State of the Art on: Abstractions in Extensive-Form Games

Jacopo P. Gargano, jacopopio.gargano@mail.polimi.it

1. Introduction to the research topic

The area of focus of our research is Algorithmic Game Theory, a field of study that aims to analyze strategic conditions and to design algorithms able to find strategies for the involved agents allowing them to reach an *equilibrium*[1]. A strategic environment is mathematically modeled through formal representations so as to describe its problems and solutions, if any. Solving a strategic problem requires the use of the theory of computation and algorithm design: the former to analyze the problem complexity and evaluate its *difficulty*[2], the latter to solve the problem - usually corresponding to finding *equilibria*[1]. Therefore, Algorithmic Game Theory is a combination of Mathematics, specifically Game Theory, and of Computer Science.

The problem of analyzing *abstractions*[1] in strategic games is related to specific fields of Computer Science: Artificial Intelligence and Machine Learning. When it comes to generating *traces*[1] so as to model abstractions, Artificial Intelligence concepts like *exploration* and *exploitation* come into play. Moreover, Machine Learning has its contribution through *clustering* algorithms and *online learning*[1]. Online Convex Optimization is crucial as most of the problems faced by Game Theory are modeled as function optimization problems - usually as the minimization of convex functions over convex sets. Finally, Theoretical Computer Science is fundamental to analyze both space and time complexity of a problem, specifically of its representation or of the algorithms proposed to solve it.

Our research will be applied to Security, which is a critical concern around the world that arises in problems ranging from physical to cyberphysical systems. Security mainly deals with the problems of recognizing malicious agents and threats, allocating the available, usually limited, security resources and misleading potential attackers. Game Theory is well-suited to adversarial reasoning for security resource allocation and scheduling problems [30].

Conferences and Journals

Although *abstractions in extensive-form games* lies at the intersection of multiple research areas, the most pertinent conferences and journals are those related to Artificial Intelligence, specifically those focused on Algorithmic Game Theory. We evaluated conferences and journals using several factors to identify the most relevant ones with respect to our research. The adopted criteria took into consideration the following:

- GGS[3] and Microsoft Academic rankings[4] to evaluate the quality of conferences
- IF[5] and Microsoft Academic rankings[6] to evaluate the quality of journals
- Acceptance rate[7]
- Number of influential articles and authors in the field[8]
- Opinion of researchers working in the field

[1] See Section 1.1
[2] In terms of computational complexity (e.g. NP-hardness).
[3] The GII-GRIN-SCIE Conference Rating, 2018, `http://gii-grin-scie-rating.scie.es/conferenceRating.jsf`
[4] `https://academic.microsoft.com/conferences/`
[5] *Impact Factor*: the number of citations received in that year of articles published in a specific journal during the two preceding years, divided by the total number of publications in that journal during the two preceding years – higher is better
[6] `https://academic.microsoft.com/journals/`
[7] Lower is better.
[8] Higher is better.

The most relevant conferences with respect to *abstractions in extensive-form games* and their relative research areas are:

- *AAAI: Association for the Advancement of Artificial Intelligence* – Artificial Intelligence
- *NIPS: Neural Information Processing Systems* – Artificial Intelligence
- *IJCAI: International Joint Conference on Artificial Intelligence* – Artificial Intelligence
- *AAMAS: Adaptive Agents and Multi-Agents Systems* – Game Theory
- *CDC: Conference on Decision and Control* – Game Theory
- *ACM EC: Conference on Economics and Computation* – Game Theory, Theoretical Computer Science

The most relevant journals with respect to *abstractions in extensive-form games* and their relative research areas are:

- *Artificial Intelligence* – Artificial Intelligence
- *arXiv: Artificial Intelligence* – Artificial Intelligence, Planning
- *Journal of Artificial Intelligence Research* – Artificial Intelligence
- *Games and Economic Behavior* – Game Theory
- *International Journal of Game Theory* – Game Theory
- *Algorithmica* – Theoretical Computer Science

1.1. Preliminaries

Game Theory consists in the study of mathematical models of conflict and cooperation between intelligent rational[9] decision-makers. It provides the general mathematical techniques for analyzing situations in which two or more individuals make decisions that will influence their welfare [26].
The main concepts related to Algorithmic Game Theory, specifically games, their representations and abstractions, are hereby presented.

1.1.1 Games and Representations

Definition 1 – Game *A* game *is a process consisting in:*

- *A set of players;*
- *An initial situation;*
- *Rules that players must follow;*
- *All possible final situations – the outcomes;*
- *The preferences of all players – the utilities.*

Definition 2 – Sequential Game *A* sequential game *is a game in which players play in succession, taking turns.*

The majority of all real-world strategic games are imperfect information perfect recall sequential games. Formally, they are represented by extensive-form games. A game tree is the most common graphical representation of a sequential game.

[9] *Rational agent*: an agent is said to be rational when they are able to use decision theory, that is, choosing the alternatives that provide them the maximum possible utility, which is a measure of their preferences over the outcomes of the game.

Definition 3 – Game Tree *A* game tree *is a triple* (V, E, x_0) *where* (V, E) *is an oriented graph,* V *the set of vertices and* E *the set of edges, and* x_0 *(the root of the tree) is a vertex in* V *such that there is a unique path from* x_0 *to* x_i*,* $\forall x_i \in V$.

Definition 4.1 – Imperfect Information Extensive-Form Game [24] *An* imperfect information extensive-form game Γ *is a tuple (N, A, V, L, H,* χ*,* ρ*,* σ*, U), where:*

- N *is the set of players;*
- A *is the set of actions, and* $A_h \subseteq A$ *is the set of available actions at information set* h[10]*;*
- V *is the set of nonterminal nodes, and* $V_i \subseteq V$ *is the set of decision nodes belonging to player* $i \in N$*. These are also often considered as the states of the game;*
- L *is the set of terminal nodes, also known as leaves.* $L \cap V = \emptyset$*;*
- $H = (H_1, ..., H_n)$ *is the collection of information sets*[10]*; for each* $i \in N$, H_i *is an information partition of* V_i *such that decision nodes within the same information set* $h \in H_i$ *are not distinguishable by player i;*
- $\chi : V \to 2^A$ *is the action function, which assigns to each nonterminal node a set of possible actions;*
- $\rho : V \to N$ *is the player function, which assigns to each nonterminal node the player* $i \in N$ *who chooses an action at that node;*
- $\sigma : V \times A \to V \cup L$ *is the successor function, which maps a nonterminal node and an action to a new nonterminal node or to a terminal one;*
- $U = (u_1, ..., u_n)$*, is a collection of utility functions where* $u_i : L \to \mathbb{R}$ *is a real-valued utility function for player* $i \in N$ *on the terminal nodes L.*

Definition 4.2 – Perfect Information Extensive-Form Game [25] *A* perfect information extensive-form game *is an imperfect information extensive-form game in which all information sets consist of a single vertex.*

When a game is finite but large[11] (e.g. poker) or it is infinite (being the available actions in a continuous space), it is not possible to build an explicit representation of it. The only available option to obtain exact information on the game is to collect game samples in the form of game *traces* and corresponding payoffs for the players. In this setting, payoffs are available as the output of an *oracle*, which can be intended as a simulator, rather than specified analytically or through a payoff matrix, which is the classical approach [35].

Definition 5 – Simulation-based Game [35] *A* simulation-based game *is a tuple* (N, S, O)*, where N is the set of players, S is the set of strategies and O is an oracle producing a possibly noisy sample from the joint payoff function of players, given a joint strategy profile.*

Definition 6 – Empirical Game *An* empirical game *is an abstracted, that is, smaller and simpler, version of a simulation-based game constructed via finite sampling.*

A game consists of a sequence of states in which players make moves and end up in other states. A *trace* of a game represents a possible sequence of states and actions leading to a terminal state and a corresponding utility for the players. It therefore represents one of the many[12] possible plays by the players.

[10] *Information set h*: a set of decision vertices ($h \in V_i$) of player i that are indistinguishable by him given his information at that stage of the game [25].

[11] *Large game*: a game whose representation through a tree is infeasible.

[12] The number of *traces* of a finite game is equal to $|L|$.

Definition 7 – Trace *A* trace *of a game is a vector $\tau = (v_1, a_1, ..., v_n, a_n)$, where $v_i \in V$ are the traversed states, $a_i \in A$ are the undertaken actions, $i \in [1, n]$, v_n and a_n such that $l = \sigma(v_n, a_n) \in L$.*

Definition 8 – Behavioral Strategy *A* behavioral strategy *is a function $s_i : H_i \rightarrow \Delta^{|A_{H_i}|}$, where $i \in N$, that associates to each information set $h \in H_i$ a probability distribution over the available actions A_h.*

Solving a game consists in finding an equilibrium, which in its most classical form is a Nash Equilibrium (NE). A *strategy profile*[13] is a NE if, for each player of the game, a player does not benefit from deviating from their strategy, keeping the strategies of all the other players fixed.

Definition 9 – Nash Equilibrium [25] *Given a game, a strategy profile $\bar{s}$ is a* Nash Equilibrium *if and only if, $\forall i \in N$, $\forall s_i \in S_i$ it holds:*

$$U_i(\bar{s}) \geq U_i(s_i, \bar{s}_{-i})$$

where $U_i(s)$ is the expected utility[14] *of player i adopting strategy s, S_i is the set of strategies of player i, $\bar{s}_{-i}$ is a strategy profile containing the strategies of all players except that of player i, (s_i, $\bar{s}_{-i}$) is the strategy profile obtained by the combination of s_i and $\bar{s}_{-i}$.*

1.1.2 Abstractions

A tree is an effective way to represent a sequential game. However, the number of nodes of a tree is exponential in its depth and highly depends on the branching factor. In sequential games nodes represent the *states* of the game; the depth[15] of a terminal node represents the number of *moves* to reach that node, that is, an outcome of the game; the branching factor is the average number of *actions* available to the players at each node. The complexity of decision making is positively correlated to these factors and this is why when analyzing large games, in order to lower game complexity while trying to retain all relevant information, abstractions are used.

Definition 10 – Abstraction [6] *An* abstraction *of a game is a smaller version of the game with the purpose of capturing the most essential properties of the real domain, such that the solution of the abstracted game provides a useful approximation of an optimal strategy for the underlying real game.*

Definition 11 – Coarseness *The* coarseness *of an abstraction is a measure of how approximate the abstraction is. The more information is kept, the less the abstraction is coarse and the more it is fine-grained.*

Abstractions are preliminarily divided in:

- *Lossless information abstractions*: abstractions not loosing any information about the game [17].
- *Lossy information abstractions*: a more abstracted version of lossless information abstractions resulting in a loss of information about the game [16].

Abstractions can be divided in three categories: information abstractions, action abstractions and simulation-based abstractions.

Information Abstractions

Information abstraction is an abstraction method such that the agents cannot distinguish some of the states that they can distinguish in the actual game [28]. These are also referred to as *state abstractions*.

[13] *Strategy profile*: a vector of strategies, one for each player.
[14] *Expected utility*: sum of utilities for each leaf weighted over the probability of reaching that leaf.
[15] *Depth of a node*: the length of the path from the root of a tree to that node.

Definition 12 – State Abstraction [1] *A* state abstraction *is an abstraction where the set of states is restricted by grouping together similar*[16] *states. More formally, a* state abstraction $\Phi : V \to V_\Phi$ *maps each state* $v \in V$ *to an abstract state* $v_\Phi \in V_\Phi$, *where typically* $|V_\Phi| \ll |V|$.

More specific implementations of information abstractions include:

- *Expectation-based abstractions*: an abstraction method using states clustering to abstract states and integer programming to assign *children* to states in the abstraction tree minimizing the expected error [16].
- *Potential-aware abstractions*: abstractions where each state of the game is associated to a histogram over future possible states, capturing its *potential* - that is, a measure of how close a state is to a positive outcome of the game for a player [18, 28].
- *Strategy-based abstractions*: an iterative abstraction method where the equilibrium strategies found in an abstraction are used to guide the generation of the next abstraction [28].
- *Extensive-form game abstractions*: abstractions applied to information sets instead of states [21].

Action Abstractions

Action abstraction is an abstraction method where the number of available actions to each player is less than in the original game [28].

The methods that are used the most comprise *bucketing* and *discretization*. The first clusters actions together according to their similarity[16] in order to significantly reduce the space of available actions. The latter discretizes a continuous space of actions so as to transform an infinite game into a finite one.

Simulation-based Abstractions

Simulation-based abstraction refers to simulation-based games. The abstracted version of a simulation-based game is an empirical game. For these games a complete and accurate description, in the form of knowledge of the game's utility functions, is not available [34].

In this kind of games the abstracted version of the game is built starting from data in the form of traces. The data is fed to an *oracle* – a simulator – that outputs a possibly noisy payoff for a given strategy of the players. If the oracle is queried with all the possible traces and if it outputs the exact payoffs for the players, that is, it does not output noisy payoffs, then the whole original game is reconstructed.

Simulation-based abstraction can be intended as a bottom-up approach, as the game is built starting from traces. Instead, the aforementioned approaches of information abstraction and action abstraction start from the model and build a smaller version of the tree, resulting in a top-down approach.

Finally, the purpose of abstractions is to reduce the complexity of a game by approximating it. A measure of the approximation is hereby presented.

Definition 13 – Uniform ϵ-approximation [34] *A game* Γ' *is said to be a* uniform ϵ-approximation *of another game* Γ *when*

$$||\Gamma - \Gamma'||_\infty \leq \epsilon$$

where $||\Gamma - \Gamma'||_\infty := \sup_{i \in N,\, s \in S} |u_i(s) - u'_i(s)|$.

[16] The similarity criteria is domain-specific.

1.2. Research topic

The main problem faced by Game Theory is that of game representation and resolution. Solving a game typically means finding its Nash equilibria[17]. Finding an exact Nash equilibria is not always feasible. This is why approximated solutions of the game, corresponding to quasi-optimal strategies, are considered good. These go under the name of ϵ-Nash equilibria.

Algorithmic Game Theory is focused on designing algorithms to solve games, comprising finding Nash equilibria. In the early days of Algorithmic Game Theory, relatively small games were analyzed and their reduced size allowed them to be solved through the use of linear programming [6]. Recently however, with the introduction of the concept of regret minimization in imperfect information games [36], the most used ways to solve games for Nash equilibria are based on Counterfactual Regret Minimization (*CFR*) [36], which can solve larger games. Not surprisingly there exist many variations of it with improved performances [7, 14, 23, 31]. In general, several equilibrium-finding algorithms able to solve a game were designed, however a central challenge in solving games is that the game might be too large. For instance, two-player no-limit Texas hold'em poker has more than 10^{165} nodes [20].

In order to solve the issue of complexity in decision making with large games, the concept of abstractions was developed [6]. Abstractions are a method to lower the game complexity while retaining all relevant information. The method generally consists in building a smaller version of the game tree with a reduced number of states and actions.

Substantial effort was put into abstractions. In 2010 Sandholm [28] described the state of the art of abstractions, and this paper, specifically in Section 2, aims to do so at this time – 2019. The most notable applications of abstraction techniques were developed by Brown and Sandholm with *Libratus* [10] and *Pluribus* [11]. Despite using both information and action abstractions, they claim abstractions are not enough to solve a large extensive-form game. To cope with the limitations that abstractions carry on the quality of the solution, *refinement techniques* were implemented [4, 9, 10, 11]. These techniques aim to improve the quality of the abstraction as the game goes on, by solving *nested subgames*, dropping unnecessary information and adding actions with the ultimate goal of refining the abstracted version of the game, that is, making it less coarse.

In practice, sequential games are very large and their complexity prevents them to be fully represented, explored and analyzed to find equilibria. Being able to study large and infinite games through abstractions is crucial to extend the applicability of game theoretical principles to real-world problems. These include every possible strategic situation that is representable through a sequential game, including but not limited to recreational games, sports, governance and conflicts. This is why our research topic is of great significance.

Specifically, in the field of Security, adversaries can be distinguished between *attackers* and *defenders*. *Security problems* are two-player games where the defender (the security force) commits to a security policy, and the attacker (terrorist, smuggler, etc.) conducts surveillance to learn the policy. The attacker then either takes the best action or may be sufficiently deterred and dissuaded from attacking the protected target [30].

Our research topic is mainly positioned in the field of Algorithmic Game Theory, in particular it aims to further develop and analyze the issues of the aforementioned topics of research.

2. Main related works

2.1. Classification of the main related works

The research and works carried out on abstractions may be classified according to the following criteria:

- **Representation**: explicit VS implicit.
- **Abstraction method**: information VS action VS simulation-based.

[17] According to Nash's Existence Theorem, every game with a finite number of players in which each player can choose from finitely many pure strategies has at least one Nash equilibrium.

- **Abstracted game generation**: offline VS online.
- **Implementation**: general VS domain-specific (marked with †).
- **Date of publication**: past VS recent (in bold).

Representation-wise, *explicit* means that generally the game can be explicitly built through a top-down approach; *implicit* means that the game can only be theoretically built as there are infinitely many actions and states, as these belong to a continuous space that is not to be discretized. The concept of *online abstracted game generation* comprises all the techniques that are able to build abstractions in an *online fashion*[18], possibly applying refinement techniques. A work is classified as *recent* if it was published three or less years ago from now – 2019: the starting year of our research. The works classified under *Other* do not have a specific collocation according to the aforementioned criteria, however they are still related to the research topic.

	Explicit		Implicit
	Information	Action	Simulation-based
Offline	**[2]**§, **[3]**§, [6]†, [15]†, [16], [17], [18]†, [21], [22], [29]†	**[3]**§, **[5]**, [8], [19]†	**[32]**
Online	**[4]**, **[10]**†, **[11]**†	[9], **[10]**†, **[11]**†, **[13]**†	**[34]**
Other	**[7]**†, **[33]**		

Table 1: Classification of the main related works.

2.2. Brief description of the main related works

After classifying the main related works in Section 2.1, the most relevant ones to the research topic are hereby analyzed more in detail, presenting their contributions and highlighting their limitations.

Information Abstractions

The majority of research carried out on abstractions is about information abstraction. Early works were initiated by Shi and Littman in [29], and by Billing et al. in [6]. These use *linear programming* and *bucketing* to abstract states in 2-player poker. However, these are not particularly interesting for our research as the methods they used are now considered basic and considerable improvements have been made over time.

The most relevant works were presented by Gilpin and Sandholm. Before their works were published, abstractions were computed by hand. *Automated abstractions* were introduced by Gilpin and Sandholm in [15]. Their work introduces a Texas hold'em poker player (*GS1*) that is able to solve a large linear program offline so as to compute optimal strategies for the abstracted first part of the game. When playing, it exploits the computed best strategies and observes the state of the game after the initial moves. Then, it updates the probabilities of reaching final outcomes given the information acquired and adapts its strategy accordingly. Although being an innovative implementation for the time, *GS1*'s performance against human players was far from declaring it to be an overall winner.

[18] *Online*: in the context of abstractions, *online* refers to starting from a very coarse abstraction and progressively adding information to it so as to refine it.

§This work belongs to the field of Reinforcement Learning.

†Domain-specific implementation: poker.

In 2007, Gilpin and Sandholm introduced the concept of *ordered game isomorphic abstraction transformation*, which allows to convert any Nash equilibrium of an abstracted game into one in the original game. They achieved this through *GameShrink*. However, when addressing large games, this method does not preserve equilibrium, however it still yields close-to-optimal strategies [17].

Interesting advances were obtained in the same years, still by Gilpin and Sandholm, through the introduction of *expectation-based abstractions* [16] and *potential-aware abstractions* [18].

The former work uses *k-means clustering* to obtain a state abstraction and it aims to minimize the expected error when assigning children to these states through linear programming. It then simulates the outcome of the game in order to keep the complexity relatively low. The authors themselves inspire further research through the idea of abstracting in an iterative manner where an abstraction is *refined* based on the statistical model of the player in self-play [16].

In the latter, the main idea is that of capturing the *potential* of a state. The potential is to be intended as the likelihood of ending in a positive outcome leaf starting from that state and considering the size of the payoff in the measure. The limits of this work lie in the fact that given two states with the same potential it is not possible to evaluate which of the two will have its potential significantly vary sooner. Consequently it is not possible to distinguish the two based on the cost of obtaining relevant information about them, with lower cost being better.

In [21] Kroer and Sandholm introduce *extensive-form game abstractions*: abstractions on *information sets* rather than on states. They give theoretical guarantees on solution quality in abstractions in extensive-form games. They also contributed with the introduction of an *equilibrium refinement* that can be used to analyze the quality of general Nash equilibria from abstract games. Despite working for any game with perfect recall, the set of abstractions they can compute is only a subset of all possible abstractions.

Action Abstractions

Action abstractions were analyzed in [19] by Hawkin and Holte in 2011. They focused their research on abstractions by studying the choice of the value of parameters of an action.

The first substantial contribution to the field of action abstraction was done by Brown and Sandholm in 2014 with [8]. They provide the first action abstraction algorithm with convergence guarantees for extensive-form games. In particular, the presented algorithm is able to select a small number of discrete actions to use from a continuum of actions, transforming an infinite game into a finite one, considerably reducing the size of the game.

Basak and Kiekintveld in [5] introduce the idea of abstracting games by clustering strategies and then solving them by finding and solving suitable subgames. However, they state that there are several abstraction approaches, mainly related to the abstraction method and to the way of solving the abstracted game: understanding which method is the best or the most appropriate is still an open problem.

An interesting contribution was brought by Abel et al. in [3]. In particular, they combined state and action abstraction and introduced a *value loss* that is extended to capture *near-optimality* of joint state-action abstraction.

Abstraction Refinement

One of the first online methods for abstractions was developed by Brown and Sandholm in 2015 [9]. Their method consists in generating coarse abstractions and later adding actions making them finer-grained. This result is to be considered quite innovative as it allowed an agent to begin learning with a coarse abstraction and then strategically insert actions without having to restart the equilibrium finding. According to the authors, this method converges to a better solution than equilibrium finding in fine-grained abstractions. Moreover, the algorithm is game independent and it is considered to be useful in solving games with large action spaces.

The online approach has been adopted also in state abstraction by Avni et al. in [4]. They present an *abstraction-refinement* method that is able to *refine* the abstraction function when approximations are too coarse to find a Nash equilibrium.

Beyond Abstractions

The most advanced techniques do not rely on abstractions only. These were developed by Brown and Sandholm first in [10] and then in [11], contributing respectively with *Libratus* and *Pluribus*, the latter named "superhuman AI for multiplayer poker".

Libratus features three main modules. The first computes an abstraction of the game and solves it through self-play via an improved version of Monte Carlo Counterfactual Regret Minimization (*MCCFR*) [14, 23], obtaining what the authors call a *blueprint strategy*. The second module comes into play later in the game as a *refinement* by constructing a finer-grained abstraction for a particular part of the game that is reached during play and solves it in real time. They exploit the *nested subgame solving* technique on *off-tree actions*[19], that is, solving subgames with the off-tree actions included. This technique comes with a provable safety guarantee [12]. Finally, *Libratus* is able to self-improve by enhancing the *blueprint strategy*. It does so by filling in missing branches in the *blueprint abstraction* and solving those for a strategy. The implementation of *Libratus* is limited to two-player heads-up no-limit poker, however the authors claim that their game-theoretic approach is application-independent and that it will be important for the future growth and widespread application of AI.

Pluribus is an enhanced version of its predecessor *Libratus* that is able to play six-player heads-up no-limit poker. Despite proving itself to be an undisputed winner against top players, results are not solidly supported by theory. In fact, finding an exact or approximated Nash equilibrium in zero-sum games with more than two players is computationally hard [27]. Moreover, even if a Nash equilibrium could be computed efficiently in a game with more than two players, it is not clear if playing such an equilibrium strategy would be wise[20]. Finally, the goal of the authors was not to obtain a specific game-theoretic solution concept, but consisted in creating an AI able to empirically defeat human opponents.

Simulation-based Abstractions

Little research has been carried out in the field of simulation-based abstractions.

A theoretical contribution was given by Tuyls et al. with [32] a year ago. In this work they derive guarantees on the quality of all equilibria learned from finite samples providing theoretical bounds for empirical game-theoretical analysis of complex multi-agent interactions. They show that a Nash equilibrium of the empirical game is an approximate Nash equilibrium of the true underlying game and they provide insights on the number of data samples required to obtain a close enough approximation.

This year Viqueira et al. presented [34]. In this work the authors study simulation-based games, that in their abstracted form are called empirical games. They start from game traces and approximate game utilities from them, generating an abstracted version of the game. They are able to learn all equilibria of a game through two algorithms, one of which is a pruning algorithm refining the empirical game at each iteration, until the equilibria are approximated to the desired accuracy. There are however some limitations to this work, since, according to the authors, their algorithm can only find pure strategy[21] ϵ-Nash equilibria. They say that computing mixed strategy Nash equilibria is intractable, being PPAD[22] complete.

2.3. Discussion

An analysis on the available literature on abstractions in extensive-form games was carried out and presented in this document. To sum up our study on the state of the art for our research topic, we present a critical examination of the main focus points of research in the past years. We investigate which problems are stills open and which are the areas where further work is needed.

By observing Table 1, it is evident that the majority of research in the field was focused on *information abstraction* in an offline fashion. In the last few years more research was put into *action abstraction*. However, interestingly, in the past four years, most of the research was focused on refinement techniques. The most notable work was produced by Brown and Sandholm giving birth to Libratus [10], which unsurprisingly won the Marvin Minsky

[19] *Off-tree actions*: actions that are outside the precomputed abstraction.

[20] See the Lemonade Stand Game for an example [37].

[21] Strategies can be *pure* or *mixed*. Actions of a mixed strategy are taken according to a probability distribution; in a pure strategy only one action is taken and all others never are.

[22] *PPAD*: Polynomial Parity Arguments on Directed graphs – a complexity class.

Medal. In their work they combine abstractions with *MCCFR*, *nested subgame solving* and *self-improvement*. Their work is supported by strong theoretical evidence and their results are outstanding. However, it must be noted that most of their research is focused on poker, even if they state that their game-theoretic approach is application independent and that they use poker as an implementation since *"no other popular recreational game captures the challenges of hidden information as effectively and as elegantly as poker"* [11].

Little research has been carried out on simulation-based games and related abstractions. To the best of our knowledge, only Tuyls et al. in [32] and Viqueira et al. in [34] were able to achieve substantial results. However, there are some limitations to the research of Viqueira et. al, since their algorithm can only find pure strategy ϵ-Nash equilibria. Simulation-based games are of great interest since they are the only way a game with infinite actions and states can be represented. In fact, collecting game traces represents the only way to obtain exact and eventually complete information on the game.

Being able to find mixed strategy Nash equilibria in large or infinite games would allow great breakthroughs in real-world scenarios. There are many areas where game-theoretic principles are already applied so as to find optimal strategies for the involved agents. Just to cite a few: theoretical economics, networks and flows, political science, military applications, evolutionary biology. However, large games are becoming of great interest as most real-world meaningful applications usually correspond to infinite games, being the available actions in a continuous space.

Security is recognized as a world-wide challenge and game theory is an increasingly important paradigm for reasoning about complex security resource allocation, being security resources usually very limited. Tambe et al. in [30] present some of the successful applications they were able to design and deploy through game-theoretic approaches. Among the physical ones, they were able to protect ports, airports, transportation, wildlife including endangered fish and forest from poachers and smugglers, and lower public transportation fare evasion. The challenge we face regarding physical security is that existing algorithms still cannot scale up to very large scale domains such as scheduling randomized checkpoints in cities.

Being able to solve large games would allow the application of game-theoretic principles, that is, finding optimal strategies, to any real-world meaningful strategic situation. For instance, other critical infrastructures can be protected, illegal drug, money and weapons trafficking could be drastically limited, and urban crime could be suppressed.

Furthermore, network security is an important problem faced by organizations who operate enterprise networks housing sensitive information and perform important functions. In recent years there have been several successful cyber attacks on enterprise networks by malicious actors. A network administrator should respond to requests from an adversary attempting to infiltrate their network. These adversary agents must be investigated by cyber analysts to determine whether or not they are an attack and usually the attacks outnumber them. Cybersecurity problems are more complex than physical ones, as the space of actions can be much larger, leading to infinite games.

Finally, we believe that further research must be carried out in the field of simulation-based abstractions, aiming to provide theoretical guarantees. Moreover, it would be game-changing to find a domain-independent method to obtain approximated, or even better exact, optimal strategies starting from game traces and corresponding possibly noisy payoffs and solving an abstracted version of the game.

References

[1] Abel, D. A theory of state abstraction for reinforcement learning. In *Proceedings of the Doctoral Consortium of the AAAI Conference on Artificial Intelligence* (2019).

[2] Abel, D., Arumugam, D., Asadi, K., Jinnai, Y., Littman, M. L., and Wong, L. L. State abstraction as compression in apprenticeship learning. In *Proceedings of the AAAI Conference on Artificial Intelligence. AAAI Press* (2019).

[3] Abel, D., Umbanhowar, N., Khetarpal, K., Arumugam, D., Precup, D., and Littman, M. L. Value preserving state-action abstractions. In *ICLR* (2019).

[4] Avni, G., Guha, S., and Kupferman, O. An abstraction-refinement methodology for reasoning about network games. *Games 9*, 3 (2018), 39.

[5] Basak, A. Abstraction using analysis of subgames. In *Thirtieth AAAI Conference on Artificial Intelligence* (2016).

[6] Billings, D., Burch, N., Davidson, A., Holte, R., Schaeffer, J., Schauenberg, T., and Szafron, D. Approximating game-theoretic optimal strategies for full-scale poker. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence* (San Francisco, CA, USA, 2003), IJCAI'03, Morgan Kaufmann Publishers Inc., pp. 661–668.

[7] Brown, N., Lerer, A., Gross, S., and Sandholm, T. Deep counterfactual regret minimization. *arXiv preprint arXiv:1811.00164* (2018).

[8] Brown, N., and Sandholm, T. Regret transfer and parameter optimization. In *Twenty-Eighth AAAI Conference on Artificial Intelligence* (2014).

[9] Brown, N., and Sandholm, T. Simultaneous abstraction and equilibrium finding in games. In *Twenty-Fourth International Joint Conference on Artificial Intelligence* (2015).

[10] Brown, N., and Sandholm, T. Superhuman ai for heads-up no-limit poker: Libratus beats top professionals. *Science 359*, 6374 (2018), 418–424.

[11] Brown, N., and Sandholm, T. Superhuman ai for multiplayer poker. *Science 365*, 6456 (2019), 885–890.

[12] Burch, N., Johanson, M., and Bowling, M. Solving imperfect information games using decomposition. In *Twenty-Eighth AAAI Conference on Artificial Intelligence* (2014).

[13] Garivier, A., Kaufmann, E., and Koolen, W. M. Maximin action identification: A new bandit framework for games. In *Conference on Learning Theory* (2016), pp. 1028–1050.

[14] Gibson, R., Lanctot, M., Burch, N., Szafron, D., and Bowling, M. Generalized sampling and variance in counterfactual regret minimization. In *Twenty-Sixth AAAI Conference on Artificial Intelligence* (2012).

[15] Gilpin, A., and Sandholm, T. A competitive texas hold'em poker player via automated abstraction and real-time equilibrium computation. In *Proceedings of the National Conference on Artificial Intelligence* (2006), vol. 21, Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, p. 1007.

[16] Gilpin, A., and Sandholm, T. Better automated abstraction techniques for imperfect information games, with application to texas hold'em poker. In *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems* (2007), ACM, p. 192.

[17] Gilpin, A., and Sandholm, T. Lossless abstraction of imperfect information games. *Journal of the ACM (JACM) 54*, 5 (2007), 25.

[18] Gilpin, A., Sandholm, T., and Sørensen, T. B. Potential-aware automated abstraction of sequential games, and holistic equilibrium analysis of texas hold'em poker. In *Proceedings of the 22Nd National Conference on Artificial Intelligence - Volume 1* (2007), AAAI'07, AAAI Press, pp. 50–57.

[19] Hawkin, J. A., Holte, R., and Szafron, D. Automated action abstraction of imperfect information extensive-form games. In *Twenty-Fifth AAAI Conference on Artificial Intelligence* (2011).

[20] Johanson, M. Measuring the size of large no-limit poker games. *arXiv preprint arXiv:1302.7008* (2013).

[21] Kroer, C., and Sandholm, T. Extensive-form game abstraction with bounds. In *Proceedings of the Fifteenth ACM Conference on Economics and Computation* (New York, NY, USA, 2014), EC '14, ACM, pp. 621–638.

[22] Kroer, C., and Sandholm, T. Extensive-form game imperfect-recall abstractions with bounds. *arXiv preprint: http://arxiv. org/abs/1409.3302* (2014).

[23] Lanctot, M., Waugh, K., Zinkevich, M., and Bowling, M. Monte carlo sampling for regret minimization in extensive games. In *Advances in neural information processing systems* (2009), pp. 1078–1086.

[24] Leyton-Brown, K., and Shoham, Y. Multiagent systems: Algorithmic, game-theoretic, and logical foundations, 2009.

[25] Maschler, M., Solan, E., and Zamir, S. Game theory. *Cambridge University Press*, 2013.

[26] Myerson, R. B. Game theory: Analysis of conflict. *Harvard University Press*, 1991.

[27] Rubinstein, A. Inapproximability of nash equilibrium. *SIAM Journal on Computing 47*, 3 (2018), 917–959.

[28] Sandholm, T. The state of solving large incomplete-information games, and application to poker. *AI Magazine 31*, 4 (Sep. 2010), 13–32.

[29] Shi, J., and Littman, M. L. Abstraction methods for game theoretic poker. In *International Conference on Computers and Games* (2000), Springer, pp. 333–345.

[30] Tambe, M., Jiang, A. X., An, B., Jain, M., et al. Computational game theory for security: Progress and challenges. In *AAAI spring symposium on applied computational game theory* (2014).

[31] Tammelin, O. Solving large imperfect information games using cfr+. *arXiv preprint arXiv:1407.5042* (2014).

[32] Tuyls, K., Perolat, J., Lanctot, M., Leibo, J. Z., and Graepel, T. A generalised method for empirical game theoretic analysis, 2018.

[33] Viqueira, E. A., Cousins, C., Mohammad, Y., and Greenwald, A. Empirical mechanism design: Designing mechanisms from data. In *UAI* (2019).

[34] Viqueira, E. A., Cousins, C., Upfal, E., and Greenwald, A. Learning equilibria of simulation-based games, 2019.

[35] Vorobeychik, Y., and Wellman, M. P. Stochastic search methods for nash equilibrium approximation in simulation-based games. In *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems-Volume 2* (2008), International Foundation for Autonomous Agents and Multiagent Systems, pp. 1055–1062.

[36] Zinkevich, M., Johanson, M., Bowling, M., and Piccione, C. Regret minimization in games with incomplete information. In *Advances in neural information processing systems* (2008), pp. 1729–1736.

[37] Zinkevich, M. A., Bowling, M., and Wunder, M. The lemonade stand game competition: solving unsolvable games. *ACM SIGecom Exchanges 10*, 1 (2011), 35–38.