truth assignments e which maps each variable of ϕ to the boolean truth value so that ϕ is true in e. If leader behaves in his information sets according to e and the follower chooses the literal which makes the clause true in e, this pure strategy profile is a SEFCE with expected payoff for both players equal to 1.

← Assume ϕ to be not satisfiable and an equilibrium strategy profile λ with expected utility 1. Without loss of generality, let λ_l is pure. If λ_l is mixed, then it has a guaranteed maximal expected outcome 1 in every reachable subtree, which means it might as well be pure and the outcome stays the same. The action a leader recommends to the follower in every information set $I \in I_f$ must be semantically equal to the action the leader plays in $I_l(a)$. If not, the probability of reaching such I is non-zero and it would yield an expected utility strictly less than 1. These recommended actions specify a literal in every clause, which satisfies it. So λ encodes an assignment which makes ϕ true and that is a contradiction.

□

As a consequence, the compact representation of SEFCE cannot be expected when chance moves are allowed. There is no characterization by a polynomial number of linear inequalities, unless P = NP.

4.2 Multi-player games

The uppermost motivation for searching for algorithm which computes SEFCE with multiple followers was computational. The concept of correlated Stackelberg is closely related to the original correlated equilibrium, where the polynomial transition from the two-player case to multi-player case proved to be realizable, first in the class of normal-form games [41, 21]. Moreover, the similar technique was also proved to be suitable for games in extensive form, which was accomplished by [19]. However, the author showed that when facing the situation with strictly more then two correlated players, the compact representation can not be expected. The description is always exponential in the size of the game tree.

To obtain the description of SEFCE, it is possible to follow the same approach as in the two-player case. The constraints related to leader's information sets in the linear program (3.6) are dropped and the criterion maximizing his expected utility is added. The formal description follows.

Theorem 4.4. Stackelberg extensive-form correlated equilibrium in multi-player game *A probability distribution λ defines a SEFCE if and only if λ is a solution of the following linear program that maximizes leader's expected utility*

$$\max_{\lambda,u,v} \sum_{t\in Z} u_l(t)C(t) \sum_{\pi\in agr(t)} \lambda(\pi) \tag{4.3}$$

and satisfies for players $i \in \{1, ..., n\}\backslash l$ the incentive constraints

$$\begin{aligned}
u(c) =& \sum_{\substack{t\in Z:\\ c\in seq_i(t)}} u_i(t)C(t) \sum_{\pi\in agr(t)} \lambda(\pi) \qquad \forall I \in I_i, \forall c \in A(I) \\
v(I', c) \geq& \sum_{\pi_i\in agr(\sigma_I c)} \sum_{\substack{t\in Z:\\ seq_i(I')d=seq_i(t)}} \sum_{\pi_{-i}\in agr(t)} u_i(t)C(t)\lambda(\pi_i, \pi_{-i}) + \sum_{\hat{I}:\sigma(\hat{I})=\sigma_{I'}d} v(\hat{I}, c) \\
& \forall I, I' \in I_i; seq_i(I) \sqsubseteq seq_i(I')\forall c \in A(I)\forall d \in A(I') \\
u(c) =& v(I, c) \qquad \forall I \in I_i, \forall c \in A(I)
\end{aligned} \tag{4.4}$$

Proof. When the follower f receives the signal recommending move c at information set I', the constraints ensure that $v(I', c)$ is the optimal payoff he can obtain from deviating from c. By assumption all other followers obey their recommendations, which means that player f plays his best response. The leader moves first, so the constraints do not apply to him. His expected payoff is maximized by the objective function. The linear program (4.4) with criterion (4.3) hence describes a SEFCE.

□

The system used in Theorem 4.4 completely characterizes the set of all SEFCE. However, this system can be simplified to contain a smaller amount of constraints, similarly to the linear program representing an EFCE [19]. The reason is that for every move $c \in A(I)$, the last constraint makes sure that the expected payoff contribution of c and the optimal expected payoff at the information set I where the acting follower receives c is equal. Therefore, the variables representing $u(c)$ and $v(I', c)$ are redundant and they can be removed. In [19] the constraints are reduced in two steps. First, the variable $v(I', c)$ is replaced with $u(c)$, and second, the variable $u(c)$ is substituted with the expression describing the expected payoff contributions of c – the sum from the first constrain in (4.4). The system (4.4) from Theorem 4.4 is therefore equivalent to the following constraints.

$$
\begin{aligned}
&\sum_{\substack{t\in Z:\\ c\in seq_i(t)}} u_i(t)C(t) \sum_{\pi\in agr(t)} \lambda(\pi) \geq \\
&\qquad \sum_{\pi_i\in agr(\sigma_I c)} \sum_{\substack{t\in Z:\\ seq_i(I')d=seq_i(t)}} \sum_{\pi_{-i}\in agr(t)} u_i(t)C(t)\lambda(\pi_i,\pi_{-i}) + \sum_{\hat{I}:\sigma(\hat{I})=\sigma_{I'}d} v(\hat{I},c) \\
&\qquad \forall I, I' \in I_i; seq_i(I) \sqsubseteq seq_i(I') \forall c \in A(I) \forall d \in A(I') \\
&v(I',c) \geq \sum_{\pi_i\in agr(\sigma_I c)} \sum_{\substack{t\in Z:\\ seq_i(I')d=seq_i(t)}} \sum_{\pi_{-i}\in agr(t)} u_i(t)C(t)\lambda(\pi_i,\pi_{-i}) + \sum_{\hat{I}:\sigma(\hat{I})=\sigma_{I'}d} v(\hat{I},c) \\
&\qquad \forall I, I' \in I_i; seq_i(I) \sqsubseteq seq_i(I') \forall c \in A(I) \forall d \in A(I')
\end{aligned}
\tag{4.5}
$$

The number of constraints in the system (4.5) is polynomial. In fact, for every pair of information sets I and I' of player p such that I precedes I' (which includes also the possibility that $I = I'$), the system contains one constraint for every $c \in A(I)$ and $d \in A(I')$. The expected number of constraints $E[C]$ is

$$
E[C] = \sum_{i\in\{1,\ldots,n\}\setminus l} \sum_{I\in I_i} \sum_{\substack{I'\in I_i:\\ seq_i(I)\sqsubseteq seq_i(I)}} |A(I)| \cdot |A(I')| \sim |P| \cdot E_p[A]^2 \cdot E_p[I]^2, \tag{4.6}
$$

where $E_p[A]$ is an expected number of actions in every information set and $E_p[I]$ is an expected number of information sets of every player. Therefore, the linear program (4.4) for computing SEFCE has an exponential number of variables, but only a polynomial number of constraints.

4.2.1 Existence of equilibrium

Even if the linear program (4.4) is proved to identify a correlated Stackelberg equilibrium, the existence of the equilibrium itself does not follow directly. The polytope described by the constraints could be empty (which means the linear program is unfeasible) or conversely, its volume could be infinite in the direction of the objective function (which means the linear program is unbounded). Without knowing the existence of the equilibrium is inevitable, it is difficult (and perhaps meaningless) to try to analyze its properties. Fortunately, a SEFCE exists in every extensive game with perfect recall.

Observation 4.5. Existence of SEFCE *Every multi-player, perfect-recall extensive game has a SEFCE, which can be constructed by solving a linear program.*

Proof. In [19], the author proved that EFCE in multi-player games with perfect recall always exists, which means that the convex polytope which is defined by the linear program (3.6) and represents the set of EFCE is always non-empty. The concept of SEFCE is a modification of EFCE created so that one of the players is chosen to act as a leader, the criterion which maximizes his expected utility is added and the original linear program describing EFCE is stripped of the constraints related to leader's information sets. The optimization is thus done over a convex polytope which is at least as large as the polytope representing EFCE. The only problem with the existence of SEFCE might thus be that the linear program for computing SEFCE is unbounded. However, this cannot happen since the criterion maximizes a convex combination of leader's possible outcomes, which are (by the definition of the relevant class of games) finite, so the maximization is bounded by the maximum utility the leader can obtain in the game. The leader's expected utility is therefore finite, the maximum is always reached and the equilibrium always exists.

□

Note that simply transform an extensive game into its normal-form representation and then finding the equilibrium is not possible. Similarly to EFCE, also SEFCE is a sequentially revealing equilibrium, while in SCE the leader recommends the followers their whole strategy at once, as soon as the game starts.

4.2.2 Computational complexity

As it was shown in section 4.1, in games with only a leader and one follower SEFCE has a compact description – a linear program with size polynomial in a size of a game. Unfortunately, the existence of chance moves marks a transition from polynomial class to NP-hardness, because no compact description can be expected when chance nodes are added into a game tree. Similarly to EFCE [52], the existence of a compact representation would provide a polynomial algorithm for identifying an equilibrium which maximizes a sum (or any other linear function) of expected payoffs of the players (or in case of SEFCE – of a leader). This would imply that P = NP.

In EFCE, it holds that the set of the equilibria in games with strictly more than two players always has a non-compact representation. However, the situation is different in correlated Stackelberg equilibrium. In this equilibrium the leader is not correlated with the followers, which means that the NP-hardness of maximum-payoff EFCE with multiple players does no directly imply that finding SEFCE in games with only two followers is NP-hard. Conversely, it actually suggests that there might be a polynomial algorithm for computing it. The intuition is justified by considering previous hardness results achieved in analysis of SEFCE, which are listed in Table 4.1. Unfortunately, this section provides an analysis of games with more than three players and shows that computing SEFCE is NP-hard even in these games.

It is more suitable to begin with less general result for games with at least four players. One can observe that the situation is similar to that in proof of Theorem 4.3. The key idea – the structure of the information sets of the follower and the necessary randomization – is cautiously reformulated for the case of games with four players.

Theorem 4.6. *For an imperfect-information perfect-recall extensive game with more than four players, the problem of finding SEFCE is NP-hard.*

The proof is once again based on the reduction from 3-SAT problem. To achieve the essential assumption of the proof of Theorem 4.3, which means that each clause is chosen randomly, the utilities of the follower who replaces the chance node are chosen so that he has an incentive to randomize. The same idea is used in [52] for proving the hardness of EFCE.

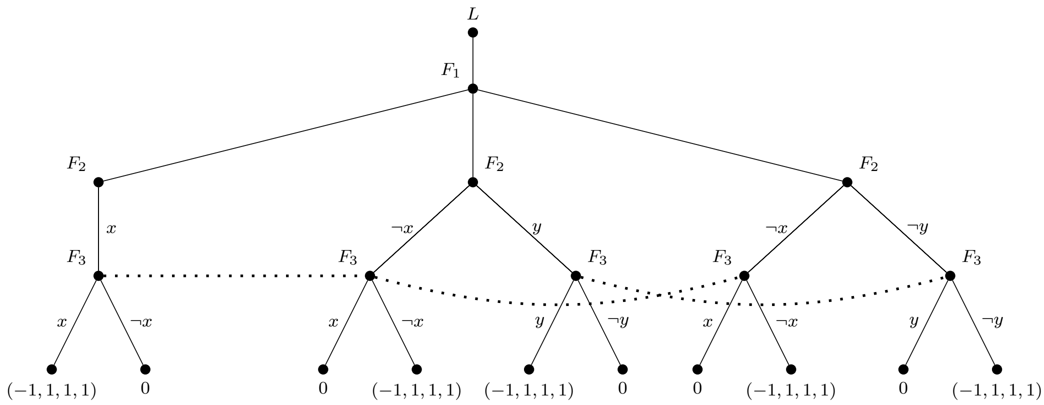

Figure 4.3. A reduction from an unsatisfiable 3-SAT instance $\phi = x \wedge (\neg x \vee y) \wedge (\neg x \vee \neg y)$ to a four-player imperfect-information game with perfect recall. The first player is the correlating leader, the other three players act like the followers. Inspired by a reduction from [52].

Proof. Let ϕ be a boolean formula in conjunctive normal form with n clauses, such that every clause contains at most three literals. A four-player game $\Gamma(\phi)$ is constructed so that in the root state the leader selects his trivial and only possible action. This simplification is possible because his only role in this game is to correlate the players. His action leads to the state of the first follower, who picks one of the clauses of ϕ. Every choice then leads to a singleton information set of the second follower, who is acknowledged which clause was chosen. His action is to choose one literal of the clause. The third follower then decides the boolean value of each variable without knowing if the second follower picked the positive or negated variant. If he chooses it correctly, the leader, the second follower and the third follower get utility 1 while the first follower gets -1. Otherwise, all players gain zero. The first follower acts like an opposing force in this scenario. Apparently, the number of actions in the game tree is at most $10n + 1$ and the number of information sets is at most $4n + 2$, so the reduction is linear in the size of the formula. ϕ is satisfiable if and only if the expected utility of the leader in SEFCE of $\Gamma(\phi)$ is 1.

$\rightarrow$ If ϕ is satisfiable, then there exists a truth assignments e which maps each variable of ϕ to the boolean truth value so that ϕ is true in e. If the leader recommends the actions according to e in the third follower's information sets and the second follower is recommended to choose the literal which makes the clause true in e, the leader is able to ensure himself the maximal expected utility 1. This pure strategy profile is a SEFCE with expected payoff for the leader and the second and third follower equal to 1, independently on the strategy of the first follower, who cannot avoid obtaining the utility -1.

$\leftarrow$ Assume ϕ to be not satisfiable and an equilibrium strategy profile λ with leader's expected utility 1. Then λ_{f_3} must be pure, otherwise this expected utility could not be achieved. The action a leader recommends to the second follower in every information set $I \in I_{f_2}$ must be semantically same to an action which the third follower is recommended to pick in the subsequent information set. If not, the first follower can play an action leading to I and it would yield an expected utility strictly less than 1. Similarly to the previous case, the strategy profile λ is an equilibrium, because no one can get a strictly better utility by deviating. Interestingly, this holds independently on which action is recommended to the first follower in his information set, because the expected utility of the leader significantly restricts the recommended actions in the information sets of other followers. The actions recommended to the second follower describes the literal in every clause which satisfies it. So λ encodes an assignment which makes ϕ true and that is a contradiction.

□

The proof is valid even if only one of the followers has non-singleton information sets. That means the limited information of the players about their location in the game tree does not have to significantly differ from the information provided to players in games with perfect information. The NP-hardness hence affects even games without complex structure of information sets. Furthermore, note that if instead of the state of the first follower the chance node was replaced by a state of the leader (which means the game would have only three players) the logical progression of the proof would fall apart. The nodes of the followers are placed so that the players has to match the literals satisfying each clause with their truth assignments. Therefore, the structure can not be altered in any way. Moreover, the leader would never have an intention to randomize, because he can always make the followers to play a Nash equilibrium which maximizes his expected payoff. This equilibrium would hence violate at least one implication of the equivalence with the condition of satisfiability, independently on whether the leader's utilities are positively correlated, negatively correlated or even uncorrelated.

In [52], the authors demonstrated that the signals which the players receive from a correlation device in EFCE can be inconsistent if the "preceding" relation on the information sets is not antisymmetric and the game is described using a sequence form. This can be easily seen to be violated if there are third player or chance nodes in the game. Therefore, there is no linear program based on the sequence-form representation which can compute a correlated Stackelberg equilibrium in general extensive game. However, this result does not directly imply there is no other compact representation beyond the sequence form, which would permit to find SEFCE effectively.

The key idea for proving that a correlated Stackelberg equilibrium in extensive games cannot be effectively constructed is revealed by inspection of the proof of NP-hardness of SEFCE in games with chance or four players. The proof is based on the fact that each clause of a 3-SAT formula is chosen randomly. Hence, it is necessary to incite the leader to randomize over this set of clauses. Fortunately, this can be achieved using a structure called pseudo-chance node, introduced in [30].

Lemma 4.7. [30] *In Stackelberg extensive-form game it is possible to simulate a chance node with uniform distribution over its N descendants by a tree structure with appropriate utilities for the leader and one follower.*

Proof. Consider a game depicted in Figure 4.4. This game with two players contains $2(N-1)$ internal nodes and $2N-1$ leaves. In an optimal solution, the leader will make the follower to move right everywhere, because letting him move left would result in obtaining a payoff 0. In addition, it is in the interest of the leader get to the rightmost leaf where his utility is maximized. First examine the leader's optimal commitment at the bottom internal node of the tree. He needs to commit to placing the probability at least 0.5 on the left action in order to make the follower move right at the node above. However, it is not advantageous for the leader to commit to move left with the probability higher than 0.5. The reason is his utilities are higher up in the tree, whereas the follower always receives a payoff 1. This strategy makes the follower indifferent in the next higher node and by assumption of SSE, he breaks ties in leader's favor. By similar reasoning, the optimal strategy of the leader in the higher node is to move left with probability 1/3, etc.

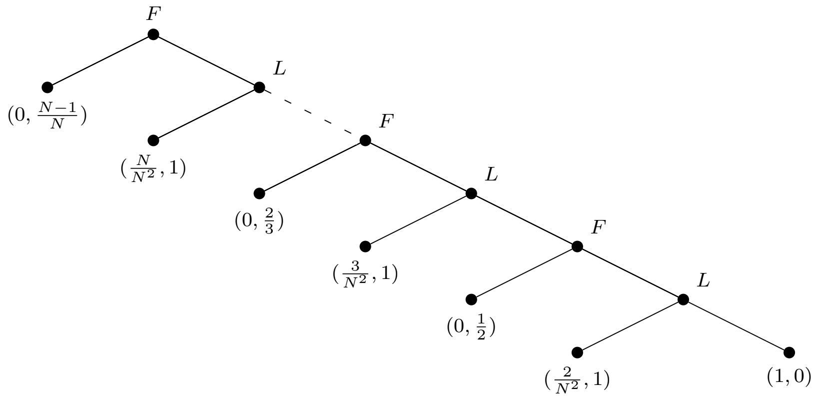

Figure 4.4. A pseudo-chance node used in Lemma 4.7.

Consequently, the optimal strategy β^* of the leader in his i^{th} node from the bottom is

$$\beta^* = (\frac{1}{i+1}, \frac{i}{i+1}). \tag{4.7}$$

Given this behavioral strategy, the probability that the leader will move left exactly at his ith node from the bottom (and not before) is

$$\frac{1}{i+1} \prod_{j=i+1}^{N-1} \frac{j}{j+1} = \frac{1}{N}, \tag{4.8}$$

and the probability that the follower never moves left is

$$\prod_{j=1}^{N-1} \frac{j}{(j+1)} = \frac{1}{N}. \tag{4.9}$$

This structure is thus able to substitute a chance node with uniform probability distribution over all possible actions of the leader.

□

The expected utility of the leader in this strong Stackelberg equilibrium is

$$u_l(\beta^*) = \frac{1}{N} + \frac{1}{N} \sum_{i=2}^{N} \frac{i}{N^2}. \tag{4.10}$$

The important observation is that the expected utility of the leader is strictly higher than any utility the player can get without trying to reach the rightmost leaf. Now it is possible to proceed to formulation of the theorem about hardness of finding SEFCE in games with more than three players.

Theorem 4.8. *For an imperfect-information perfect-recall extensive game with more than three players, the problem of finding SEFCE is NP-hard.*

The reduction from 3-SAT is again used to prove the computational complexity. Moreover, the pseudo-chance node from Lemma 4.7 is required to incentivize the leader to randomize. Recall that the same approach as in Theorems 4.3 and 4.6 is not applicable in case of three players, because the leader would not be forced to commit to strategy which guarantees a uniform distribution over the leaves.

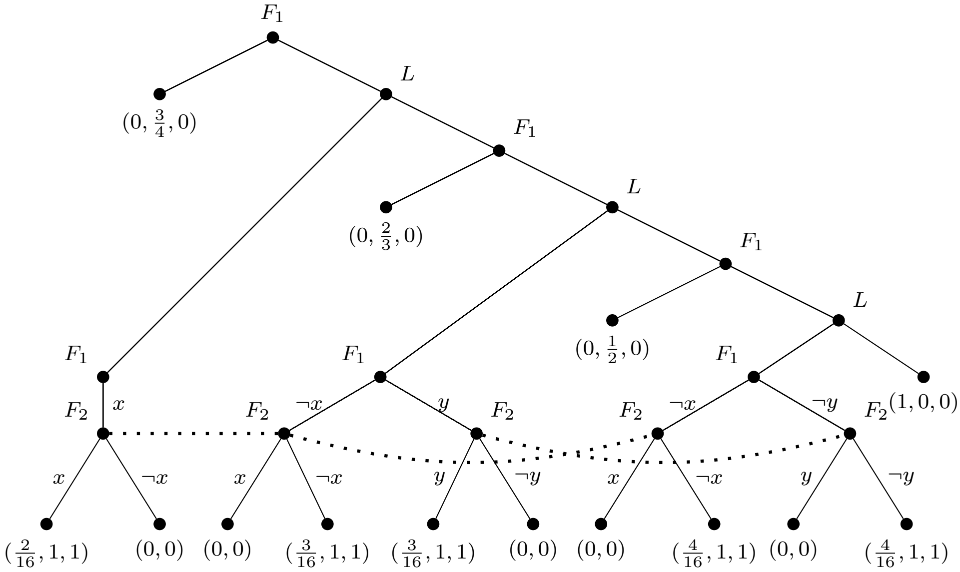

Figure 4.5. A reduction from an unsatisfiable 3-SAT instance $\phi = x \wedge (\neg x \vee y) \wedge (\neg x \vee \neg y)$ to a three-player imperfect-information game with perfect recall. The first player is the correlating leader, while the other two players are the followers. Inspired by a reduction from [52].

Proof. Let ϕ be a boolean formula in conjunctive normal form with n clauses, such that every clause contains at most three literals. A three-player game $\Gamma(\phi)$ is constructed so that in the root is a pseudo-chance node with uniform distribution over $n+1$ nodes. Each of these nodes (except the rightmost one, which stays the same) represents one of the clauses of ϕ. The first follower F_1 is acknowledged which clause was chosen and he has to choose one literal of the clause. The second follower F_2 then decides the boolean value of each variable without knowing if the first follower picked the positive or negated variant. If he chooses it correctly, both followers will obtain the utility 1 and the leader the appropriate utility i/n^2. Otherwise, all players gain zero. The number of actions in the game tree is at most $13n$ and the number of information sets is at most $6n$, so the reduction is linear in the size of the formula. ϕ is satisfiable if and only if the expected utility of the leader in SEFCE of $\Gamma(\phi)$ is $u_l(\beta^*)$ of the pseudo-chance node with $n+1$ equally like outcomes.

$\rightarrow$ If ϕ is satisfiable, then there exists a truth assignments e which maps each variable of ϕ to the boolean truth value so that ϕ is true in e. If the leader recommends the actions according to e in the second follower's information sets and the first follower is recommended to choose the literal which makes the clause true in e, the leader is able to ensure the maximal expected utility 1 for both followers in every subtree under the pseudo-rchance node. By Lemma 4.7, his optimal strategy is thus the same as in the pseudo-chance node itself and cannot be any better because the first follower would otherwise does not want to proceed right in the pseudo-chance node structure. This distribution over the pure strategies is a SEFCE, because no player can obtain more by deviating from this strategy.

$\leftarrow$ Assume that ϕ is not satisfiable. If the leader is about to obtain the same expected payoff as in the pseudo-chance node, he has to attempt to reach the rightmost leaf with the same probability as in the pseudo-chance node. Otherwise, as noted right after the proof of Lemma 4.7, his expected utility is strictly lower. The optimal strategy of the leader is necessarily constructed bottom-up. In every state of the leader, he has to ensure that the expected payoff of the first follower when he moves right in the state above is the same as as if he moves left. Otherwise he would deviate from the recommended move right. Note that his left move leads always immediately to a leaf. Because ϕ is not satisfiable and the game tree is designed so that the followers have to match each clause with the literal which satisfies it in order to obtain the maximum payoff, there exists a subtree T_i encoding a clause i in which the leader cannot any longer guarantee this outcome for the first follower. Denote $u(T_i) < 1$ the expected payoff of the first follower in this subtree. The leader considers to move left with the probability p_L and right with the probability $1 - p_L$. In order to make the follower indifferent, it must hold that

$$u(T_i)p_L + (1 - p_L)\frac{i-1}{i} = \frac{i}{i+1}. \tag{4.11}$$

The leader hence moves left with probability

$$p_L = \frac{1}{(i+1)(u(T_i)i - i + 1)}, \tag{4.12}$$

which is either negative (meaning that he is not able to play an optimal strategy even in the lower nodes) or strictly greater than the probability of going left in the optimal strategy in the pseudo-chance node – recall that it is $1/(i+1)$. The rightmost leaf is therefore reached less often, which means the expected payoff in SEFCE for the leader is strictly less than the expected payoff in the pseudo-chance node.

□

Chapter 5
Experiments

This chapter describes the experiments performed with the linear program for computing SEFCE. The computation is done using several methods for solving general linear programs. Note that this chapter focuses on computing an exact equilibrium, while the design of algorithms for computing an approximated SEFCE is not the aim of this thesis. First, the chapter states the conditions under which the experiments were carried out. It specifies the solving algorithms used for testing and the parameters of individual instances of games – specifically the structure, size, number of players and the role of the leading player. Second, the results are given and discussed.

5.1 Settings

The implementation is done in Java in three steps. First, the extensive-form games are modeled in a domain-independent framework. The game trees have to be represented in pure strategies. Second, a linear program is constructed on top of the representation. Finally, a solving algorithm is used to obtain the solution.

For defining games, the implementation uses Game-Theoretic (GT) Library[1]), a project of Computational Game Theory group from Artificial Intelligence Center at Czech Technical University in Prague. Game-Theoretic Library is written in Java and contains domain-independent implementations of algorithms for solving extensive-form games. It defines a general modular framework for describing games, which provides a unified environment for building game trees. The efficient computation of equilibria requires the games to be described compactly. In Game-Theoretic Library, extensive games are represented using either behavioral strategies or realization plans based on sequence form. However, the design of the library focuses mainly on two-player games. Consequently, as the first step in the implementation of an algorithm which computes SEFCE with multiple followers, this framework has to be modified to suit the purposes of games with multiple players, which are represented in pure strategies. For this purpose, the algorithm first transforms the sequence form into an equivalent normal form of the game. Once the framework was adapted, the linear programs can be generated. Their formulation omit redundant pure strategies by using only reduced ones.

The linear programs computing SEFCE are solved in the IBM ILOG CPLEX Optimization Studio[2]). Cplex is an optimization software package developed mainly for solving integer programming problems and very large instances of linear programming problems using various methods. These methods include either primal or dual variants of the simplex method, the barrier interior point method and several others. The solver is also capable to choose the method automatically, using heuristics based on the analysis of the problem. The algorithm for finding the correlated Stackelberg equilibrium chooses specifically the primal simplex, the dual simplex or the interior-point method.

[1]) `http://jones.felk.cvut.cz/repo/gtlibrary/`
[2]) `http://www-01.ibm.com/software/commerce/optimization/cplex-optimizer/`

The version of software used in the experiments is listed in Table 5.1. The linear programs were generated in Java from a game representation obtained from Game-Theoretic Library and subsequently solved using Cplex's interface called Concert.

Java:	version 1.8.0_91
IBM ILOG Cplex:	version 12.4
Game-Theoretic Library:	version from 10. 4. 2016

Table 5.1. The version of software used in the experiments.

The experiments were performed on a desktop computer with processor Intel(R) Core(TM) i7 CPU 860 @ 2.80GHz and 16GB RAM.

5.1.1 Game domains

To guarantee the robustness of obtained results, the algorithm was tested on *randomly generated games*. The game trees in this domain can be built according to parameters including number of players, existence of chance nodes, maximum tree depth, fixed or flexible branching factor (BF), maximal branching factor, correlation of utilities or random seeds. A random seed initializes a pseudorandom number generator, which influences the structure of the game tree. It alters the number of actions for each player, the utility values, or the observations the players obtain, which hence specify the information sets. This domain is therefore suitable for testing the scalability of the solving algorithms.

Number of players	Chance nodes	Max. depth	Max. BF
2	No	2	2 – 12
2	Yes	2	2 – 12
2	No	3	2 – 6
2	Yes	3	2 – 6
2	No	4	2 – 4
2	Yes	4	2 – 4
2	No/Yes	5	2
3	No	3	2 – 4
3	Yes	3	2 – 4
3	No	4	2 – 4
3	Yes	4	2 – 4
3	No/Yes	5	2
3	No/Yes	6	2
4	No/Yes	4	2
4	No/Yes	5	2
5	No/Yes	5	2
6	No/Yes	6	2

Table 5.2. The configurations of game trees used for experiments.

The 64 configurations of game trees proposed for the experiments are listed in Table 5.2. The first player was chosen as a leader in all cases. The number of actions for each player differs across the information sets, hence the generated game trees are assymetric. The algorithm was tested also in games with only two players, in order to compare the computation times in games either with or without chance nodes; or with or without multiple followers.

The experiments were performed 20 times on each configuration with different random seed (and hence a different game tree structure) to guarantee a statistical significance of the results. The computation was stopped if the running time exceeded 20 minutes. The approximate sizes of selected individual instances are depicted in Figures 5.1 and 5.2.

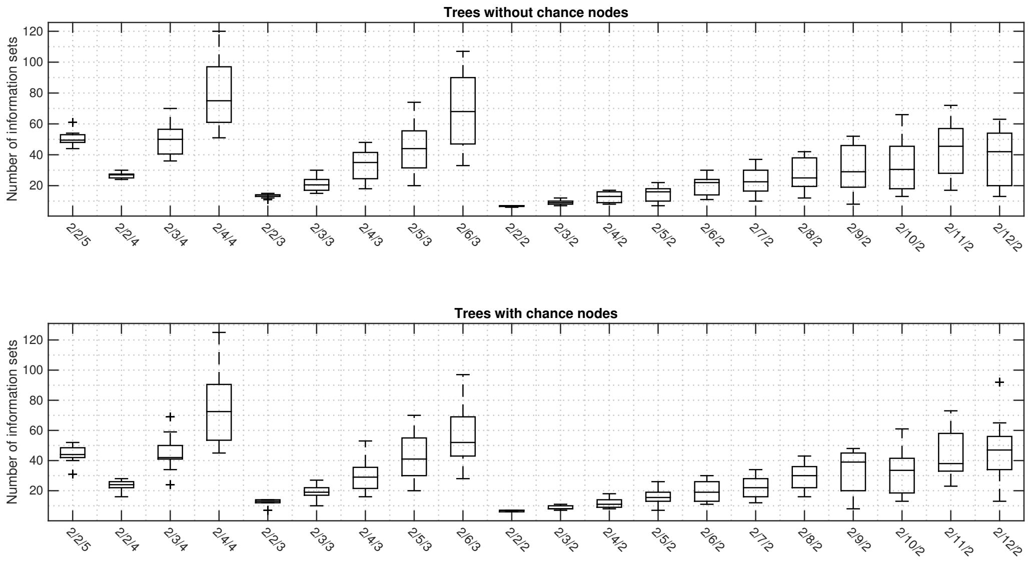

Figure 5.1. Number of information sets in two-player games. The configuration of every instance is encoded as: number of players / branching factor / depth of the tree.

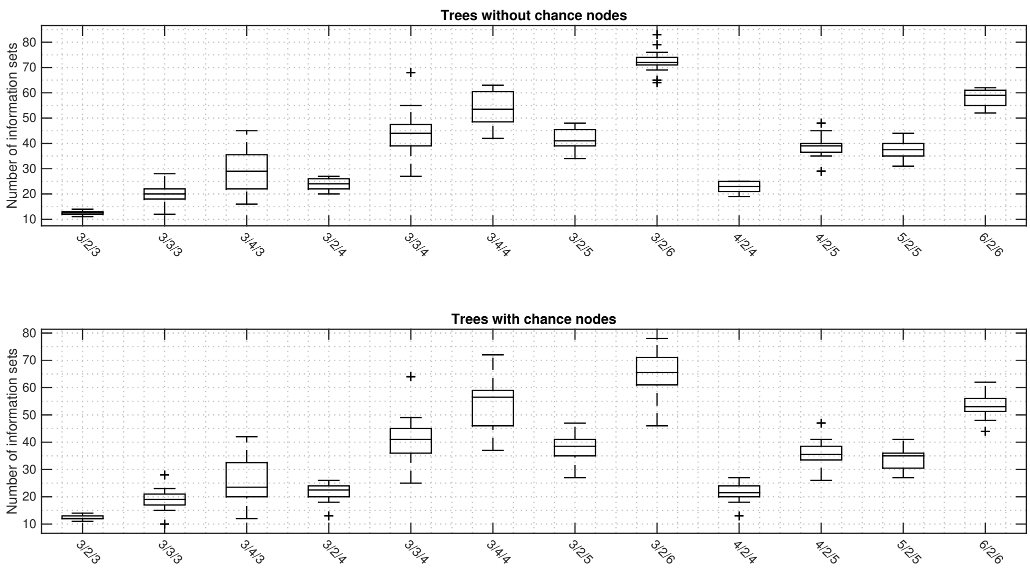

Figure 5.2. Number of information sets in multiplayer games. The configuration of every instance is encoded as: number of players / branching factor / depth of the tree

Furthermore, Figures 5.3 and 5.4 compare the measured generation times of individual game settings. The results indicate that generation times grow exponentially with increasing size of the game trees. In fact, the generation of an appropriate linear program is a crucial part which affects the time of computation of correlated Stackelberg equilibrium the most.

Example Consider a configuration with 3 players, branching factor 4 and depth of the game tree 4 (3/4/4). The corresponding linear program computing SEFCE has approximately $3.1 \cdot 10^6$ variables and 600 constraints, even if the game tree is relatively small and shallow. The size of the respective linear program is hence enormous, as the number of strategies grows exponentially in a number of information sets.

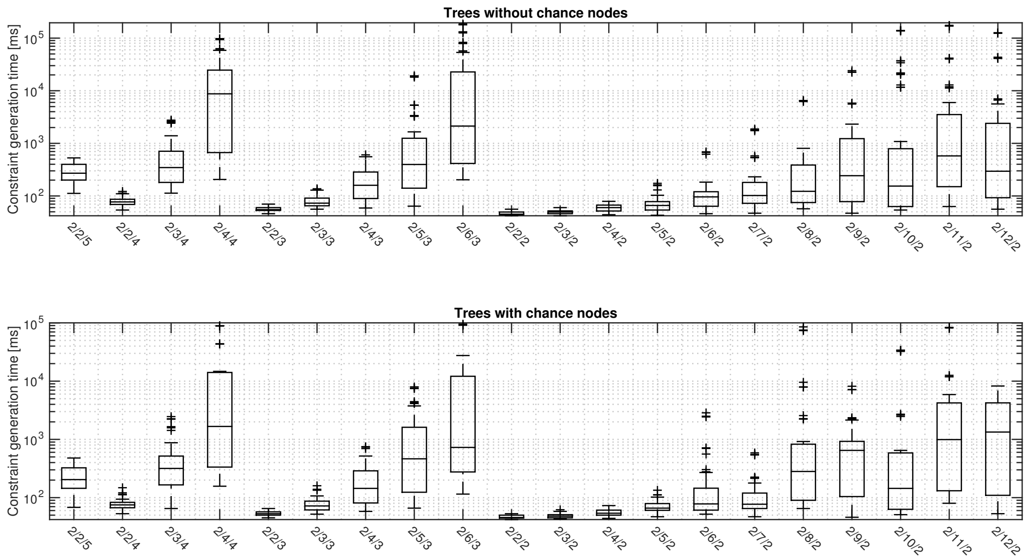

Figure 5.3. Constraints generation time in two-player games. The configuration of every instance is encoded as: number of players / branching factor / depth of the tree.

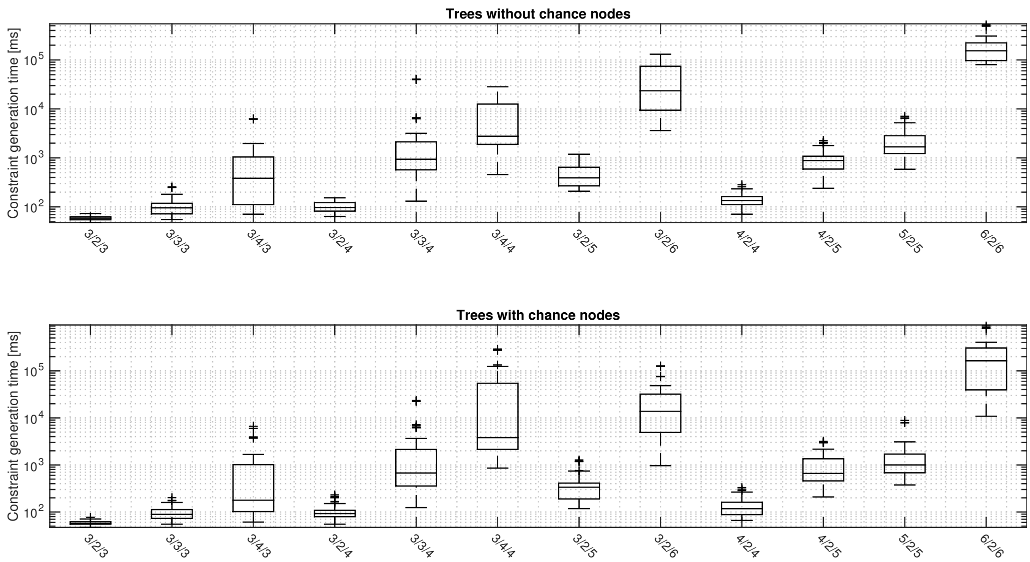

Figure 5.4. Constraints generation time in multiplayer games. The configuration of every instance is encoded as: number of players / branching factor / depth of the tree.

5.2 Results by different solving methods

A number of variables in the formulation of linear program computing SEFCE grows asymptotically at the same rate as a number of strategy profiles. Unfortunately, the set of strategies for each player has a size exponential in a number of information sets. On the other hand, a number of constraints is linear in a number of players and quadratic in a number of information sets. This means that given two game trees with the same nodes, edges and an amount of information sets, the tree in which more players act is expected to have smaller respective linear program computing its SEFCE.

In the primal linear program, the number of variables significantly exceeds the number of constraints. Furthermore, the corresponding matrix form of the program is very sparse. In this case the general guides for choosing a solving algorithm[1]) suggest that the preferred methods are either primal variant of simplex, which is much more dependent on number of constraints, or any interior point method (also referred to as barrier methods by Cplex).

Another option might be to solve the dual of the linear program. In contrast to the primal, the dual has a polynomial number of variables, but exponential number of constraints. By the duality theorem, its objective value is equal to the objective of the primal, assuming they both are feasible and bounded. The assumption is valid, because SEFCE always exists (as proved in Theorem 4.5). Moreover, the slackness is complementary and hence the product of all primal variables and dual slack variables is 0, as is the product of all dual variables and primal slack variables. Therefore, the optimal solution of the dual reduces the size of the original primal problem, which can be consequently solved faster.

Due to the enormous number of constraints, the formulation of the dual is more suitable for techniques like lazy constraint generation. This method starts with the subset of original constraints and iteratively add those which are violated by the optimal solution so far. Therefore, it enables to solve much larger problems, especially those in which most constraints are not active. Moreover, the simplex method actually simultaneously solves the primal and dual. It means that from an optimal simplex tableau it is possible to read off both an optimal solution to the primal and an optimal solution to the dual.

The interpretation of the primal system is clear, as its formulation is directly derived from the definition of SEFCE. Moreover, the dual also provides a game-theoretic meaning. The interpretation follows from the constructive proof of EFCE in [19], where the dual variable are assumed to characterize a player's "deviation plan". The constraints then encodes a rationality concept called "joint coherence" [39]. This concept claims that the choices with uncertain outcomes should be coherent so that they do not provide opportunities for arbitrage (also called "Dutch books") to an external observer who acts like a betting opponent. Equivalently, the strategies of rational players should not be easily exploitable.

In the following sections are compared the experimental results achieved with different solving methods; with the a priori assumptions about the linear program for computing correlated Stackelberg equilibrium in extensive games.

[1]) `http://www.ibm.com/support/knowledgecenter/#!/SSSA5P_12.2.0/ilog.odms.cplex.help/Content/Optimization/Documentation/CPLEX/_pubskel/CPLEX412.html`

5.2.1 Simplex method

The method most frequently used to solve LP problems is the simplex method. The simplex method is known to be very efficient in practice, even if its formulation by Dantzig was proved to have a worst-case computational complexity of exponential time [26]. In the following figures are presented the solving times of simplex method in two-player games, as well as in multi-player games. Note that the results are approximately an order of magnitude lower than the respective generation times.

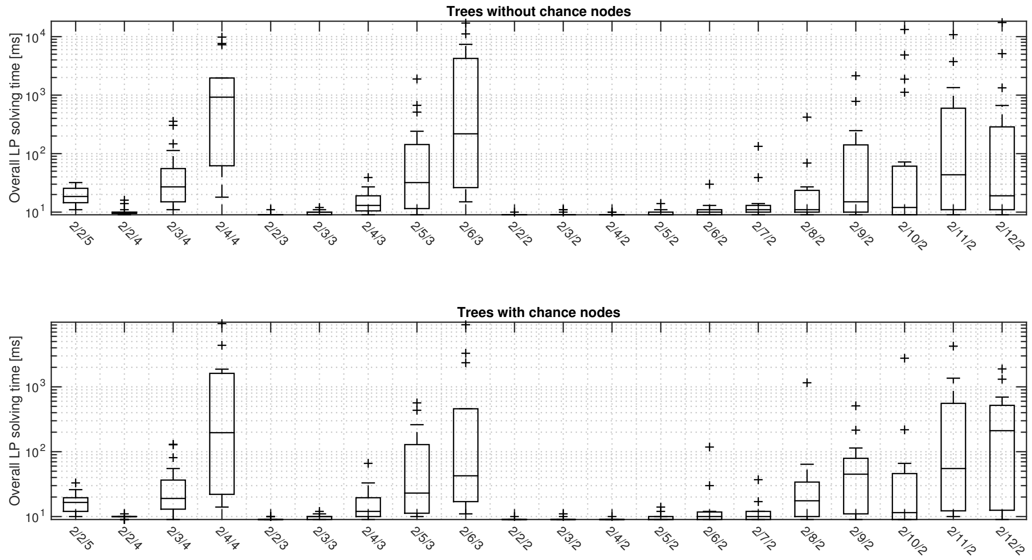

Figure 5.5. Solving time of linear programs in two-player games. The configuration of every instance is encoded as: number of players / branching factor / depth of the tree.

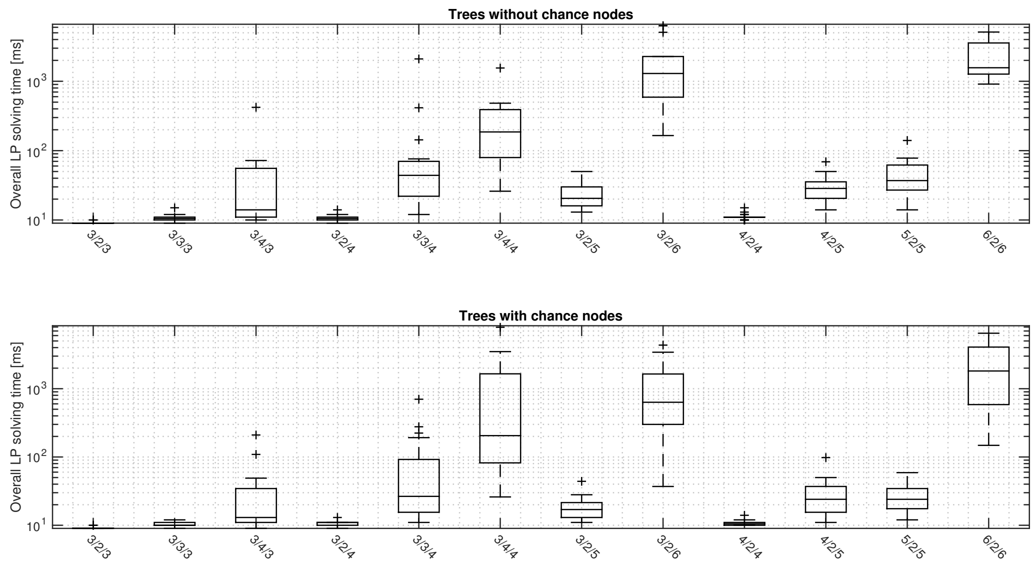

Figure 5.6. Solving time of linear programs in multiplayer games. The configuration of every instance is encoded as: number of players / branching factor / depth of the tree.

However, it still can be seen that the solving times grow exponentially in size of the tree, which is caused by the sizes of the linear programs.

5.2.2 Interior point method

In contrast to the simplex method, interior point method reaches an optimal solution of a linear program by traversing the interior of the feasible polytope. Interior point method is suitable for large-scale, sparse problems, which might be beyond the capabilities of the simplex method. Moreover, the method runs in polynomial time. The following figures depict the solution times of interior point method in two-player and multi-player games.

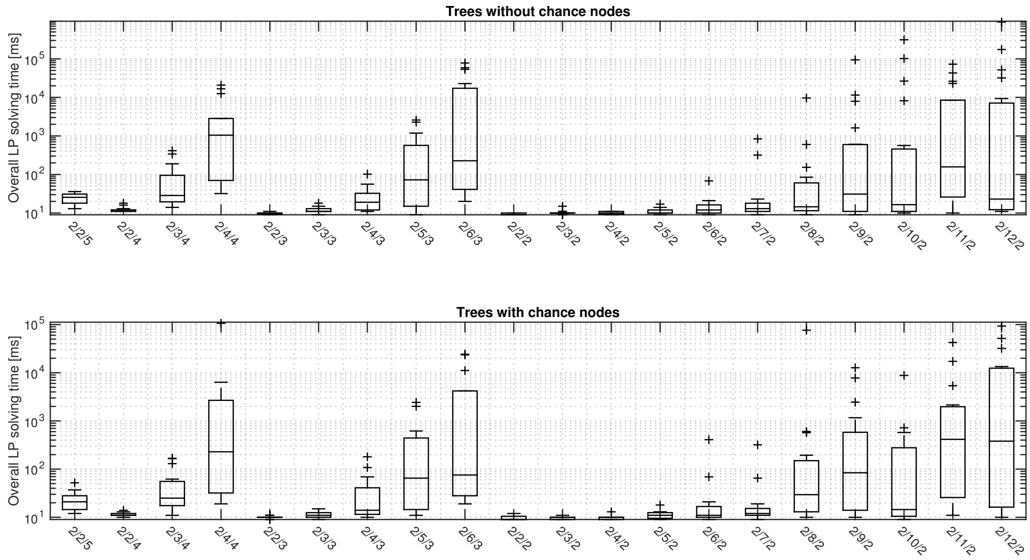

Figure 5.7. Solving time of linear programs in two-player games. The configuration of every instance is encoded as number of players / branching factor / depth of the tree.

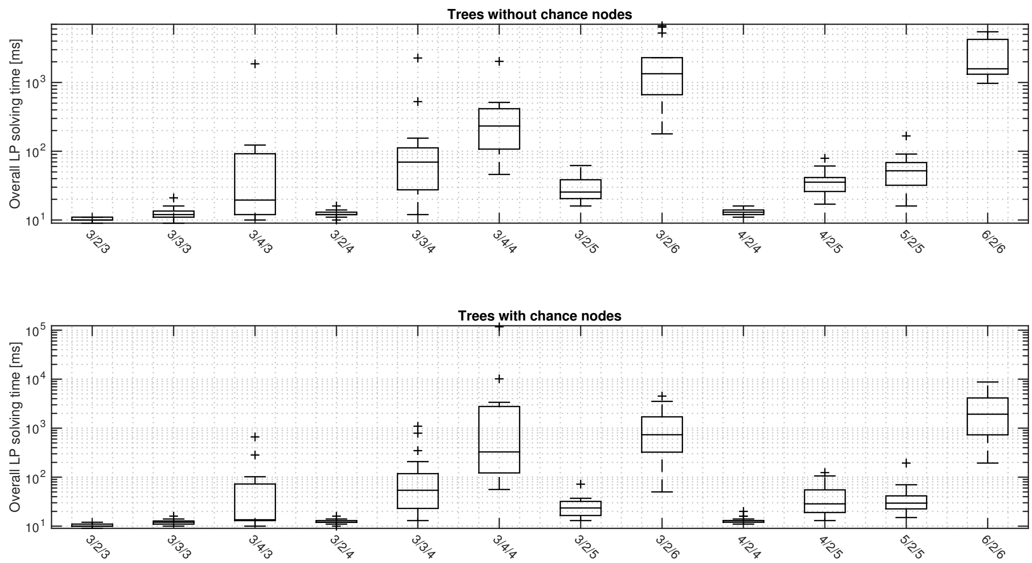

Figure 5.8. Solving time of linear programs in multiplayer games. The configuration of every instance is encoded as number of players / branching factor / depth of the tree.

The results show the same exponential increase in solving time with growing size of the tree.

5.2.3 Lazy constraint generation

Rather than a method, lazy constrain generation is a technique often used to solve linear programs with a large number of constraints. The original formulation is first relaxed so that only a program with a subset of the constraints is solved. The separation procedure adds to the relaxation any constraint which is violated by a current solution. The process is iterated until all constraints are satisfied. The technique works best if it is possible to identify a set of difficult constraints, while the others are most likely to be fulfilled. Lazy constraint generation is used to solve a dual of the program for computing SEFCE, because the dual has an exponential number of constraints. At the beginning, all constrains are relaxed. The obtained results are shown in the following figures.

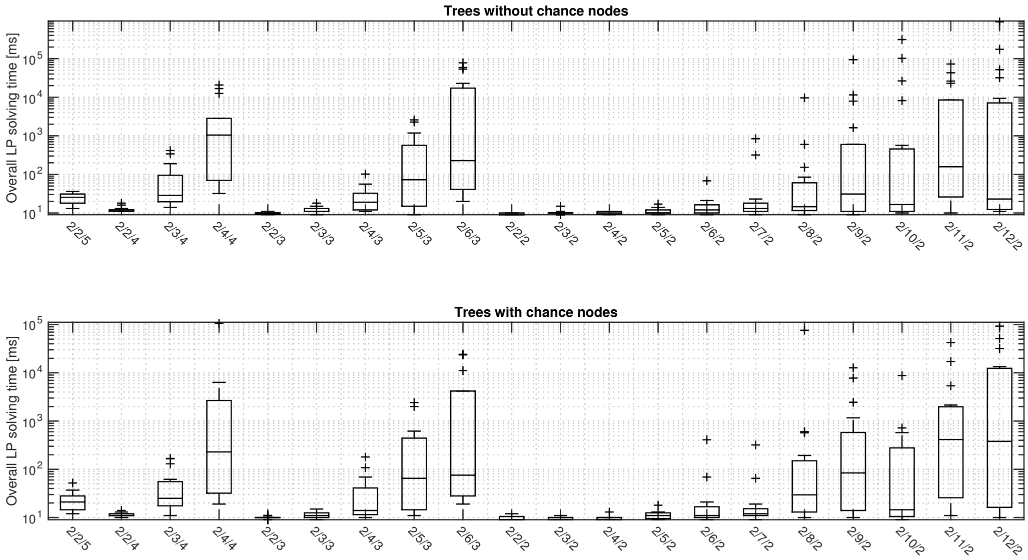

Figure 5.9. Solving time of linear programs in two-player games. The configuration of every instance is encoded as number of players / branching factor / depth of the tree.

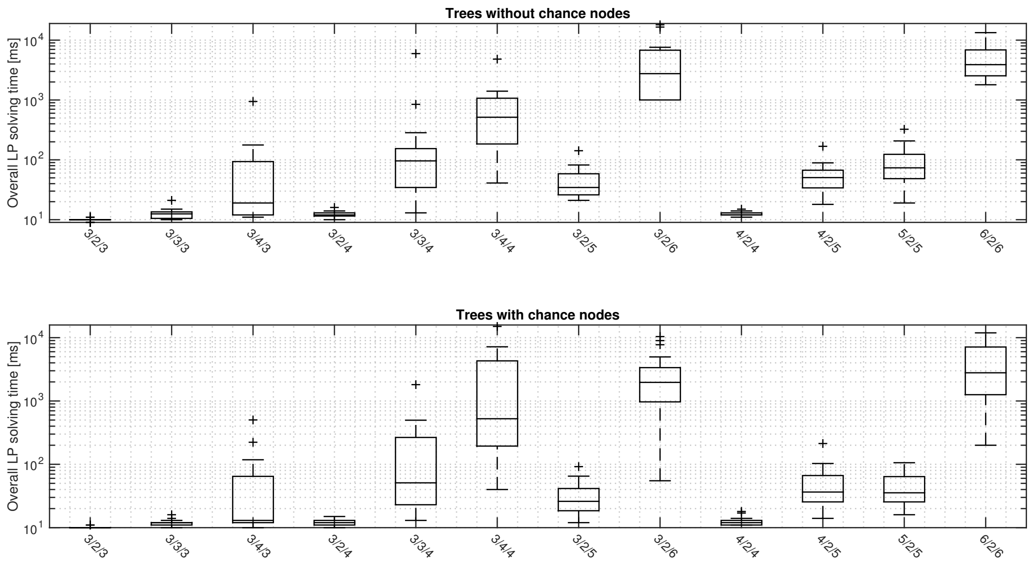

Figure 5.10. Solving time of linear programs in multiplayer games. The configuration of every instance is encoded as number of players / branching factor / depth of the tree.

The presented results confirm that the computation takes a time exponential in a size of the game instance.

5.2.4 Discussion

Finally, the following figure compares the medians of solving times of each instance by different solving methods.

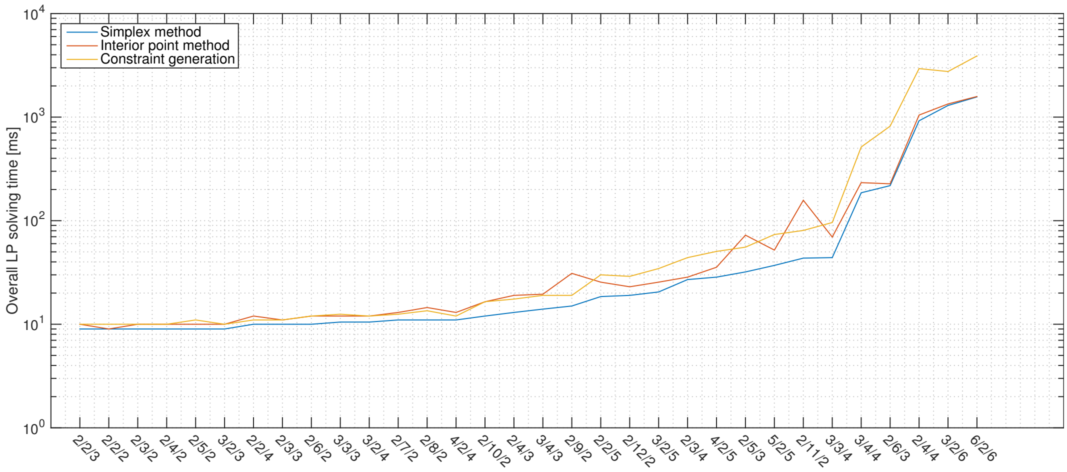

Figure 5.11. Comparison of solving time medians of different solving methods. The configuration of every instance is encoded as: number of players / branching factor / depth of the tree.

It can be seen that the simplex method outperforms both the interior points method and the constraint generation. This finding complies with the assumptions, which expected the simplex method to be suitable for computing SEFCE. However, a poor performance of constraint generation, especially in larger instances of games, is surprising.

Beyond the presented methods, the performance of the solving algorithms can be improved using a domain-dependent knowledge about the structure of the problem. One of the examples might be the delayed column generation (Dantzig decomposition) and delayed row generation. Both these methods require the linear program to be manually decomposed into a master problem and smaller subproblems. Therefore, they are not automatically implemented in solvers like Cplex. The decomposition can not be generally applied to the linear program computing SEFCE, because this system does not comply with the required structure for the decomposition.

Another possibility is to introduce a game-theory related technique. This might include either a restriction to specific subclasses of games which can be represented compactly, so that the resulting linear program for SEFCE does not have an exponential size; or an incremental strategy generation algorithms like the Oracle method [35]. However, even though the original formulation of the Oracle algorithm is based on pure strategies, applying it directly to compute SEFCE is not possible, given that the recommendations are generated sequentially. Nor the reformulation of the Oracle method with the compact sequence-form representation [5] can be used, because it is restricted to two-player games. The implementation would require to extend it to multiple players, which is beyond the scope of this thesis.

Chapter 6
Conclusion

This thesis introduces the solution concept called Stackelberg extensive-form correlated equilibrium in games with multiple followers. In this scenario, one of the players has a commitment power to publicly declare his strategy ahead of the rest of the players. Moreover, he is able to coordinate the course of the game throughout a series of signals sent to his opponents. However, each signal is assumed to be in a sealed envelope and is only revealed to a player when he reaches the point where he can act according to this signal. The applications of this concept can be found frequently in economics, politics or national security. For example, the ongoing NATO's Baltic mission is a situation which can be profoundly modeled as SEFCE.

The thesis first presented related concepts and explained the differences on concrete examples. The algorithms were shown to contain significant similarities with the concept of SEFCE. The analysis of the already existing algorithm for computing correlated Stackelberg equilibrium in games of two players proved the following first main result.

Theorem 4.3. *For a two-player, perfect-recall extensive game with imperfect information and with chance moves, the problem of finding SEFCE is NP-hard.*

The thesis subsequently introduced the linear program for computing correlated Stackelberg equilibrium in games with multiple followers and showed that this concept always exists in every extensive game with perfect recall. The investigation of the computational complexity led to the second main result.

Theorem 4.8. *For an imperfect-information perfect-recall extensive game with more than three players and without chance moves, the problem of finding SEFCE is NP-hard.*

The implementation is able to generate a linear program describing the set of SEFCE for every extensive game with perfect recall. The experiments were performed on randomly generated game tress with simplex method, interior point method and lazy constrain generation as solving algorithms for general linear programs. The results confirmed that the computation takes a time exponential in a size of a game instance. However, solving is still approximately an order of magnitude faster than a generation of a respective linear program.

6.1 Future work

Unless P = NP, the correlated Stackleberg equilibrium can not be expected to a have a compact and efficiently (polynomial-time) generated description. Even so, solving larger sequential games can be achieved by either strengthening the solving algorithm with domain-specific knowledge or decomposing the generation of linear program into multiple smaller sets of constraints.

First, the Oracle algorithms might be used to iteratively find the correlated Stackelberg in games with restricted set of strategies, until the equilibrium in the restricted game is equal to SEFCE in the original game. This would require to find an algorithm for computing best response of the follower, with respect to the fact that the recommendations are revealed sequentially. The formulation of [5] would have to be modified for the case of multiple players. Second, the lazy constraint generation might generate the constraints on the fly and stop when a current solution is guaranteed to be optimal, which would bypassed to necessity to enumerate all the constraints. The method is dependent on the oracle which would decide whether the current solution is truly an equilibrium. Third, the algorithm might be restricted to a class of extensive-form games with compact representation, so that the resulting linear program does not have an exponential size.

Moreover, the most demanding part of the computation – the generation of constraints – can be done in parallel; e.g., for each player separately. Hence, it would take much shorter amount of time to generate the constraints. Finally, the approximate algorithmic approaches into computing SEFCE would provide a possibility to solve much larger games while maintaining a bounded error.

References

[1] Allard, Kenneth. *Somalia Operations: Lessons Learned*. University Press of the Pacific, 2002. ISBN 9780898758856.
https://books.google.cz/books?id=hCIIAAAACAAJ.

[2] Aumann, Robert. Subjectivity and correlation in randomized strategies. *Journal of Mathematical Economics*. 1974, Vol. 1, No. 1, pp. 67-96.
http://EconPapers.repec.org/RePEc:eee:mateco:v:1:y:1974:i:1:p:67-96.

[3] Bošanský, Branislav, and Jiří Čermák. Sequence-form Algorithm for Computing Stackelberg Equilibria in Extensive-form Games. In: *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*. AAAI Press, 2015. pp. 805–811. AAAI'15. ISBN 0-262-51129-0.
http://dl.acm.org/citation.cfm?id=2887007.2887119.

[4] Bošanský, Branislav, Simina Brânzei, Kristoffer Arnsfelt Hansen, Peter Bro Miltersen, and Troels Bjerre Sørensen. Computation of Stackelberg Equilibria of Finite Sequential Games. *CoRR*. 2015, Vol. abs/1507.07677.
http://arxiv.org/abs/1507.07677.

[5] Bošanský, Branislav, Christopher Kiekintveld, Viliam Lisý, and Michal Pechoucek. An Exact Double-Oracle Algorithm for Zero-Sum Extensive-Form Games with Imperfect Information. *Journal of Artificial Intelligence Research (JAIR)*. 2014, Vol. 51, pp. 829–866. Available from DOI 10.1613/jair.4477.
http://dx.doi.org/10.1613/jair.4477.

[6] Chu, Francis, and Joseph Halpern. On the NP-completeness of Finding an Optimal Strategy in Games with Common Payoffs. *Int. J. Game Theory*. Heidelberg, Germany, Germany: Physica-Verlag GmbH, oct, 2001, Vol. 30, No. 1, pp. 99–106. ISSN 0020-7276. Available from DOI 10.1007/s001820100066.
http://dx.doi.org/10.1007/s001820100066.

[7] Conitzer, Vincent, and Dmytro Korzhyk. Commitment to Correlated Strategies. In: *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2011, San Francisco, California, USA, August 7-11, 2011*. 2011.
http://www.aaai.org/ocs/index.php/AAAI/AAAI11/paper/view/3645.

[8] Conitzer, Vincent, and Tuomas Sandholm. Computing the Optimal Strategy to Commit to. In: *Proceedings of the 7th ACM Conference on Electronic Commerce*. New York, NY, USA: ACM, 2006. pp. 82–90. EC '06. ISBN 1-59593-236-4. Available from DOI 10.1145/1134707.1134717.
http://doi.acm.org/10.1145/1134707.1134717.

[9] Conti, Stefano. *Algorithms for finding leader-follower equilibrium with multiple followers*.
http://hdl.handle.net/10589/92066.

[10] Cook, Stephen A. The Complexity of Theorem-proving Procedures. In: *Proceedings of the Third Annual ACM Symposium on Theory of Computing*. New

York, NY, USA: ACM, 1971. pp. 151–158. STOC ’71. Available from DOI 10.1145/800157.805047.
http://doi.acm.org/10.1145/800157.805047.

[11] Čermák, Jiří, Branislav Bošanský, Karel Durkota, Viliam Lisý, and Christopher Kiekintveld. Using Correlated Strategies for Computing Stackelberg Equilibria in Extensive-Form Games. In: *Proceedings of the Thirtyth AAAI Conference on Artificial Intelligence*. 2016.
http://agents.fel.cvut.cz/~durkota/files/AAAI16.pdf.

[12] Daalder, Ivo H, and Michael E. O’Hanlon. *Winning ugly: NATO’s war to save Kosovo*. Brookings Institution Press, 2004.

[13] Forges, Françoise. Five legitimate definitions of correlated equilibrium in games with incomplete information. *Theory and Decision*. Vol. 35, No. 3, pp. 277–310. ISSN 1573-7187. Available from DOI 10.1007/BF01075202.
http://dx.doi.org/10.1007/BF01075202.

[14] Forges, Françoise. An Approach to Communication Equilibria. *Econometrica*. 1986, Vol. 54, No. 6, pp. 1375-85.
http://EconPapers.repec.org/RePEc:ecm:emetrp:v:54:y:1986:i:6:p:1375-85.

[15] Friedman, James W. *Oligopoly and the theory of games*. North-Holland Pub. Co., 1977. Advanced textbooks in economics. ISBN 9780720405057.
https://books.google.cz/books?id=ItG1lghL7-oC.

[16] Gawrych, George W. *The Albatross of Decisive Victory: War and Policy Between Egypt and Israel in the 1967 and 1973 Arab-Israeli Wars*. Greenwood Publishing Group, 2000.

[17] Gilboa, Itzhak, and Eitan Zemel. Nash and correlated equilibria: Some complexity considerations. *Games and Economic Behavior*. 1989, Vol. 1, No. 1, pp. 80-93.
http://EconPapers.repec.org/RePEc:eee:gamebe:v:1:y:1989:i:1:p:80-93.

[18] Hamilton, Jonathan, and Steven M. Slutsky. Endogenous timing in duopoly games: Stackelberg or cournot equilibria. *Games and Economic Behavior*. 1990, Vol. 2, No. 1, pp. 29-46.
http://EconPapers.repec.org/RePEc:eee:gamebe:v:2:y:1990:i:1:p:29-46.

[19] Huang, Wan. *Equilibrium Computation for Extensive Games*. London School of Economics, 2011. Ph.D. Thesis.
http://www.maths.lse.ac.uk/Personal/stengel/phds/WanThesis.pdf.

[20] Huang, Wan, and Bernhard von Stengel. Internet and Network Economics: 4th International Workshop, WINE 2008, Shanghai, China, December 17-20, 2008. Proceedings. In: Christos Papadimitriou, and Shuzhong Zhang, eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008. pp. 506–513. ISBN 978-3-540-92185-1. Available from DOI 10.1007/978-3-540-92185-1_56.
http://dx.doi.org/10.1007/978-3-540-92185-1_56.

[21] Jiang, Albert Xin, and Kevin Leyton-Brown. Polynomial-time Computation of Exact Correlated Equilibrium in Compact Games. *CoRR*. 2010, Vol. abs/1011.0253.
http://arxiv.org/abs/1011.0253.

[22] Kamien, Morton I, Yair Tauman, and Shmuel Zamir. On the value of information in a strategic conflict. *Games and Economic Behavior*. 1990, Vol. 2, No. 2, pp. 129 - 153. ISSN 0899-8256. Available from DOI

http://dx.doi.org/10.1016/0899-8256(90)90026-Q.
http://www.sciencedirect.com/science/article/pii/089982569090026Q.

[23] Karmarkar, Narendra K. A New Polynomial-time Algorithm for Linear Programming. *Combinatorica*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., dec, 1984, Vol. 4, No. 4, pp. 373–395. ISSN 0209-9683. Available from DOI 10.1007/BF02579150.
http://dx.doi.org/10.1007/BF02579150.

[24] Khachiyan, Leonid G. Polynomial algorithms in linear programming. *USSR Computational Mathematics and Mathematical Physics*. 1980, Vol. 20, No. 1, pp. 53 - 72. ISSN 0041-5553. Available from DOI http://dx.doi.org/10.1016/0041-5553(80)90061-0.
http://www.sciencedirect.com/science/article/pii/0041555380900610.

[25] Kiekintveld, Christopher, Manish Jain, Jason Tsai, James Pita, Fernando Ordóñez, and Milind Tambe. Computing Optimal Randomized Resource Allocations for Massive Security Games. In: *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems - Volume 1*. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems, 2009. pp. 689–696. AAMAS '09. ISBN 978-0-9817381-6-1.
http://dl.acm.org/citation.cfm?id=1558013.1558108.

[26] Klee, Victor, and George J. Minty. *How good is the simplex algorithm?*

[27] Koller, Daphne, Nimrod Megiddo, and Bernhard von Stengel. Efficient Computation of Equilibria for Extensive Two-Person Games. *Games and Economic Behavior*. 1996, Vol. 14, No. 2.
http://ai.stanford.edu/~koller/Papers/Koller+al:GEB96.pdf.

[28] Korzhyk, Dmytro, Vincent Conitzer, and Ronald Parr. Complexity of Computing Optimal Stackelberg Strategies in Security Resource Allocation Games. In: Maria Fox, and David Poole, eds. *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2010, Atlanta, Georgia, USA, July 11-15, 2010*. AAAI Press, 2010.
http://www.aaai.org/ocs/index.php/AAAI/AAAI10/paper/view/1870.

[29] Krishna, Vijay. *Auction Theory*. Academic Press, 2002. ISBN 9780124262973.
https://books.google.cz/books?id=F2FON7GVtcMC.

[30] Letchford, Joshua, and Vincent Conitzer. Computing optimal strategies to commit to in extensive-form games. In: *Proceedings 11th ACM Conference on Electronic Commerce (EC-2010), Cambridge, Massachusetts, USA, June 7-11, 2010*. 2010. pp. 83–92. Available from DOI 10.1145/1807342.1807354.
http://doi.acm.org/10.1145/1807342.1807354.

[31] Letchford, Joshua, Liam MacDermed, Vincent Conitzer, Ronald Parr, and Charles L. Isbell. Computing Optimal Strategies to Commit to in Stochastic Games. In: *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence, July 22-26, 2012, Toronto, Ontario, Canada.* 2012.
http://www.aaai.org/ocs/index.php/AAAI/AAAI12/paper/view/5151.

[32] Lou, Jian, and Yevgeniy Vorobeychik. Decentralization and Security in Dynamic Traffic Light Control. In: *Proceedings of the Symposium and Bootcamp on the Science of Security*. New York, NY, USA: ACM, 2016. pp. 90–92. HotSos '16. ISBN 978-1-4503-4277-3. Available from DOI 10.1145/2898375.2898384.
http://doi.acm.org/10.1145/2898375.2898384.

[33] Maschler, Michael. *A price leadership method for solving the inspector's non-constant-sum game*. Available from DOI 10.1002/nav.3800130103.
http://dx.doi.org/10.1002/nav.3800130103.

[34] McAfee, Randolph, and John McMillan. Auctions and Bidding. *Journal of Economic Literature*. 1987, Vol. 25, No. 2, pp. 699-738.
http://EconPapers.repec.org/RePEc:aea:jeclit:v:25:y:1987:i:2:p:699-738.

[35] McMahan, H. Brendan, Geoffrey J. Gordon, and Avrim Blum. Planning in the Presence of Cost Functions Controlled by an Adversary. In: *Machine Learning, Proceedings of the Twentieth International Conference (ICML 2003), August 21-24, 2003, Washington, DC, USA*. 2003. pp. 536–543.
http://www.aaai.org/Library/ICML/2003/icml03-071.php.

[36] Molis, Arūnas. Baltic Military Cooperation: Past, Present and the Future.. *Lithuanian Foreign Policy Review*. 2009, No. 22.

[37] Myerson, Roger. Multistage Games with Communication. *Econometrica*. 1986, Vol. 54, No. 2, pp. 323-58.
http://EconPapers.repec.org/RePEc:ecm:emetrp:v:54:y:1986:i:2:p:323-58.

[38] Nash, John F. Non-Cooperative Games. *Annals of Mathematics*. Annals of Mathematics, 1951, Vol. 54, No. 2.

[39] Nau, Robert F, and Kevin F McCardle. Coherent behavior in noncooperative games. *Journal of Economic Theory*. 1990, Vol. 50, No. 2, pp. 424 - 444. ISSN 0022-0531. Available from DOI http://dx.doi.org/10.1016/0022-0531(90)90011-8.
http://www.sciencedirect.com/science/article/pii/0022053190900118.

[40] Panda, Swetasudha, and Yevgeniy Vorobeychik. Stackelberg Games for Vaccine Design. In: *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems, 2015. pp. 1391–1399. AAMAS '15. ISBN 978-1-4503-3413-6.
http://dl.acm.org/citation.cfm?id=2772879.2773330.

[41] Papadimitriou, Christos H., and Tim Roughgarden. Computing Correlated Equilibria in Multi-player Games. *J. ACM*. New York, NY, USA: ACM, aug, 2008, Vol. 55, No. 3, pp. 14:1–14:29. ISSN 0004-5411. Available from DOI 10.1145/1379759.1379762.
http://doi.acm.org/10.1145/1379759.1379762.

[42] Paruchuri, Praveen, Jonathan P. Pearce, Janusz Marecki, Milind Tambe, Fernando Ordóñez, and Sarit Kraus. Playing games for security: an efficient exact algorithm for solving Bayesian Stackelberg games. In: *7th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2008), Estoril, Portugal, May 12-16, 2008, Volume 2*. 2008. pp. 895–902. Available from DOI 10.1145/1402298.1402348.
http://doi.acm.org/10.1145/1402298.1402348.

[43] Pita, James, Manish Jain, Janusz Marecki, Fernando Ordóñez, Christopher Portway, Milind Tambe, Craig Western, Praveen Paruchuri, and Sarit Kraus. Deployed ARMOR Protection: The Application of a Game Theoretic Model for Security at the Los Angeles International Airport. In: *Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems: Industrial Track*. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems, 2008. pp. 125–132. AAMAS '08.
http://dl.acm.org/citation.cfm?id=1402795.1402819.

[44] Píbil, Radek, Viliam Lisý, Christopher Kiekintveld, Branislav Bošanský, and Michal Pěchouček. Decision and Game Theory for Security: Third International Conference, GameSec 2012, Budapest, Hungary, November 5-6, 2012. Proceedings. In: Jens Grossklags, and Jean Walrand, eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012. pp. 201–220. ISBN 978-3-642-34266-0. Available from DOI 10.1007/978-3-642-34266-0_12.
http://dx.doi.org/10.1007/978-3-642-34266-0_12.

[45] Rubin, Paul H. The Theory of the Firm and the Structure of the Franchise Contract. *The Journal of Law & Economics*. JSTOR, 1978, Vol. 21, No. 1, pp. 223–233.

[46] Schelling, Thomas C. *The Strategy of Conflict*. Harvard University Press, 1980. ISBN 9780674840317.
https://books.google.cz/books?id=7RkL4Z8Yg5AC.

[47] Shieh, Eric, Bo An, Rong Yang, Milind Tambe, Craig Baldwin, Joseph DiRenzo, Ben Maule, and Garrett Meyer. PROTECT: A Deployed Game Theoretic System to Protect the Ports of the United States. In: *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems - Volume 1*. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems, 2012. pp. 13–20. AAMAS '12. ISBN 0-9817381-1-7, 978-0-9817381-1-6.
http://dl.acm.org/citation.cfm?id=2343576.2343578.

[48] Shoham, Yoav, and Kevin Leyton-Brown. *Multiagent systems: Algorithmic, game-theoretic, and logical foundations*. Cambridge University Press, 2009.
http://www.masfoundations.org/mas.pdf.

[49] Skarphedinsson, Össur. *Iceland's Interests and a Responsible Foreign Policy*.
https://www.mfa.is/media/Skyrslur/Executive-summary.pdf.

[50] Spence, Michael. Job Market Signaling. *The Quarterly Journal of Economics*. 1973, Vol. 87, No. 3, pp. 355-374. Available from DOI 10.2307/1882010.
http://qje.oxfordjournals.org/content/87/3/355.abstract.

[51] Stackelberg, Heinrich. *Market Structure and Equilibrium*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011. ISBN 978-3-642-12586-7. Available from DOI 10.1007/978-3-642-12586-7_1.
http://dx.doi.org/10.1007/978-3-642-12586-7_1.

[52] Stengel, Bernhard von, and Françoise Forges. Extensive-Form Correlated Equilibrium: Definition and Computational Complexity. *Math. Oper. Res.* 2008, Vol. 33, No. 4, pp. 1002–1022. Available from DOI 10.1287/moor.1080.0340.
http://dx.doi.org/10.1287/moor.1080.0340.

[53] Stengel, Bernhard von, and Shmuel Zamir. *Leadership with commitment to mixed strategies*.
http://www.cdam.lse.ac.uk/Reports/Files/cdam-2004-01.pdf.

[54] Tambe, Milind. *Security and Game Theory: Algorithms, Deployed Systems, Lessons Learned*. 1st ed. New York, NY, USA: Cambridge University Press, 2011. ISBN 1107096421, 9781107096424.

[55] Toaldo, Mattia. Decentralising authoritarianism? The international intervention, the new 'revolutionaries' and the involution of Post-Qadhafi Libya. *Small Wars and Insurgencies*. Jan 2, 2016, Vol. 27, No. 1, pp. 39-58. ISSN 0959-2318. Available from DOI 10.1080/09592318.2016.1122905.
http://dx.doi.org/10.1080/09592318.2016.1122905.

[56] Torre, Pablo E.F. de la. *Airline Alliances: The Airline Perspective*. Massachusetts Institute of Technology, Department of Aeronautics and Astronautics, Flight Transportation Laboratory, 1999. FTL report.
https://books.google.cz/books?id=lekLNQAACAAJ.

[57] Vaněk, Ondřej, Zhengyu Yin, Manish Jain, Branislav Bošanský, Milind Tambe, and Michal Pěchouček. Game-theoretic Resource Allocation for Malicious Packet Detection in Computer Networks. In: *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems - Volume 2*. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems, 2012. pp. 905–912. AAMAS '12. ISBN 0-9817381-2-5, 978-0-9817381-2-3.
http://dl.acm.org/citation.cfm?id=2343776.2343826.

[58] Wan, Zhiyu, Yevgeniy Vorobeychik, Weiyi Xia, Ellen Wright Clayton, Murat Kantarcioglu, Ranjit Ganta, Raymond Heatherly, and Bradley Malin. A Game Theoretic Framework for Analyzing Re-Identification Risk. *PLoS ONE*. Public Library of Science, 03, 2015, Vol. 10, No. 3, pp. 1-24. Available from DOI 10.1371/journal.pone.0120592.
http://dx.doi.org/10.1371%2Fjournal.pone.0120592.

[59] Zamir, Shmuel, Morton I. Kamien, and Yair Tauman. *Information Transmission* . Available from DOI http://dx.doi.org/10.1016/B978-0-12-370182-4.50019-2.
http://www.sciencedirect.com/science/article/pii/B9780123701824500192.

Appendix A
Specification

Czech Technical University in Prague
Faculty of Electrical Engineering

Department of Computer Science and Engineering

DIPLOMA THESIS ASSIGNMENT

Student: **Bc. Jakub Černý**

Study programme: Open Informatics
Specialisation: Artificial Intelligence

Title of Diploma Thesis: **Stackelberg Extensive-Form Correlated Equilibrium with Multiple Followers**

Guidelines:

Stackelberg Equilibrium (SE) is a solution concept where the leader commits to a strategy that is observed by the follower that plays a best response. Stackelberg Extensive-Form Correlated Equilibrium (SEFCE) extends this solution concept in extensive-form games so that the leader can also send to the follower recommendations what action to play. Recently, a new algorithm has been introduced for computing SE using SEFCE in two player games. However, a more practical case with multiple followers that allow coordination among multiple agents remains unexplored. The goal of the student is to (1) formalize the solution concept based on SEFCE with multiple followers, (2) analyze the computational complexity for computing this equilibrium, (3) design and implement an algorithm for computing this equilibrium, and (4) experimentally analyze the scalability of the new algorithm.

Bibliography/Sources:

[1] B. von Stengel and F. Forges; Extensive-form Correlated Equilibrium: Definition and Computational Complexity, Mathematics of Operations Research 33(4): 1002-1022.
[2] B. Bosansky, J. Cermak; Sequence-Form Algorithm for Computing Stackelberg Equilibria in Extensive-Form Games, AAAI 2015
[3] B. Bosansky, S. Branzei, K. A. Hansen, P. B. Miltersen, T. B. Sørensen; Computation of Stackelberg Equilibria of Finite Sequential Games, WINE 2015
[4] J. Cermak, B. Bosansky, K. Durkota, V. Lisy, C. Kiekintveld; Using Correlated Strategies for Computing Stackelberg Equilibria in Extensive-Form Games, AAAI 2016 (to appear)

Diploma Thesis Supervisor: Mgr. Branislav Bošanský, Ph.D.

Valid until the end of the winter semester of academic year 2017/2018

L.S.

prof. Dr. Michal Pěchouček, MSc.
Head of Department

prof. Ing. Pavel Ripka, CSc.
Dean

Prague, April 8, 2016

Appendix B
Real-World Applications of SEFCE

The range of potential applications of SEFCE is wide. The coordination is required on many levels of human society and its effectiveness is crucial. Even if the parent organization is fully in change, constructing concrete plans and schedules for individual members is completely unthinkable in every large structure. The correlated Stackelberg equilibrium naturally implements the abstraction of complex control on several levels, hence modeling the situation more realistically.

This chapter presents several examples in two straightforward fields of application – economy and military. It discusses the interaction in these systems and shows concrete examples where the coordination might help.

B.1 Economy

Economic cooperation is a concept that is commonly used as a simile for industrial, financial or productive cooperation. The coordination is required in every large organization, which structure often consists of various departments that contribute to the corporation's desired goals. The individual subgoals are distributed across departments like Marketing, Finance, Accounting, Human Resource, or IT. Moreover, there are often smaller divisions within autonomous companies. The diversity in the structure of corporations is great, as the enterprises may range from single firm to multi-corporate conglomerate. Every individual part of the corporation has to comply with the overall strategy, even though it also follows its own goals.

Consider the following several examples of economic cooperation.

- *Airline alliances.* [56] The airline alliances form an aviation industry arrangement between several airlines. The individual companies agree to cooperate on a more or less substantial level. For example, some alliances provide marketing branding to help travelers make inter-airline connections easier. The branding may include unified aircraft liveries of member aircraft. The alliance might agree on a common strategy, while the individual members still strive to maximize their own profit.
- *Franchises.* [45] Franchising is a practice when the franchiser grants a permission to the franchisee to use his firm's business model and brand for a given period of time. The franchiser often obtains three important payments – a royalty for the trademark; a compensation for the training and advisory services; and a percentage of the franchisee's sales. Both the franchiser and the franchisee have different interests to protect. The franchiser has to secure the protection of his trademark, while still controlling the overall business model and securing know-how. The franchisee must ensure that the services of the trademark meet the standards. In contrast to retailing, the franchisee is hence not fully able to make business decisions. However, he is still seen as an independent merchant. Franchising is common for restaurants, convenience stores, hotels, hair salons, etc.
- *Auction systems.* [29] By definition, an auction is a market institution with an explicit set of rules determining resource allocation and prices on the basis of bids

from the market participants [34]. Participants in auctions bid against one another in order to acquire a desired good. The common assumption is that each subsequent bid is required to be higher than the previous bid. An auctioneer announces the rules under which he commits to sell the goods, while the individual participants will not bid if the price exceeds the price they are willing to accept. For the auctioneer it might hence be profitable to coordinate the participants so that all goods are sold while none of the bidders exceeds their budget.

B.2 Military

During the last century, organizations like NATO developed a global network of partners facing security threats like terrorism, piracy, cyber warfare or even large-scale conflicts. Another examples of the international coordination include the United Nations, the European Union, the Organization for Security and Cooperation in Europe (OSCE) or the African Union. These alliances participate in missions in Afghanistan (14 countries) or Kosovo (10 countries). The scale of the missions is wide and demands long-term planning in order to coordinate the forces of several partner countries. The partnerships and cooperation further proliferate through institutions and multilateral forums including the Euro-Atlantic Partnerships Council, the Mediterranean Dialogue and the Istanbul Cooperation Initiative.

The coordination is managed in alliance's commands structures which adopts a global (Stackelberg) strategy. Afterwards, the military units of participating countries take positions consistent with the leading strategy. The cooperation is sequential, as the commands are not delivered all at once. One of the reasons might be the security, the second is to guarantee a faster response to altering conditions. The modeling as SEFCE is hence eligible.

Now consider several short descriptions of past or ongoing international military operations, where the coordination was more or less succesfully used and which might have also ended differently in case the cooperation was more effective. The examples emphasize the scope of coordination required to manage all the forces.

- *Baltic Air Policing.* [36] Within the Alliance, the integrity of airspace of Baltic states is secured as a collective task of individual members. Estonia has on its territory several NATO bases, which assist the Estonian armed forces with the protection of maritime and land borders. Moreover, the Allies completely secure the protection of Estonian airspace. Also the other two Baltic states – Latvia and Lithuania – do not have their own army aviation units and their airspace is covered by Baltic Air Policing mission in the context of Quick Reaction Alert (QRA). The policing of the airspace of the Baltic States is managed on a three-month rotation by the members of Alliance, including the Czech JAS-39 Gripen. NATO also coordinates other region's states, even if Finland and Sweden are not members of the Alliance and hence not a part of NATO's collective defence clause. The partnership includes the exchange of information or coordinated training and exercises in order to develop better joint situational awareness. The Baltic policing mission is under command of NATO HQ Aircom.
- *Iceland Air Policing.* [49] Another mission in the Alliance's QRA is Iceland Air Policing. The range of tasks is similar to Baltic Air Policing, however, the presence of NATO aircraft in Iceland is not permanent, as in the case of the Baltic countries. The mission is conducted jointly and collectively with Icelandic Coast Guard, which

is equipped with ships and helicopters. The former US air base serves mainly the Coast Guard and occasionally the fighter aircraft of the Allies. The deployment is again conducted on a two-week or three-week rotation by NATO members. The mission falls under the NATO HQ Aircom.

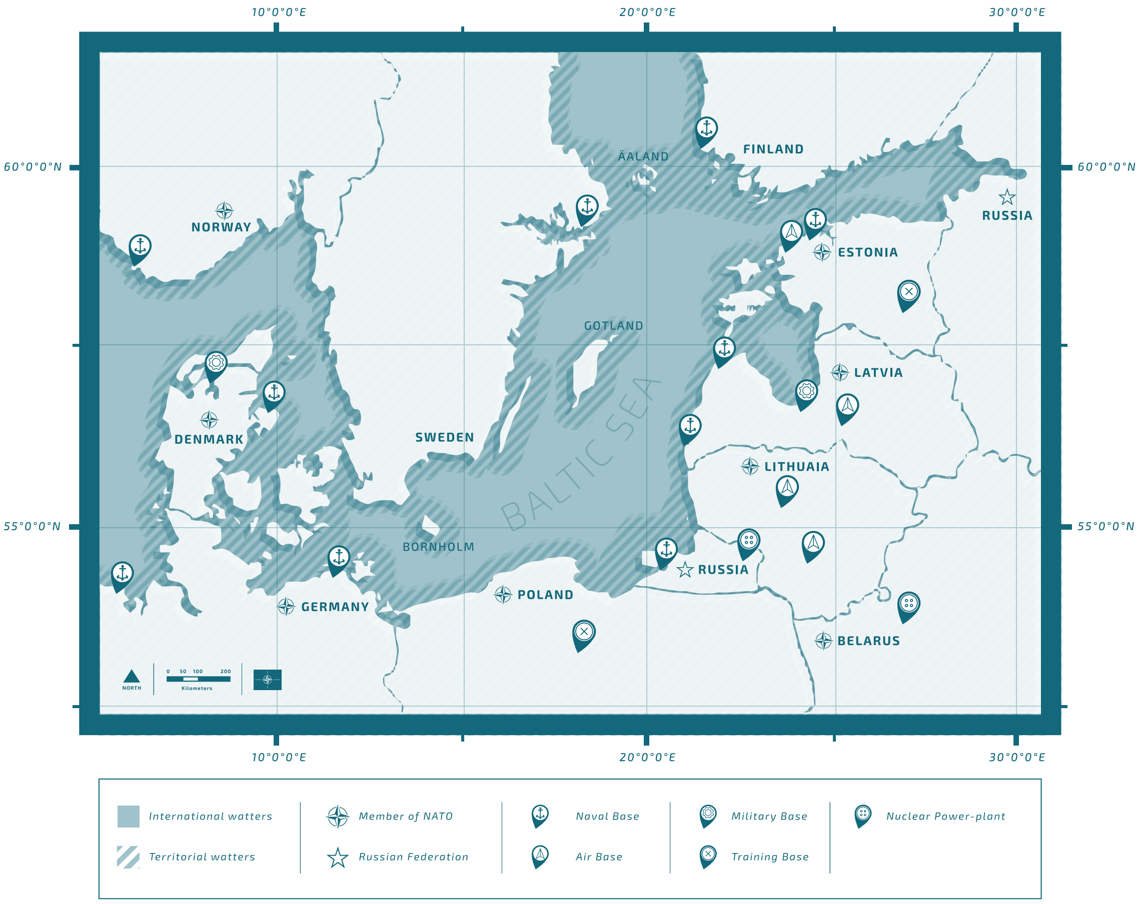

Figure B.1. A map of military bases involved in Baltic Air Policing mission.

- *Somalia Civil War.* [1] The operation of the United Nations in Somalia started in 1992 as a peacekeeping mission UNOSOM I/II and its goal was to stabilize the country which was already in a civil war. The joint forces of UNITAF (United Task Force) were under US command and contained military units from more than 17 countries (for example Great Britain, France, but also Ethiopia, Saudi Arabia or Pakistan). At the beginning the humanitarian aid was successfully distributed across the country, but due to the increasing number of casualties and the continued unwillingness to participate in the inconclusive conflict the forces withdrew at spring 1995. Nowadays, Somalia is torn into many autonomous parts and the central government does not fully control even the capital Mogadishu.
- *Yom Kippur War.* [16] In October 1973, the coalition of Arab states led by Egypt and Syria declared war on Israel. Egypt started attacking Sinai, Syria stared attacking Golan Heights both commanding their military operations alone but coordinating with each other. They were also commanding expeditionary forces from six other Arab countries. The war began when the Arab coalition launched a joint surprise attack on Israeli positions. The invasion was conducted during Yom Kippur holiday, the most important holiday of the Jewish year. The Goal of the coalition was retaking lands lost during Six Day War. Unlike previously Israel failed to launch a preemptive attack because Prime Minister of Israel Golda Meir concluded that war was not a

certainty. The war began with a massive and successful Egyptian crossing of the Suez Canal. After crossing the cease-fire lines, Egyptian forces advanced virtually unopposed into the Sinai Peninsula. After three days, Israel had mobilized most of its forces and halted the Egyptian offensive, resulting in a military stalemate. Israel had encountered military difficulties on both fronts. It became clear by October 9 that no quick reversal in Israel's favor would occur and that IDF losses were unexpectedly high.

The war had far-reaching implications. The Arab World, felt vindicated by early successes in the conflict. In Israel, despite impressive operational and tactical achievements on the battlefield, the war led to recognition that there was no guarantee that Israel would always dominate the Arab states militarily.

- *Libya Civil War.* [55] War in Libya had several phases. In January 2011, upset at delays in the building of housing units and over political corruption, protesters in major Lybian cities broke into, and occupied, housing that the government had been building. Protesters also clashed with police and attacked government offices. A civil war broke out during the Arab Spring. After a number of atrocities were committed by the government, multinational coalition led by NATO forces intervened in late March 2011 with the aim to protect civilians against attacks by the government's forces. NATO ambassadors agreed that NATO would take command of the no-fly zone enforcement. The initial coalition of France Canada, Denmark, UK, Italy, Norway, Qatar, Spain and US grew to nineteen states, with newer states mostly enforcing the no-fly zone and naval blockade.

  In August, rebel forces launched an offensive on the government-held coast of Libya, taking back territory lost months before and ultimately capturing the capital city of Tripoli. In the aftermath of the civil war, insurgency by former Gaddafi loyalists continued. There have been various disagreements and strife between local militia and tribes, including fighting on 23 January 2012 in the former Gaddafi stronghold of Bani Walid. Some have refused to disarm and cooperation with the transitional Lybian government has been strained, leading to demonstrations against militias and government action to disband such groups or integrate them into the Libyan military. Continuous struggle to stabilize Lybia led directly to a second civil war in Libya.

- *Kosovo War.* [12] The Kosovo War was an armed conflict in Kosovo in 1998. It was fought by the forces of the Yugoslavia, which controlled Kosovo before the war, and the Kosovo Albanian rebel group known as the Kosovo Liberation Army (KLA), with air support from the NATO. KLA sought the separation of Kosovo from the Yugoslavia. Yugoslav security forces killed many Albanian civilians during the war; there were also attacks on Yugoslav security forces and moderate Serb-friendly Albanians by the KLA. According to Human Rights Watch, the vast majority of the violations from January 1998 to April 1999 were attributable to Serbian Police or the Yugoslav Army.

  In October 1998, the North Atlantic Council issued activation orders for the execution of air campaign in Yugoslavia. The goal was strategic bombardment of Yugoslavian targets. NATO over the course of the war deployed and coordinated over one thousand aircrafts. Making it the largest military air operation since the World War 2 in europe. The war ended with the Kumanovo Treaty, with Yugoslav forces agreeing to withdraw from Kosovo to make way for an international presence. The NATO bombing campaign has remained controversial, as it did not gain the approval of the UN Security Council.

Appendix C
Abbreviations, Functions and Symbols

C.1 Abbreviations

AAAI	Association for the Advancement of Artificial Intelligence.
AI	Artificial intelligence.
CE	Correlated Equilibrium.
EFCE	Extensive-form Correlated Equilibrium.
LP	Linear program.
NATO	North Atlantic Treaty Organization.
NE	Nash equilibrium.
SAT	Boolean satisfiability problem.
SCE	Stackelberg correlated Equilibrium.
SE	Stackelberg Equilibrium.
SEFCE	Stackelberg Extensive-form Correlated Equilibrium.
SSE	Strong Stackelberg Equilibrium.
UN	United Nations.

C.2 Functions and symbols

$\sqsubseteq$	A prefix relation on sequences.
$-i$	A set of opponents of player i.
β	A behavioral strategy.
δ	A mixed strategy.
Δ	A set of mixed strategies.
π	A pure strategy.
Π	A set of pure strategies.
Π^*	A set of reduced pure strategies.
σ	A sequence in a sequence-form game representation.
Σ	A set of sequences in a sequence-form game representation.
$\rho(h)$	A player function for node h.
A	A set of sets of actions for each player.
$A(h)$	A set of actions in node h.
$agr(\sigma)$	A set of agreeing strategies for sequence σ.
$agr(h)$	A set of (possibly partial) agreeing strategy profiles for node h.
B	A set of behavioral strategies.
$C(a)$	A probability function for performing a chance action a.
$C(h)$	A probability of reaching node h due to chance.
$Ext(\sigma_i)$	A set of extensions of a sequence σ_i.
$g_i(\sigma_1, ..., \sigma_n)$	An extended utility function for player i.
H	A set of nodes in a game tree.
$inf_i(\sigma_i)$	An information set in which the last action of σ_i is taken.
N	A set of players.
$p(\sigma_1, ..., \sigma_n)$	A correlation plan of sequences $\sigma_1, ..., \sigma_n$.
$r_i(\sigma_i)$	A realization plan of sequence σ_i for player i.
$rel(\sigma_i)$	A set of sequences of $-i$ which form a relevant pair with σ_i.
$seq_i(h)$	A set of sequences leading to node h for player i.
$seq_i(I)$	A set of sequences leading to information set I for player i.
$u_i(a_1, ..., a_n)$	A utility function for player i.
Z	A set of terminal nodes in a game tree.
$\mathbb{R}$	A set of real numbers.

Appendix D
CD Content

At the CD are located several files which require a third party software to be executed. Specifically, this includes the source code of algorithm for computing SEFCE, the text of this thesis and numerous figures and schemata. All necessary programs can be downloaded for free (or at least in an academic license) from the websites in footnotes:

- **Java** In version at least 1.7[1])
- **IBM ILOG Cplex** In version at least 2.4[2])
- **TEX** Both LaTeX and plainTeX[3])

This document was typeset in PlainTeX using CSplain[4]) in order to print a few Czech characters and the CTUstyle[5]) template by Petr Olšák, to whom I'm really grateful for that. The images of game trees and payoff matrices were programmed manually using Tikz.

The enclosed CD contains following files and directories:

- **cernyj49.pdf** – the text of this thesis
- **doc** – directory with the TeX source files of this document
  - **figs** – contains all figures
  - **specification** – contains the specification of this thesis
  - **text** – contains text source files
  - **trees** – contains game tree structures in Tikz
- **source** – directory with the implementation in Java

[1]) http://www.oracle.com/technetwork/java/javase/downloads/index.html
[2]) https://developer.ibm.com/academic/
[3]) https://www.tug.org/texlive/
[4]) http://petr.olsak.net/csplain.html
[5]) http://petr.olsak.net/ctustyle.html