Università della Calabria

Dipartimento di Matematica e Informatica

Dottorato di Ricerca in Matematica e Informatica

XXIX ciclo

Tesi di Dottorato

Pure Strategies in Security Games

Settore Disciplinare INF/01 – INFORMATICA

Coordinatore: Ch.mo Prof. Nicola Leone

Supervisore: Ch.mo Prof. Gianluigi Greco

Dottorando: Dott. Marek Adrian

Abstract

In the past decades game theory allowed to model mathematically conflicts between groups of players, each with their own agenda, and found ways to apply them in real life situations. One of such applications, which has been studied heavily in recent years, are security games. Their primary purpose is to help find a way to distribute scarce resources of an entity whose job is to minimize the potential chances of a successful attack on a system which is under its care. The underlying assumption is that the second player, whose task is to launch an successful attack on the system, has sufficient time to observe the way in which the resources are used to find any patterns, or regularities. These games have been successfully applied to several real life situations like the way checkpoints and canine units are deployed on the LAX airport, the way air field marshals are scheduled to fly on commercial airplanes, or the way patrol routs are made in the Boston port. All of these works concentrate on considering mixed strategies, that is using a random element to decide how the defender manages his resources. While this approach is understandable in the case of one player managing all of the resources, it becomes less obvious in the situation where multiple defenders, with knowledge of other players priorities and limited means to coordinate are given. The problem of finding what can be said about pure strategies in such a situation is the goal of this thesis.

This thesis provides a brief overview of what is currently known about modeling security games. The contribution is the following: a model for a multiple defender security game with sequential resource allocation is presented and the notions of reasonable behavior are described; a polynomial algorithm for finding a reasonable move and predicting other players decisions is presented. Moreover, it is proven that even if the actual play of the other players differs from what was predicted, the result will still satisfy the assumption of rationality of the players. Finally, to emulate coordination among the players, the model is expanded by adding an additional player, called the Overseer, whose goal is to ensure that a set of targets is protected by deciding the order in which all the other players commit. It is shown that deciding whether such a sequence of players exist for a set of targets of any size can be reduced to a set of one element in polynomial time. A partial result for which this problem is in P class is shown.

Acknowledgements

I would like to thank:

- Prof. **Gianluigi Greco**, for introducing me to the area of security games and pure strategies, and for the patience and longanimity he has shown.
- Prof. **Nicola Leone**, for providing a great and inspiring working environment, and for all the guidance and support I received from him.
- All my friends and colleagues from the Department of Mathematics and Computer Science at the University of Calabria and Theoretical Computer Science at Jagiellonian University, for forming my interests and scientific development, and for making the years of pursuing a PhD degree special; in particular I thank Dr. **Andrzej Grzesik** for the countless discussions and hosting me at the University of Warsaw during the stage visit.
- My family, for their support and superpowers: my parents **Anna** and **Henryk**, for setting a good example and being always there for me with their advice and help; my brother **Maciek** for being the right man in the right place whenever I need a rescue; my wife **Weronika**, who has good ideas I eagerly follow and helpful/valuable hints I not always wanted to accept; **Jaś** and **Maja** for showing me new perspectives and being the greatest motivation for growth.

Contents

Chapter 1

Introduction

1.1 Background and motivation

The research in this thesis is focused on finding a good computational model to find pure strategies in security games. To describe what a pure strategy is we need to explain what a game, and in particular, what a security game is. To that end, it is most appropriate to use game theory [22], which is concentrated on building mathematical models of conflict resolution. In this theory a *game* is a situation, in which we have at least two separate agents, called players, each of which has

- a set of possible actions, out of which he will pick one, and
- a valuation function which describes the gain he gets depending on the choices of all players involved in the game.

A *move* is a description of how a player picks his action. A game is called a *security game* if it is used to model potential threats to a system and possible ways to counter them. In such games one of the players is distinguished as the *attacking player*. The other are called *defending players*. The set of possible actions is common for all players and is used to describe specific objects in the system, which may be potential points of attack. If, as the end result, the attacking player has picked the same object as one of the defending players, then it will be significantly worse for him, than in the case in which none of the defending players picked that object. The opposite is also true. If, in the end, one of the defending players has picked the same object the attacking player has, the result is much better for all defending players than otherwise.

While several papers [1, 7, 10, 11, 17, 18, 19, 27, 39, 40] considering mixed strategies (that is when the move is a probability distribution over the set of possible actions) have been written, the case of pure strategies (in which the move is picking one of the actions) is left unexplored. The case of pure strategies has several differences compared to the same game with mixed strategies available. For example what is called a *reasonable* strategy is guarantied to exist in the mixed case [26], but not in the pure case. As such questions often gives new insights about the whole problem, we decided to pursue this case.

1.2 Contributions

Firstly, it can be argued that considering the attacking player is not necessary. As he has full information about what the defending players will do and there is no uncertainty involved, he will simply choose the highest valued object out of all that have not been picked by the defending players. With that in mind, the model can be simplified by considering only the defending players, assigning a value to each object depending on his original valuating function, and defining the new valuating function for each player as the sum of values of all objects picked by all players. To simplify the reasoning, it is assumed that the players each have one move, have all values of objects distinct, and pick the objects consecutively in a given order. Now the question to consider is what a reasonable move in such a setting looks like, and how it can be argued that there is a strategy to this decision.

Out of known descriptions of games the perfect information model allowed to describe the problem in the easiest way to analyze. In this case, the game is described as a tree in which each node (except the leaves) is a decision point for one of the players, each edge corresponds to a move made by a player, and each leaf corresponds to the payouts each player is going to get. In such model, a strategy is a function which assigns to each node of the tree the move that the player makes in that situation. It is important to see that if a player uses a strategy, he in a way assumes that he knows what the players after him will do. The strategy can be played by starting at the root of the tree and choosing the next node to be moved into. The intuition behind a *good* strategy is that none of the players alone can change their decisions in such a way that the value of the outcome increases for him.

Finding such a strategy directly, while possible, would be inefficient, as the size of the strategy is exponential with respect to the number of players, so we proposed a slightly modified version of how to decide if a move is good. The

player predicts what moves the following players will make. Based on that prediction he chooses his own move. The move will be called *reasonable*, if a good strategy can be found, which after playing out would give the same outcome.

We have constructed an algorithm that finds a reasonable move in that sense for each player in polynomial time. Furthermore, the algorithm gives correct results in cases where some, maybe all, of the players can make more than one move. We have shown that if the players strengthen their assumptions about other players, in the sense that they will assume that every choice indifferent to the other player will be made at random, we can argue that the outcome of such play is an outcome of a good strategy.

To this point the model assumes no means of communication or coordination between the players. In an attempt to simulate coordination efforts from an outside party we defined a broader problem in which a new player, called the Overseer, is added. Instead of a regular move he can choose the order in which the players make the moves trying to make sure that some specific objects will be picked. We have showed that the size of the set of specific objects does not affect the complexity of the decision. Then a specific case of the Overseer problem is considered can solved, where the object is always exactly the n-th desired object for all players, where n is the number of players.

A paper summarizing these finding is currently being prepared for a conference.

1.3 Outline

The thesis is organized as follows:

Chapter 2 contains all the basic definitions used in the following chapters with some examples of them being used.

Chapter 3 focuses on describing what we know so far about security games, pure Nash equilibria, and contains an analysis how this knowledge is related to the topic of the thesis.

In Chapter 4 the problem is formulated. Furthermore it contains the algorithm used to make a prediction based on which decision a player will commit to. Then it is proved that this prediction will result in a move that is a part of a good strategy. Next it is proved that following the results of the algorithm will still construct a good strategy, even if the predictions of the player happen to be false. Finally, the necessary definitions are provided to introduce an Overseer. A reduction of the whole problem to a simpler version in polynomial time is given as is a partial

solution to the Overseer problem.

Chapter 5 contains a brief summary of the results of this thesis. In the end, a brief outline how future work can be pursued is given.

Chapter 2

Preliminaries

In this chapter we introduce the basic definitions we will use trough the thesis.

2.1 Normal Form Games, strategies and Nash equilibrium

The definitions presented in this section and the next and are basic notions in game theory and taken from the handbook "Essentials of game theory: A concise multidisciplinary introduction" [22]. The most basic form of game is called a Normal Form Game. It is a situation with a set of players, each with his own set of actions and a valuation function. Each player simultaneously assigns probabilities to actions to determine how likely he is to commit to that action. This is called a strategy of the player. Next for each player an action is picked at random according to his strategy. Each player gets a payout given to him by their valuating function depending on all of the actions chosen. The convention is to say that players commit to actions, but for convenience, which will be apparent after stating the main problem, all actions will be called objects.

Definition 2.1.1 (Normal Form Game)
A normal form game *is a tuple* (N, O, u) *where:*

- N *is a finite set of players, indexed by* i*;*
- $O = O_1 \times \cdots \times O_n$ *where* O_i *is a finite set of objects available to player* i*. Each vector* $(o_1, \ldots, o_n) \in O$ *is called an object profile;*

- $u = (u_1, \ldots, u_n)$, *where* $u_i : O \to \Re$ *is a valuating function for player* i.

Having defined a game we need also to create means to reason about how the players plan their moves. For that end we introduce the concept of a strategy. This allows us to explore how a player, knowing which objects are available to him, will reason about how to commit to one of them.

Definition 2.1.2 (Strategy)
Any assignment of probabilities by a player to his set of available objects is called a strategy. *If the assignment gives one of the objects probability* 1 *then it is called a* pure *strategy. Any other strategy is called* mixed. *A* strategy profile *is a vector containing strategies of all players in a given game.*

Knowing how players can make their decisions we still need a method to distinguish a good strategy from a bad one. This cannot be done without adding some assumptions about the players motivations. This idea has been approached from several different angles, depending what type of behavior is being analyzed. Sometimes it is assumed that the players will pick an object which gains them less than maximum personal possible value to maximize the overall gained value. In other cases players are assumed to be selfish, looking only to maximize their own personal gain. We will focus on the latter approach.

For that end the idea of a best response and the Nash equilibrium will be introduced as a proposition for verifying the validity of a solution. To simplify, the following notation will be used. For the vector $o = (o_1, \ldots, o_n)$ we denote $o_{-i} = (o_1, \ldots, o_{i-1}, o_{i+1}, \ldots, o_n)$ and (o_{-i}, o_i) for $(o_1, \ldots, o_n)$.

Definition 2.1.3 (Best response)
For the player i *a* best response *to an object vector* o_{-i} *is an object* $o \in O_i$ *such that* $\forall o' \in O_i \; u_i(o_{-i}, o) \geq u_i(o_{-i}, o')$.

The concept that we now define was introduced in 1951 by John Nash in "Non-cooperative Games"[26]. In this paper he considers games in which players act independently without any means of communication, or collaboration. For that he introduces the concept of the *equilibrium point*, which is now known as the *Nash equilibrium*.

Definition 2.1.4 (Nash equilibrium)
A strategy $a = (o_1, \ldots, o_n)$ *is a* Nash equilibrium *iff* $\forall i \in \{1, \ldots, n\}$ o_i *is a best response to* o_{-i}.

To better understand these concepts consider the following example:

	C	R
C	$(1,1)$	$(5,0)$
R	$(0,5)$	$(4,4)$

Table 2.1: Prisoners Dilemma

Example 2.1.1
In this game, two players are described as criminals who were arrested after committing a robbery. The sentence for such a crime is 5 years in jail. There is no direct evidence, so there is no way to convict both of them for full time. They are separated and both are given an alternative: they can Cooperate and testify against their partner, or they can Refuse. If both refuse the partial evidence will be enough to lock them for 1 year. If both cooperate, then both will have their sentence reduced for one year. If only one cooperates he walks free, while the other serves full time. To be consistent with the notation used in this thesis, the value represents the number of years reduced from the sentence.

The obvious solution is for both of them to refuse. So the strategy vector for this is (R,R). Is R the best response for the first robber to the R of the second one? No. By changing his strategy to C he increases his value to 5, while reducing the second players value to 0. In the strategy vector (C,R), player 1 has indeed the best response for player 2, but this is not the case for player 2. His best response is C which increases his value to 1. In the case of (C,C) both players have used the best response to the other player strategy, therefore this is a Nash equilibrium.

This example may leave one wondering why it is reasonable to use the idea of best response, when in the result a definitely suboptimal solution, as the values of (C,C) are strictly lower than (R,R) for both players, is achieved. But this is a direct result of the assumptions about the players. Without any consequences for trying to get a better deal, or any reason to trust that the other player will not try to take advantage of you, there it is not reasonable not to try to play it safe and by chance even get a better result.

One of the large strengths of this idea is that Nash proved that any game with finite players and finite possible actions has a mixed strategy Nash equilibrium. This is not true for pure strategies. To illustrate that take the classical Rock-Paper-Scissors game as an example shown in Table 2.2.

Example 2.1.2
In this example it is clear that none pure strategy Nash equilibrium exists, as

	R	P	S
R	$(0,0)$	$(-1,1)$	$(1,-1)$
P	$(1,-1)$	$(0,0)$	$(-1,1)$
S	$(-1,1)$	$(1,-1)$	$(0,0)$

Table 2.2: Rock-Paper-Scissors Game

for any pair of actions picked, one of the players will not gain the value 1 and can switch to an action which gives him that value. On the other hand a mixed equilibrium by Nash theorem must exist, and it is easy to show that when both players distribute the probability equally between their actions that is the case.

2.2 Perfect Information Games

The Nash equilibrium is a very useful tool in analyzing normal form games, as it defines situations in which none of the players can gain anything by changing his strategy by himself. However, while it is possible to analyze sequential games in normal form, the model quickly becomes too large to handle. It is more convenient to describe such games as decision trees in which each node represents all of the moves made so far, and each edge represents the move the player makes. Such models are called *perfect information games*.

Definition 2.2.1 (Perfect information game)
A perfect information game *(in extensive form) is a tuple* $G = (N, O, H, Z, \chi, \rho, \sigma, \upsilon)$ *where:*

- N *is the set of players;*
- O *is a (single) set of objects;*
- H *is a set of non terminal choice nodes;*
- Z *is a set of terminal nodes, disjoint from* H*;*
- $\chi : H \to 2^O$ *is the object function, which assigns to each choice node a set of possible objects;*
- $\rho : H \to N$ *is the player function, which assigns to each non terminal node a player* $i \in N$ *who chooses an object in that node;*

- $\sigma : H \times O \to H \cup Z$ *is the successor function, which maps a choice and an object to a new choice node or terminal node such that* $\forall h_1, h_2 \in H \; \forall o_1, o_2 \in O \; (\sigma(h_1, o_1) = \sigma(h_2, o_2) \Rightarrow h_1 = h_2 \wedge o_1 = o_2)$*;*
- $\upsilon = (\upsilon_1, \ldots, \upsilon_n)$ *where* $\upsilon_i : Z \to \Re$*, is a real valued utility function for player* i *on the terminal nodes* Z*.*

In this form, it is not immediately obvious what a pure strategy is. The idea is that the player declarers what he would choose in any possible situation in which it is his move.

Definition 2.2.2 (Perfect information game pure strategies)
Let $G = (N, O, H, Z, \chi, \rho, \sigma, \upsilon)$ *be a perfect-information extensive-form game. Then the pure strategies of player* i *consist of the Cartesian product* $\Pi_{h \in H, \rho = i} \chi(h)$*.*

Having defined what a strategy for a player is it is possible to define what a strategy for a whole game is.

Definition 2.2.3
For a perfect information game G *any function* $s : H \to O$ *is called a* pure strategy.

The notions of best response and Nash equilibrium work fine with such a look on the strategy. Unfortunately, the notion of Nash equilibrium is not enough to declare if a given strategy is good in this case. This is because it is interested in the main result of a game and will not catch, if a player makes sub optimal moves in situations that are not supposed to happen, like if players before him make bad moves. To fix that the concept of a *subgame perfect equilibrium* is introduced, which demands that starting from any possible point of the game the strategies of the players will still form a Nash equilibrium.

Definition 2.2.4 (Subgame in a perfect-information game)
Given a perfect-information extensive-form game G, the subgame *of G rooted at node h is the restriction of G to the descendants of h. The set of subgames of G consists of all subgames of G rooted at some node in G.*

Definition 2.2.5 (Subgame perfect equilibrium)
The subgame perfect equilibria (SPE) of a game G *are all strategy profiles as such that for any subgame G' of G the restriction of s to G' is a Nash equilibrium of G'.The set of all strategies of game G that are* SPE *is denoted as* $SPE(G)$*.*

2.3 Stackelberg games

The next important concept appeared in 1934 with the book "Market Structure and equilibrium" by Heinrich von Stackelberg[43]. In this economy book Stackelberg reasons about free market, different types of market structures that can happen in it, and the relationships between them. Among the concepts described in it is the idea of a *leader advantage*. This is the leaders advantage which was later renamed as the Stackelberg strategy, which is the basis concept we use to model the problem in my thesis. This idea has been refined over years and can now be introduced using current game theoretical notations.

Definition 2.3.1 (Stackelberg Game)
A game is called a Stackelberg game if its set of players is divided in two subsets, one containing players that are called leaders *and the other containing* followers*. Each action of the followers is defined as a function $f : L \to O$ where L is the set of all possible strategy profiles of the attackers and O is the set of all possible objects. It is assumed that all of the leaders commit to a strategy before the followers. A strategy is called a* Stackelberg strategy *if it is a subgame perfect equilibrium with respect to the notion of followers actions.*

To better explain it we will use the following classical example.

	L	R
U	$(1, 1)$	$(3, 0)$
D	$(0, 0)$	$(2, 1)$

Table 2.3: Example

Example 2.3.1
Here the situation of a market is simplified to a normal form game between two players. The first player picks a row and the second pick a column. The values in the intersection give the payoffs to player 1 and 2 respectively. It is easy to see that, if no information is exchanged between players, player 1 is better of picking U instead of D. For both columns the value for player 1 is strictly larger in the upper row than in the lower row. That being the case player 2 should go for picking L as for him this gives a bigger payoff. This simplification is used to describe a situation in which two competing firms find themselves in a new market and after analyzing the possible approaches they can have to it and how

the actions taken by their competitors will affect the value that they can take from such market. Stackelberg argued that such a situation seldom takes place. When a new market possibility arises usually it is that one firm notices the opportunity, makes the proper analysis of the market and enters it, while his competitors follow him. To simulate that we will assume that player 1 is the one who can act on market first. If he stays with the strategy U once player 2 arrives he will pick L and player 1 will get a value of 1 from the market. But if he picks D, then when his competition arrives the correct choice will be R allowing the first player to get the value of 2. Furthermore if we allow players to pick mixed strategies, so that they can assign probabilities which they will use a certain row, or column the first player can choose to pick U row with the probability 0.5 minus some small epsilon, and the D row with the probability 0.5 plus that small epsilon. That way the expected value of picking column R for player 2 is slightly larger than for picking column L, which will incline him to do that, and secure an expected value of about 2.5 for player 1.

This example is good to illustrate the connections between Nash equlibiria and Stackelberg strategies. The situation described when both players moved simultaneously ended with player 1 choosing U and player 2 L. This was not a Stackelberg strategy, but it is a Nash equilibrium. If player 1 tries to place some probability on playing D then he will only lower his expected value, because we consider the situation where player 2 is tied to playing L. A symmetric argument can be made for the second player. This example also shows that a Stackelberg strategy does not necessarily have to be a Nash equilibrium. The pure strategy has a lower value then the mixed one for player 1, but in both cases player 2 makes the same decision, hence the pure case is not a Nash equilibrium.

There are cases where the leader advantage will leave the follower indifferent between a few possible actions. To reason with more precision about such cases Leitmann [20] purposed to consider strategies where such situations where played either to the leaders advantage, or disadvantage. The former turned out to be a stronger concept as it always exists and can be in practice obtained by the leader with small modifications to his strategy, making the indifference more favorable for the result he wants to obtain.

Definition 2.3.2 (Strong Stackelberg strategy)
A strong stackelberg strategy *in a Stackelberg game is a Stackelberg strategy which additionally assumes that when the follower is indifferent to a choice between possible actions he choose to the leaders advantage.*

2.4 Security games

Having all these concepts we have only to add what we understand a security game is.

Definition 2.4.1 (Security game)
A security game *is a game in which a subset of players is defined as* attackers *and the rest are defined as* defenders*. The valuation function has the property that the utility of an attacker of choosing an object that none of the defenders picked is greater or equal to the value of choosing the same object if also one of the defenders picked it. The utility of an defender of choosing an object that none of the attackers picked is less or equal to the value of choosing the same object if also one of the attacker picked it.*

Definition 2.4.2 (Stackelberg Security Game)
A Stackelberg security game *is game with the properties of both the Stackelberg game and the security game, where the set of attackers is also the set of the followers, and the set of defenders is the same as the set of leaders.*

Chapter 3

State of the art

In this chapter of my thesis we will go over various books and papers that are related to the concepts we will tackle on in this thesis. We will start by pointing to the discussion whether the leader advantage has use in real life situations, as it has been argued that without perfect information for the follower using the standard Nash equilibrium is more rasonable. We will see what has been proven so far for Stackelberg security games and how it has been applied. We will mention correlated equilibria which serve as the base to describe coordination efforts in games with selfish players. Finally we will briefly sum up what is known about computing pure Nash equilibria so far.

3.1 Stackelberg vs Nash discussion

In this section a short description of the potential problems of choosing Stackelberg over Nash equilibria will be discussed. The purpose of it is to highlight the potential real life problems that where brought to attention while trying to apply the Stackelberg model.

In 1995 Bagwell[4] questioned the value of committing to pure strategies given noisy observations by followers. He has shown that even with little imperfection when receiving information by the second player, in the case of pure strategies the first player loses his Stackelberg advantage, due to the fact that the receiving player has to take into account all possible variations of the signal, and the Stackelberg strategy will have to reduce itself to a Nash equilibrium.

This in turn was questioned by Huck and Muller's parer "Perfect versus Imperfect Observability. An experimental test of Bagwell's result"[15] in 2000, where

they provided four experimental games modeled on Bagwell's example with varying noise of the signal from the first player. In all cases experienced players did not support the findings of Bagwell.

Bagwell also argued that the noise in the signal does not effect the outcome as much if mixed strategies are allowed. Later studies showed if the leader keeps his advantage when considering mixed strategies in several cases. It has been extended further to the case of n-players. In 1998 Maggi [14] showed that the leader keeps his advantage with pure strategies, when considering games with pure information. Following this Morgan and Vardy [25] considered the case where information was not private, but costly to obtain.

The topic has been studied more thoroughly since. The paper "On the value of Commitment"[21] by Letchford, Korzhyk and Conitzer concentrates on finding a good method to compare the possible gains from committing

Another example is the paper "On the value of coleration"[2] by Ashlagi, Monderer and Tennenholz

3.2 Security Games

In this section several papers concerning security games will be presented, alongside with attempts to model them using game theory and the main theorems about mixed strategies they have proven.

In 2009 Pita, Jain, Ordonez and Tambe[33] conducted an experiment to check a variation of mixed defender strategies against human players. The result showed the superiority of computed defender mixed strategies.

In 2011 Korzhyk, Yin, Kiekintveld, Conitzer and Tambe[19] showed that the Nash equilibria in security games are interchangeable. They showed also that any Stackelberg strategy is also a Nash equilibrium strategy, when the attacker can only choose one target. Furthermore they showed that these observations do not hold if we allow the attacker to choose multiple targets.

Theorem 3.2.1
Suppose that (C,a) and (C',a') are two Nash equilibria in a strategy game G. Then (C,a') and (C',a) are also Nash equilibria in G.

If a game has equilibria with such property it is said that they are *interchangeable*. It is a very powerful property, as it allows to not worry which equilibrium is chosen. A variant of this property is also a part of how we define a good move in a pure strategy setting as we have shown in theorem 4.3.2.

Theorem 3.2.2
In security games,where the schedules are of size 1 any defender's Strong Stackelberg Equilibrium strategy is also a Nash equilibrium.

However these results do not hold true if the attacker has multiple resources. The intuition behind this is following. Assume that there is an object with a very high value for the defender, but the potential gain in utility is relatively small. In a SSE strategy the defender will still want to place a resource at that target to discourage the attacker from picking it. But in any Nash equilibrium the underlying assumption is that both sides commit simultaneously and the if there are other targets with relatively larger the attacker can pick them in addition, due to the multiple resources, and the defender would be better off committing to those instead.

In 2002 Tennenholtz[41] studied safety-level strategies. These security games find the strategies that maximize player 1 utility under the assumption that player 2 wants to minimize player's 1 utility.

Definition 3.2.1 (Safety level strategy)
Given a game G and a mixed strategy of player i t the safety level value obtained by i when choosing t in the game G is the minimal expected payoff that player i may obtain when employing t against arbitrary strategy profiles of the other players. A strategy t of player i for which this value is maximal is called a safely-level strategy.

Theorem 3.2.3
The optimal safety-level value for a player in the leader election game equals its expected payoff in the strictly mixed strategy equilibrium of that game.

Theorem 3.2.4
Given a 2-person set theoretic game G with a strictly mixed strategy Nash equilibrium, then the value of an optimal safety level strategy of a player equals its expected payoff in that equilibrium.

In the paper "Simulation and Game-Theoretic Analysis of an Attacker-Defender Game"[28] A. Nochenson and C.F. Larry Heimann describe an interesting case of security games in which they describe a network security game. The attacker tries to cause as much damage as possible while the defender tries to minimize the losses taking optimizing along the cost of defense. They use the concept to model the situation of a Chief Information Security Officer (CISO) from a game theoretic point of view.

The case where an attacker has more that one resource available was presented in 2011 in the paper "Security Games with Multiple Attacker Resources"[18] by Korzhyk, Conitzer and Parr. The more important results are the following

Theorem 3.2.5
Finding a Nash equilibrium in security games with multiple attacker resources is in P.

Theorem 3.2.6
Nash equilibria in security games with multiple attacker resources are interchangeable.

Theorem 3.2.7
Finding a Stackelberg strategy in security games with multiple attacker resources is NP-hard.

A good survey about what game theory has to offer security games was presented by Conitzer in 2012 in a paper "Computing Game-Theoretic Solutions and Applications to Security"[7]. It includes an overview of most used game forms, typical forms of solutions and a brief summary to their connection to security games.

In the paper "Defender (Mis)coordination in Security Games"[17] Jiang, Procaccia, Quain, Shah and Tambe reserched the case in which the security game was modeled as a Stackelberg game. In most of the research the underlying assumption was that there is only one defender who was access to all of the resources to allocate to defend potential targets. In case of multiple defenders the simplest way to go would be then to combine their resources and use a centralized to deploy them. This however was observed not to happen in real life situation. The purpose of this paper was to show how much of potential utility can be lost when multiple defenders allocate their resources simultaneously, and what are the losses if it happens in a sequence of moves.

Definition 3.2.2 (Price of miscoordination)
In a Stackelberg security game with d defenders assume that all defenders have the same payoffs. The maximum utility is achieved by the defenders when they pool their resources together, and use all their actions as if they were owned by a single defender. We call this strategy profile the optimal correlated profile (OCP) *as the mixed strategies of individual defenders are correlated. We define the* optimal uncorrelated profile (OUP) *as the profile of uncorrelated mixed strategies for the defenders that yields maximum utility among all profiles of uncorrelated mixed*

strategies. We define the supremum (over a given class of security games) of the ratio of the utility under OCP to that under OUP as the price of miscoordination (PoM).

Theorem 3.2.8
The PoM is unbounded in general security games.

Theorem 3.2.9
The PoM in security games with identical targets is at most $\frac{e}{e-1}$ and at least $frac43$.

Definition 3.2.3 (Price of sequential commitment)
In a Stackelberg security game with d defenders assume that all defenders have the same payoffs. We assume that the defenders commit sequentially. The first defender commits to a mixed strategy that optimizes the joint utility function in the absence of the other defenders. Subsequently, each defender chooses a mixed strategy that maximizes the joint utility function given the strategies of the earlier defenders, in the absence of the later defenders. The price under the best order of commitment *($PoSCb$) and the* price under the worst order of commitment *($PoSCw$) is the supremum (over all security games) of the ratio of the utility under the* OCP *to the maximum (resp. minimum) utility over all orders of sequential commitment.*

Theorem 3.2.10
The $POSC_b$ in security games with identical targets is unbounded.

Theorem 3.2.11
Denote the max-simultaneous-coverage by k. Then the $PoSC_w$ is O(k) in security games with identical targets and complete individual coverage.

The fact that even in the best possible scenario the ratio between coordinated effort and sequential commitment can be unbounded led to our search of what can we say if we restrict ourselves to consider only pure strategies, as opposed to the focus on mixed strategies which this paper was based upon.

Another paper in 2013 by Letchford, Korzhyk and Conitzer "On the value of commitment"[21] is dedicated to find how much a player can gain by committing to a specific strategy before other players. They study how much can be gained by commiting to a pure strategy, a mixed strategy. They also check what is the ratio between these two values to find how much more beneficial it is to go with a mixed strategy.

Definition 3.2.4 (Values of commitment)
Let C be a class of games. Let SC denote a solution concept where both of the players best respond under some information model (although the action they take might not look like a best response under a different information model) and let $u_1(SC(G))$ be the utility that player 1 receives under that solution concept in game G. Let S_1 (resp. Σ_1) be the set of player 1's pure (resp. mixed) strategies. (Of course, $S_1 \subset \Sigma_1$.) Let $\sigma_1 \in \Sigma_1$ be a strategy for player 1; let $G|_{\sigma_1}$ be the game among the remaining players that results after player 1 commits to σ_1. Then the value of pure commitment is defined as

$$VoPC(C) = \sup_{G \in C} \frac{\sup_{s_1 \in S_1}\{u_1(SC(G|_{s_1}))\}}{u_1(SC(G))}$$

the value of mixed commitment is defined

$$VoMC(C) = \sup_{G \in C} \frac{\sup_{\sigma_1 \in \Sigma 1}\{u_1(SC(G|_{\sigma 1}))\}}{u_1(SC(G))}$$

and the mixed vs. pure ratio

$$MvP = \sup_{G \in C} \frac{\sup_{\sigma_1 \in \Sigma 1}\{u_1(SC(G|_{\sigma 1}))\}}{\sup_{s_1 \in S_1}\{u_1(SC(G|_{s_1}))\}}$$

Theorem 3.2.12
For the class of perfect information extensive-form games

$$VoPC = VoMC = MvP = \infty.$$

Theorem 3.2.13
For the class of simple security games

$$VoPC = VoMC = MvP = \infty.$$

The authors note that by their findings commitment to a pure strategy is generally almost as beneficial to the leader as commitment to a mixed strategy, with the exception of zero-sum games.

3.3 Practical uses

The results of using Stackelberg model in security games has wide practical uses. Among them are the ARMOR program [32] which has been deployed since 2007

to schedule patrol routes and security checkpoints on the LAX airport. It uses a simple Stackelberg security game model in which the actions are assigning schedules, that is picking a set of objects (in this case roads where checkpoints are to be set) at one time, depending on available resources. It has later been refined [33] by applying algorithms that take into account the difference between true randomness and how people perceive what an random approach is.

The next use is the IRIS program [42] used by the US Federal Air Marshals. Again it relies on the Stackelberg security game model. There were additional challenges presented as the whole transportation network is enormous and there are several constraints on it that had to bee incorporated.

These applications were followed by the GUARDS program [34] which is tries to expand these models on a much larger scale. Instead of focusing on one place it reasons about hundreds of security activities and over diverse potential threats, allowing to use it efficiently over a set off potential targets instead of just one.

These applications do not end at securing buildings. The Stackelberg model has been also applied in dealing with pouching [11]. In this paper the authors generalize the Stackelberg strategy to situations where the potential attacker has no means to put good surveillance on the defender, and also a way to modify the defenders strategy upon observation where successful poaching has happened.

3.4 Coordination efforts

In this section what how coordinating agents in multiple agent systems can be beneficial to the combined output of the game will be discussed.

In 1974 Aumann's "Subjectivity and correlation in randomized strategies" [3] introduced an new concept in approaching the problem of maximizing the profit obtainable by selfish players by adding an subjective random device to help the players on deciding what strategies they should play. As the result one of the most important concepts of equilibrium in game theory was created. It has been used to add a way to model a benevolent administrator trying to coordinate between selfish players in such way that the overall utility is maximized, while allowing players to play rationally from their point of view, as seen for example in [5, 37]. This concept, although it was not possible to use it in a direct way, is what we are trying to simulate in the Overseer problem.

3.5 Findings on pure Nash equilibria

In this section what is currently known about the complexity of finding pure Nash equilibria will be presented.

These papers have a specific thing in common: all of them relay on using mixed strategies to find the best course of action. This is for various reasons. For one we know, thanks to the famous Nash theorem from 1951, that any game with a finite number of players and a finite number of possible actions will always have a mixed Nash-equilibrium. But this stops being the case when we consider a larger number of players. This also is not true if only pure strategies are allowed. In 1973 Robert W. Rosenthal [35] described a class of games, which is called congestion games, in which it was guarantied for a pure Nash equilibrium to exist. They were further studied by Monderer and Sharpley [24] who researched potential games and there equivalency to congestion games and proved the existence of pure Nash equilibria in some infinite potential games.

The complexity of finding Nash equlibria has also been broadly studied. While the existence of mixed-strategies Nash equilibria is given by the Nash theorem the question remained what is the complexity of finding one. In 1994 Papadimitriou [29] defined a new class of hard problems called PPAD and showed that finding a Nash equilibrium belonged to it. The class name stands for "polynomial parity argument for directed graphs" and consists of problems that can be reduced to the so called *end of the line* problem. This is a search problem in which the input consists of a directed graph and an unbalanced vertex in it, that is one in which has a different number of incoming edges than the number of outgoing edges, and gives as the output another unbalanced vertex, which is guarantied to exist.

The question about finding a pure Nash equilibrium returns to the NP class, as the existence of such equilibrium is not guarantied. While the concept of pure Nash equilibrium seems simpler the problem of fining one is proven to be NP-complete. Moreover several other questions connected to it are in this class.

In 2005 Gottlob, Greco and Scarcello published a paper "Pure Nash Equilibria: Hard and Easy Games" [13] in which they researched deeper on what restrictions can have influence on the complexity of this problem. The research that in games in standard normal form finding if there exists a Nash, Pareto and strong Nash equilibrium is possible in logarithmic space for pure strategies. They note that games in such representation could have an exponential size of valuation input with respect to the number of players and actions. They focus their attention on games in graphical normal form, as they found this form to be more practical in a computational point of view than the standard normal form. They found that some strong

restrictions, like the graphical normal form having bounded neighborhood, or the graph being acyclic are not enough move the problem from the NP-class. On the other hand a combination of weaker restrictions of having a small neighborhood and bounded hypertree width will take the problem to the polynomial complexity.

Chapter 4

Obtained Results

4.1 Problem Statement

A lot has been done in the analysis of Stackelberg games with mixed strategies. This work focused on less studied case in which only pure strategies are available to players. Furthermore we assume that the defenders do not pick their strategy simultaneously, but also commit to a strategy in a given order.

Problem 4.1.1 (Main Problem)
Consider a Stackelberg game in which there are n defenders and one attacker. All players have full information about the values of each player. Only pure strategies are permitted. The defenders commit to a strategy in a previously defined and known order. Do strategies that maximize the value for each player in reasonable time can be found?

This case has been studied for mixed strategies, but no result is known if only pure strategies are allowed to be used. There is an interesting observation here to make. By allowing only pure strategies the attacker is guaranteed to pick one of the objects not chosen by the defenders. As there is no uncertainty he can pick whatever is most valuable for him after knowing how the defenders will commit. Thus he can be ignored in the problem. Also the concentration can remain on the values of objects after being defended, which simplifies this problem to the following.

Observation 4.1.1 (An equivalent problem)
Consider a set of n players and a common set of m objects. Each player has a value assigned to each object. The players know each others valuation of objects.

The players are assigned in an order and one by one each pick one of the objects. The valuation of the result for a players is the sum of values of all objects that have been picked. How should each player decide which object to pick?

Finding the whole strategy for a game in the perfect information model is unfeasible as the whole strategy description is exponential in respect to the information given by the players. We define another way to check if a given move is good in the sense that we will not look for the whole strategy, but find if a good strategy exists where such a move should be made.

Definition 4.1.1
Consider a game G with the set of objects O and a strategy s for G. A sequence $(o_1, ..., o_n) \in O^n$ is a result *of strategy s if, and only if starting in the root of the strategy and moving down an edge only if the label of the edge is the same as the label of the current vertex, the labels of the edges traveled through form the sequence $(o_1, ..., o_n)$.*

Definition 4.1.2
We say a sequence of actions $(o_1, ..., o_n)$ is called reasonable *if there exists a strategy profile $s \in SPE(G)$ such that $(o_1, ..., o_n)$ is a result of s.*

4.2 Algorithms for decisions

The idea behind the algorithm is the following: a player has no influence on what the players before him will pick. They make their decisions on what is available and what they expect the rest of the players to choose. No form of communication is available, so no form of bargaining or threat is possible. Each player knows the valuation of objects of the rest of the players, so he has a base to predict how the players after him will decide. Thus we can assume a player will not pick an object he has reasons to believe a player after him will choose. To make this prediction he will assume that the last player will pick the most valuable object from his point of view, as this is what all players assume. Then, for each previous player, he assumes that they will pick their most valuable object from the set of objects which excludes any that has already been predicted.

Algorithm 1 The basic algorithm

Input: O - set of available objects;
V - the valuation matrix;
i - the index of the player making the decision;
Output: $(o_i, ..., o_n)$ the predicted choices of objects for players i to n.

1: Delete all columns for objects that have already been chosen.
2: Define k as the number of rows in the matrix.
3: Find in the last row the column in which there is the most valuable object for the k-th player (if more then one pick at random).
4: Mark this object as o_k.
5: Remove the last row from the matrix.
6: Repeat steps 1-4 until o_i is defined.

To prove that the strategy is SPE we will need to consider all possible moves the players can make. This can make it difficult to keep the original prediction a player had, as running the algorithm on a smaller set of available objects (especially when there was uncertainty involved) can lead to a completely different sequence of predicted choices. The next algorithm is necessary to keep control when we have a sequence of predicted choices and get a smaller set of possible objects.

Algorithm 2 The modified algorithm

Input: O - set of available actions;
V - the valuation matrix;
i - the index of the player making the decision;
$(o_1, ..., o_n)$ - sequence of choices of objects predicted by the original algorithm;
Output: $(o'_i, ..., o'_n)$ the predicted choices of objects for players i to n.

1: Delete all columns for objects that have already been chosen.
2: Define k as the number of rows in the matrix.
3: Find in the last row the column in which is the most valuable object for the k-th player (if more than one and o_k is available pick o_k else pick at random).
4: Mark this object as o'_k.
5: Remove the last row from the matrix.
6: Repeat steps 1-4 until o'_i is defined.

Example 4.2.1

To illustrate described algorithm in action, this example will consider a game in which each player in his move will pick a city to defend. The goal of this example is to predict second player choice of move.

Input:
O= {Cracow , London, New York , Budapest , Berlin}
i= index of player making the decision.
quantity of players= 4
V - valuation matrix is shown in table below.

Players/Objects	*Cracow*	*London*	*New York*	*Budapest*	*Berlin*
1	*10*	*12*	*18*	*20*	*15*
2	*18*	*20*	*20*	*10*	*16*
3	*20*	*19*	*12*	*16*	*17*
4	*20*	*15*	*20*	*18*	*14*

The second player choice will be predicted according to algorithm steps.

1. *No columns have been chosen yet, so none of them will be deleted.*
2. *k=4*
3. *For fourth player both Cracow and New York are most valuable objects. Picking at random from those two has returned New York.*
4. o_4 *= New York*
5. *The fourth row was removed from the matrix*

Players/Objects	*Cracow*	*London*	*New York*	*Budapest*	*Berlin*
1	*10*	*12*	*18*	*20*	*15*
2	*18*	*20*	*20*	*10*	*16*
3	*20*	*19*	*12*	*16*	*17*

6. o_2 *was not defined, repeat steps 1-5.*

1. *Delete column containing New York,*

Players/Objects	*Cracow*	*London*	*Budapest*	*Berlin*
1	*10*	*12*	*20*	*15*
2	*18*	*20*	*10*	*16*
3	*20*	*19*	*16*	*17*

2. *k=3*
3. *For third player Cracow is most valuable object.*
4. o_3*=Cracow*
5. *The third row was removed from the matrix*

Players/Objects	*Cracow*	*London*	*Budapest*	*Berlin*
1	*10*	*12*	*20*	*15*
2	*18*	*20*	*10*	*16*

6. o_2 *was not defined, repeat steps 1-5.*

1. *Delete column containing Cracow,*

Players/Objects	*London*	*Budapest*	*Berlin*
1	*12*	*20*	*15*
2	*20*	*10*	*16*

2. *k=2*
3. *For second player London is most valuable object.*
4. o_2*=London*
5. *The second row was removed from the matrix*

Players/Objects	*London*	*Budapest*	*Berlin*
1	*12*	*20*	*15*
2	*20*	*10*	*16*

6. o_2 *was defined, end algorithm.*

Output*: {*o_2*= London,* o_3*= Cracow,* o_4*= New York}*

4.3 The main theorem

To show that we can construct a full SPE strategy we will use the following lemma, which shows that if we have a predicted sequence and we remove one object from the set of possible objects, then using the modified algorithm 2 for a player will have an outcome which will differ from the original outcome at most at one choice.

Lemma 4.3.1
Let $(o_1, ..., o_n)$ be the result of using the algorithm 1 on the game G with the set of objects O. By running the modified algorithm 2 for the game G, sequence $(o_1, ..., o_n)$ and set of objects $O \setminus \{o\}$, where $o \in O$, will give a sequence $(o'_1, ..., o'_n)$ which will differ from $(o_1, ..., o_n)$ in at most one element and only if $o \in \{o_1, ..., o_n\}$.

Proof. Case 1

First let us consider the case in which $o \notin \{o_1, ..., o_n\}$. In this case the resulting sequence will be identical to the original. Each player has a smaller set of objects to choose from, but neither of them has chosen o in the first run of the algorithm. In the modified version if there is an uncertainty which object to choose they should pick the same object as in the original run. Because removing an element from the set of available objects does not change their values for the players and the algorithm picks the previous choice if it is still one of the most valuable ones, none will change commitment.

Case 2

Now consider the case in which $o \in \{o_1, ..., o_n\}$. Let us assume that $o = o_k$ for some $1 \geq k \geq n$. Since none of the values have changed and the players form $k + 1$ to n will still have their previous choices available, the algorithm will pick those objects . Of course player k cannot choose o_k because it is unavailable for anyone. The modified algorithm finds a new object for him which we will mark as o'. It cannot be that $o' = o_i$ for $i > k$ because these objects are already unavailable for the algorithm by now. If $o' \neq o_i$ for all $i < k$ then this will be the only change result of the algorithm, because then all of the objects the modified algorithm has to pick in case of ties are available. If $a' = o_i$ for some $i < k$ then it will be assigned to player k. Running the modified algorithm for players from $i + 1$ to $k - 1$ will give the same result as the original, by the same reasoning as before. For player i we can repeat the same reasoning from the whole Case 2. As in each such repetition the index will get smaller and the sequence is finite, such replacement will happen only a finite number of times and will result in a sequence which differs only in one element from the original sequence. □

With this we can prove the following theorem.

Theorem 4.3.1
Let $(o_1, ..., o_n)$ be the result of using the algorithm on the game G with the set of objects O. Then $(o_1, ..., o_n)$ is reasonable.

Proof. We will prove this by constructing the full PSE strategy tree for the game G for which $(o_1, ..., o_n)$ is the result. The proof goes by induction on the number of players. The case of $n = 1$ is trivial.

$n = 2$] In this case constructing the strategy tree is fairly simple. We have one vertex corresponding to the decision of player 1 from which descent m edges corresponding to all possible actions for player 1. We put o_2 on all verticies of player 2 except the one connected to the edge o_2. There we can use the modified algorithm on the sequence (o_1, o_2) and the set of objects $O\backslash\{o_2\}$ to find one to put on this vertex. We put o_1 on the root. It should be easy to see that this is a PSE strategy tree with the sequence (o_1, o_2) as a result.

$n - 1 \Rightarrow n$

Now we assume that we can build a strategy tree for any game G with $n - 1$ players a given set of objects O which is PSE and a sequence given by the algorithm is the result. We will show how to use this to construct the strategy tree for any game with n players and a sequence $(o_1, ..., o_n)$ given by the algorithm. We start from a strategy tree for the game G with all vertices empty. For every vertex connected to the root we will run the modified algorithm on the game G without player 1, sequence $(o_2, ..., o_n)$, and set of objects $O\backslash\{o\}$, where o is the label of the edge between this vertex and the root. By our inductive assumption, we can construct a full strategy on the subtree starting from that vertex, which is PSE and has $(o'_2, ..., o'_n)$, given by the modified algorithm, as a result. We can see that by discarding the object o for this whole subtree we can be sure that, as long as we pick the proper object for the root, the whole strategy will stay PSE. What is left is to show that there are no better actions to put at the root than o_1. Consider first edges o which are not in the set $\{o_2, ..., o_n\}$. If player one was to pick one of them the result of playing the subtree under that edge, by the lemma, is exactly the sequence $(o_2, ..., o_n)$, so it only could be beneficial for him if $v_1^o > v_1^{o_1}$ which is contrary to the way we picked o_1. Suppose now that player 1 could benefit from committing to an action a form the set $\{o_2, ..., o_n\}$. By the lemma the resulting sequence $(a, o'_2, ..., o'_n)$ differs in at most one element from the sequence $(o_1, ..., o_n)$. If it differs, then for it to be beneficial it had to be the case that this one action has a greater value for player 1 than o_1, which is in contrary with the way o_1 was chosen. So there is no action which grants a better result for player

1 than choosing o_1. We put o_1 on the root getting a PSE strategy tree with the result $(o_1, ..., o_n)$ thus completing the construction. □

With multiple preferred objects it could happen that the whole outcome of the game is very different from what the players predicted and in fact the outcome does not have to be a SPE, which would undermine the validity of the reasonable move as a good strategy concept. The next theorem proves that no matter how often the players were wrong in their predictions the whole outcome will be in fact reasonable.

Theorem 4.3.2
Let G be a game with n players and the set o of available objects. Player 1 uses the algorithm to obtain the sequence $(o_1^1, ..., o_n^1)$ and picks o_1^1. Then player 2 uses the algorithm on the set $O\backslash\{o_1^1\}$, obtains the sequence $(o_1^2, ..., o_{n-1}^2)$ and commits to o_1^2. The following players continue in a similar fashion cutting the set of objects. Then the sequence $(o_1^1, o_1^2, ..., o_1^n)$ is reasonable.

Proof. The proof goes by induction on the number of players. The case $n = 1$ is trivial.

$n = 2$

As in the previous proof we have one vertex corresponding to the decision of player 1 from which descent m edges corresponding to all possible actions for player 1. W put o_1^1 on the one vertex of player 1. We put o_1^2 on all vertices of player 2 except the one connected to the edge o_1^2. We use the modified algorithm for the sequence (o_1^1, o_1^2) and the set of objects $O\backslash\{o_1^2\}$ to find what to place on the last vertex. This strategy will have (o_1^1, o_1^2) as a result. As to show that the strategy is PSE it suffices to notice that even if $o_1^2 \neq o_2^1$ both must be equally valued by player 2 because the algorithm gave those two elements as a possible move of player 2 on two different occasions, while both those actions where available to the player.

$n - 1 \Rightarrow n$

We assume that we can build a strategy tree for any game G with $n-1$ players and a given set of actions A which is PSE and the proper sequence is the result. To show the result for n players we start with a game tree with all vertices empty. For every vertex connected to the root we run the modified algorithm on the game G without player 1, sequence $(o_1^2, ..., o_1^n)$ and the set of actions $O\backslash\{o\}$, where o is the label of the edge between this vertex and the root. By the inductive assumption the sequence$(o_1^2, ..., o_1^n)$ is reasonable for the proper subgame, so the result of the modified algorithm is also reasonable and a PSE strategy can be constructed

on this subtree. It remains to argue that after putting o_1^1 in the root the strategy we remain PSE. It is important to notice that the sequence $(o_1^1, o_1^2, ..., o_1^n)$ is a possible result of using the regular algorithm for the game G with n players and set of objects O. Thus we can use the lemma for all the subtrees. So we can use the exact same argument as in the proof of Theorem 5.1 to show that player 1 cannot benefit from changing committing to another move than o_1^1.

□

This proof is also beneficial in the sense that it solves the problem even if the players have more that one move available, under the condition that they have to use all their moves in one turn. It is easy to see that in such case we can replace each player with a set of dummy players, each of which has one move, and that in the sequence of play all exactly where the original player was relatively to other players. The dummy players have their preferences identical to their original player. It is easy to see that an reasonable outcome in such game would give a good strategy to play in the original case as no new interference between players is introduced. This however is not necessarily true if the players can make moves in different parts of the sequence. Further investigation to that matter is required.

4.4 The Overseer

We will now bring a slight modification to our game. We add a new player to the game which we will call the Overseer who has the following properties: he has a set of objects which is a subset of all available objects, which we will call the *goal*, knows the valuation of objects for all the players, and moves first and instead of picking an object he decides the order in which the other players will play. His victory condition is to pick such a sequence of players that all objects from the goal will be selected during the rest of the game. The question whether such a sequence of players exist we call the *Overseer problem*.

First we will show that it is enough to consider only the case in which the goal is only one object.

Lemma 4.4.1
The complexity of solving the overseer problem with the goal set G of any size is the same as solving the overseer problem with a goal set of size 1.

Proof. Obviously we if we can solve the overseer problem for G of any size we can solve it for a goal set of size 1. Let us assume that we have an algorithm to

solve the problem for sets of size 1. Let us take a game G with n players, set of objects O and an overseer with a goal set G. To find a solution we will use an extended version of this game G' with $n+1$ players the set of objects $O \cup \{\alpha\}$ and an overseer with the goal set $\{\alpha\}$. The preferences of the original n players remain unchanged, but α is placed at the end of their preference list. The preferences of the new player are following: first he values the elements of the goal set G then α then all the elements of the set $O \backslash G$. We can see that the overseer problem in this case has a solution if and only if there was a solution in the original problem. □

This is clearly a decision problem which in the wort case is in NP, as checking the outcomes for all possible sequences of players takes $n!$ runs of our previous polynomial result. The next theorem shows a partial result in which the Overseer problem is in P.

Theorem 4.4.1
The overseer problem with the additional assumption, that for each player the goal is his n-th preferred object, where n is the number of players, is in P.

Proof. The size of the input in this problem will be n^2, where n is the number of players. First we have to observe that any solution will have to use all players involved. Indeed if one of the players is to pick up the goal and each player can pick only one object, then all of the more preferred objects must have been picked by other players. The algorithm to check if there is a solution for a given player goes as follows:

We make a bipartite graph of size $(n-1, n-1)$ where one side of the graph represents the rest of the players and the other represents all the objects that have to be picked, before the chosen player will pick the goal. We will calls these objects *desired*. For each of the other players we do the following: We start with their most preferred object and check if it is desired. If so we add an edge between the node representing that player and the node representing the desired object. If not we stop and go to the next player. If there was an edge added repeat this check for the next preferred object, until we are forced to move to the next player. After using all players we get a graph with a following property: There is a traversal in this graph if, and only if there is an ordering of other players such that each desired object will be picked.

First let us assume that there is an ordering in which each player picks one desired object. If for each such pairing in the graph there is an edge, this would give us the traversal. Assume now that for one pairing such edge does not exist. That would mean that he had an object more preferred that the one he picks which

is not an desired object. The only way he can pick the object he is assigned to is if someone else picks that undesired object. But this is not possible as there is a solution only if all players pick desired objects.

Now assume that we have a traversal in our graph. We will use it to construct an ordering of players in which each picks one desired object. We will now say that a player want to exchange with another, if he has an more preferred object than the one assigned by the traversal, that is assigned to the other player by the traversal. We search the graph for cycles of players such that the first wants to exchange with the second, the second with the third,..., and the last with the first. We switch a assignments accordingly so that each player is now assigned to a preferable object and the pairings are still a traversal. We continue to do so until no further cycles are to be found. The end result is what is needed to find the correct ordering. First we put all of the players for which their assigned object is the one they most prefer at the end of the order.

We will show now that at least one such player must exist. Assume that there is none. We will show that we can construct a sequence of players in which there will be a cycle of players that want to exchange. Let us start with picking one player at random. We check his most preferred object and look at the player that has that object assigned to him and place him after the first. Then we look on his most preferred object and repeat. We continue to do so until we return to a player that is already in the sequence. Starting from that player the sequence would give a cycle that is not possible after changing our traversal.

Now we remove all objects that are assigned to the players in the ordering. After the removal we place all the players that now have their assigned object being their most preferred one before all players already assigned to the ordering. At least one must exist by following the previous reasoning. We continue to repeat this process until all players are assigned, thus getting the ordering we were looking for.

Finding whether a graph has a traversal is polynomial. The construction of the graph requires no more than n^3 checks. We can repeat this construction n times to check whether any of the players can get the goal. If at least one can the the algorithm returns a yes, else it returns a no. This requires at most n uses of a polynomial algorithm, so this case of the overseer problem is in P. □

Chapter 5

Conclusions

5.1 Summary

We have simplified the problem of security games with pure strategies to a problem with players having the same role and an additive valuating function. We modeled this as a Perfect Information game to find a way to argue what a strategy is, and when it is good. We proposed a simple algorithm to find a base on which a player should make his decision. We proved that using this algorithm gives a result similar to finding a whole strategy for a game without having to explicitly state what the whole strategy is. We have proven also that the outcome of a game, even if not the same as the first prediction, is still an outcome of a good strategy, without having to show the whole decision tree.

We introduced a new player with a role to coordinate other players to achieve his goal. We proved that the size of his goal does not affect the complexity of deciding whether his achieving goal is possible. We have proven that in a special case the answer can be found in polynomial time. Moreover, if the answer is yes, we can find the sequence of players that satisfies the Overseers goal in polynomial time.

5.2 Future Work

Knowing all that has been done there are still several things that can be asked. First thing among them is whether the Overseer problem in the general case is in NP, or not. The proof that the special case is P heavily relies on the fact that we know which object have to be used beforehand which allows to construct a

bipartite graph of dependencies. This has been hard to use in the general case and no method other that a brute force search through all possible sequences of players has given satisfying results.

Another question to ask is what will happen if the players have more resources (can pick more than one object). In the case where they have to allocate them all at once we can use the result we have by adding an appropriate number of dummy players which have the same valuation of objects as the player they represent and keeping then next to that player in the sequence. The problem gets harder if we allow the players to pick their objects several times in different parts of the sequence, as the method used in our proof allows no coordination between players. The special case of the Overseer problem could still be in P if we allow player to commit in different parts of the sequence, as we could construct a similar bipartite graph, if we know how the players will choose their objects depending on the order there in. It is not clear if the special case would still be in P if a player has to pick their objects all at once, as the method used to prove it at he moment could require a player to commit in different parts of the sequence.

Finally we have no results describing the case in which the players can communicate and coordinate with each other. If they can commit only to one object then the result would not change, as they have no leverage to influence each others moves, but if we allow them to pick more such possibilities arise. Moreover any result in this direction could make it easier to solve the Overseer problem with multiple objects, as we could treat a player with multiple moves as several that coordinate with each other perfectly.

Bibliography

[1] Bo An, Milind Tambe, Fernando Ordonez, Eric Anyung Shieh, and Christopher Kiekintveld. Refinement of strong stackelberg equilibria in security games. In *AAAI*, 2011.

[2] Itai Ashlagi, Dov Monderer, and Moshe Tennenholtz. On the value of correlation. *Journal of Artificial Intelligence Research*, 33:575–613, 2008.

[3] Robert J Aumann. Subjectivity and correlation in randomized strategies. *Journal of mathematical Economics*, 1(1):67–96, 1974.

[4] Kyle Bagwell. Commitment and observability in games. *Games and Economic Behavior*, 8(2):271–280, 1995.

[5] Dietrich Braess, Anna Nagurney, and Tina Wakolbinger. On a paradox of traffic planning. *Transportation Science*, 39(4):446–450, 2005.

[6] George Christodoulou and Elias Koutsoupias. The price of anarchy of finite congestion games. In *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, pages 67–73. ACM, 2005.

[7] Vincent Conitzer. Computing game-theoretic solutions and applications to security. In *AAAI*, 2012.

[8] Constantinos Daskalakis, Paul W Goldberg, and Christos H Papadimitriou. The complexity of computing a nash equilibrium. *SIAM Journal on Computing*, 39(1):195–259, 2009.

[9] Constantinos Daskalakis and Christos H Papadimitriou. Computing pure nash equilibria in graphical games via markov random fields. In *Proceedings of the 7th ACM conference on Electronic commerce*, pages 91–99. ACM, 2006.

[10] Fei Fang, Albert Xin Jiang, and Milind Tambe. Protecting moving targets with multiple mobile resources. *Journal of Artificial Intelligence Research*, 48:583–634, 2013.

[11] Fei Fang, Peter Stone, and Milind Tambe. When security games go green: Designing defender strategies to prevent poaching and illegal fishing. In *IJCAI*, pages 2589–2595, 2015.

[12] Paul W Goldberg and Christos H Papadimitriou. Reducibility among equilibrium problems. In *Proceedings of the thirty-eighth annual ACM symposium on Theory of computing*, pages 61–70. ACM, 2006.

[13] Georg Gottlob, Gianluigi Greco, and Francesco Scarcello. Pure nash equilibria: hard and easy games. *Journal of Artificial Intelligence Research*, 24:357–406, 2005.

[14] Gene M Grossman and Giovanni Maggi. Diversity and trade. Technical report, National bureau of economic research, 1998.

[15] Steffen Huck and Wieland Müller. Perfect versus imperfect observability—an experimental test of bagwell's result. *Games and Economic Behavior*, 31(2):174–190, 2000.

[16] Albert Xin Jiang and Kevin Leyton-Brown. Computing pure nash equilibria in symmetric action graph games. In *AAAI*, volume 1, pages 79–85, 2007.

[17] Albert Xin Jiang, Ariel D Procaccia, Yundi Qian, Nisarg Shah, and Milind Tambe. Defender (mis) coordination in security games. AAAI, 2013.

[18] Dmytro Korzhyk, Vincent Conitzer, and Ronald Parr. Security games with multiple attacker resources. In *IJCAI Proceedings-International Joint Conference on Artificial Intelligence*, volume 22, page 273, 2011.

[19] Dmytro Korzhyk, Zhengyu Yin, Christopher Kiekintveld, Vincent Conitzer, and Milind Tambe. Stackelberg vs. nash in security games: An extended investigation of interchangeability, equivalence, and uniqueness. *J. Artif. Intell. Res.(JAIR)*, 41:297–327, 2011.

[20] George Leitmann. On generalized stackelberg strategies. *Journal of Optimization Theory and Applications*, 26(4):637–643, 1978.

[21] Joshua Letchford, Dmytro Korzhyk, and Vincent Conitzer. On the value of commitment. *Autonomous agents and multi-agent systems*, 28(6):986–1016, 2014.

[22] Kevin Leyton-Brown and Yoav Shoham. Essentials of game theory: A concise multidisciplinary introduction. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 2(1):1–88, 2008.

[23] Marios Mavronicolas, Vicky Papadopoulou, Anna Philippou, and Paul Spirakis. A network game with attackers and a defender: A survey. In *CD ROM Proceedings of the 2nd European Conference on Complex Systems*, 2006.

[24] Dov Monderer and Lloyd S Shapley. Potential games. *Games and economic behavior*, 14(1):124–143, 1996.

[25] John Morgan and Felix Várdy. The value of commitment in contests and tournaments when observation is costly. *Games and Economic Behavior*, 60(2):326–338, 2007.

[26] John Nash. Non-cooperative games. *Annals of mathematics*, pages 286–295, 1951.

[27] Thanh Hong Nguyen, Rong Yang, Amos Azaria, Sarit Kraus, and Milind Tambe. Analyzing the effectiveness of adversary modeling in security games. In *AAAI*, 2013.

[28] Alan Nochenson and CF Larry Heimann. Simulation and game-theoretic analysis of an attacker-defender game. In *International Conference on Decision and Game Theory for Security*, pages 138–151. Springer, 2012.

[29] Christos H Papadimitriou. On the complexity of the parity argument and other inefficient proofs of existence. *Journal of Computer and system Sciences*, 48(3):498–532, 1994.

[30] Christos H Papadimitriou and Tim Roughgarden. Computing correlated equilibria in multi-player games. *Journal of the ACM (JACM)*, 55(3):14, 2008.

[31] Praveen Paruchuri, Jonathan P Pearce, Janusz Marecki, Milind Tambe, Fernando Ordonez, and Sarit Kraus. Playing games for security: An efficient exact algorithm for solving bayesian stackelberg games. In *Proceedings of*

the 7th international joint conference on Autonomous agents and multiagent systems-Volume 2, pages 895–902. International Foundation for Autonomous Agents and Multiagent Systems, 2008.

[32] James Pita, 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*, pages 125–132. International Foundation for Autonomous Agents and Multiagent Systems, 2008.

[33] James Pita, Manish Jain, Fernando Ordóñez, Milind Tambe, Sarit Kraus, and Reuma Magori-Cohen. Effective solutions for real-world stackelberg games: When agents must deal with human uncertainties. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*, pages 369–376. International Foundation for Autonomous Agents and Multiagent Systems, 2009.

[34] James Pita, Milind Tambe, Chris Kiekintveld, Shane Cullen, and Erin Steigerwald. Guards: game theoretic security allocation on a national scale. In *The 10th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*, pages 37–44. International Foundation for Autonomous Agents and Multiagent Systems, 2011.

[35] Robert W Rosenthal. A class of games possessing pure-strategy nash equilibria. *International Journal of Game Theory*, 2(1):65–67, 1973.

[36] Tim Roughgarden. Stackelberg scheduling strategies. *SIAM Journal on Computing*, 33(2):332–350, 2004.

[37] Tim Roughgarden and Éva Tardos. How bad is selfish routing? *J. ACM*, 49(2):236–259, March 2002.

[38] Eric Shieh, 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*, pages 13–20. International Foundation for Autonomous Agents and Multiagent Systems, 2012.

[39] Eric Shieh, Manish Jain, Albert Xin Jiang, and Milind Tambe. Efficiently solving joint activity based security games. In *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, pages 346–352. AAAI Press, 2013.

[40] Milind Tambe. *Security and game theory: algorithms, deployed systems, lessons learned*. Cambridge University Press, 2011.

[41] Moshe Tennenholtz. Competitive safety analysis: Robust decision-making in multi-agent systems. *Journal of Artificial Intelligence Research*, 17:363–378, 2002.

[42] Jason Tsai, Christopher Kiekintveld, Fernando Ordonez, Milind Tambe, and Shyamsunder Rathi. Iris-a tool for strategic security allocation in transportation networks. 2009.

[43] Heinrich Von Stackelberg. *Market structure and equilibrium*. Springer Science & Business Media, 2010.