

Introducing the Hearthstone-AI Competition

Alexander Dockhorn and Sanaz Mostaghim
Institute for Intelligent Cooperating Systems

Department for Computer Science, Otto von Guericke University Magdeburg

Email: {alexander.dockhorn, sanaz.mostaghim}@ovgu.de

Abstract—The Hearthstone AI framework and competition motivates the development of artificial intelligence agents that can play collectible card games. A special feature of those games is the high variety of cards, which can be chosen by the players to create their own decks. In contrast to simpler card games, the value of many cards is determined by their possible synergies. The vast amount of possible decks, the randomness of the game, as well as the restricted information during the player’s turn offer quite a hard challenge for the development of game-playing agents. This short paper introduces the competition framework and goes into more detail on the problems and challenges that need to be faced during the development process.

Index Terms—Hearthstone, Artificial Intelligence, Competition, Challenges

I. INTRODUCTION

THE development of artificial intelligence (AI) was often guided by the plethora of available real world applications. The recent success of AI agents such as AlphaGo, brought game AI back into the center of attention for media and researchers alike. Games pose challenging and often well-balanced problems demanding the development of new agent architectures and allowing their evaluation based on their success in playing the game. Due to their competitive nature, games offer an ideal test-bed for the comparison of multiple agents under differing conditions.

Game based competitions and benchmarks motivated researchers to create specialized agents in games of many different genres, such as Chess [1], Go [2], Poker [3], Pac-Man [4], Starcraft [5] and many more. While these focused on the development of an agent for a single game, other competitions and frameworks tried to generalize the solutions to a broader scope. The Arcade Learning Environment framework (ALE) [6] as well as the General Video Game AI framework (GVGAI) [7] test an agent’s success on a wide range of games. All of these benchmarks pose unique demands on the agent’s planning and reasoning capabilities, while their accessibility ensures a simple generation and evaluation of new agents.

In this paper we introduce the Hearthstone AI competition, because we believe that it is an excellent addition to the set of currently available benchmarks. The focus of this competition is the development of autonomously playing agents in the context of the online collectible card game Hearthstone.

Interesting features of collectible card games include, but are not limited to:

- **Partial observable state space:** Critical information is hidden from the player. The agent typically does not know which cards it will draw and is unaware of the opponent’s deck and hand cards. This is especially relevant when estimating the risk of an action, since this often depends on the current options of our opponent.
- **High complexity:** Hearthstone currently features more than 2000 different cards. This high amount and the number of their unique effects drastically increases the game-tree complexity.
- **Randomness:** In contrast to similar deck-building card-games, card effects in Hearthstone often involve randomness. This makes it particularly difficult to plan ahead, such that the agent needs to continuously adapt its strategy according to the observed result.
- **Deck-building:** A deck in Hearthstone consists of 30 cards, which do not necessarily need to be unique. Card synergies mean that certain combinations of cards often have a stronger effect than playing the respective cards separately. Exploiting these synergies is a very complex task but allows to play cards to their full potential.
- **Dynamic Meta-Game:** The odds of winning the next game depend not only on the player’s skill or the current deck, but also on the rate of other decks being played. The analysis of this meta-game can be crucial for creating or choosing the next deck to be played. Finally, a deep understanding of the meta-game can be used to predict future enemy moves based on previously seen cards and select appropriate actions based on them.

In the following we will give a short introduction to Hearthstone (Section II) and present the competition framework (Section III). Competition tracks of the Hearthstone AI’19 competition are highlighted in after which an outline of future competition tracks is presented (Section IV).

More information on the competition, results of the 2018’s installment and additional resources for research on Hearthstone can be found at:

<http://www.ci.ovgu.de/Research/HearthstoneAI.html>

We invite all developers to participate in this exciting research topic and hope to receive many interesting submissions.



Fig. 1: Elements of the Hearthstone game board: (1) weapon slot (2) hero (bottom: player, top: opponent) (3) opponent's minions, (4) player's minions, (5) hero power, (6) hand cards, (7) mana, (8) decks, (9) history

II. HEARTHSTONE: HEROES OF WARCRAFT

Hearthstone is a turn-based digital collectible card game developed and published by Blizzard Entertainment [8]. Players compete in one versus one duels using self-constructed decks belonging to one hero out of nine available hero-classes. In those matches players try to beat their opponents by reducing their starting health from 30 to 0. This can be achieved by playing cards from the hand onto the game board at the cost of *mana*. Played cards can be used to inflict damage to the opponent's hero or to destroy cards on his side of the game board. The amount of mana available to the player increases every turn (up to a maximum of 10). More mana gives access to increasingly powerful cards and increases the complexity of turn while the game progresses. At the beginning of each turn the player draws a new card until his deck is empty, in which case he receives a step-wise increasing amount of *fatigue*-damage. The standard game board is shown in Figure 1.

Players need to construct their own decks to play the game. Those consist of 30 cards, which can be chosen out of more than 2000 currently available cards. Cards and game mechanics are added in regular updates. Each card bears unique effects, which the players can use to their advantage. Additionally, each player chooses a hero, which gives access to a class specific pool of cards and hero power.

Cards can be of the type minion, spell, or weapon. Figure 2 shows one example of each card type. Minion cards assist and fight on behalf of the hero. They usually have an attack, health, and mana cost-value, as well as a short ability text. Furthermore, minions can belong to a special minion type, which is the basis for many synergy effects. Once played, they can attack the enemies side of the board in every consecutive turn to inflict damage on either the opponent's minions or hero. Attacking a target also reduces the attacker's health by the target's attack value. In case any minion's health drops to zero, it is removed from the board and put into its player's

graveyard. Spell cards can be cast at the cost of mana to activate various abilities and are discarded after use. They can have a wide range of effects, e.g. raising a minion's attack or inflicting damage to a random minion. Secrets, which are a special kind of spells, can be played without immediately activating their effect. After a trigger condition was fulfilled, the secret will be activated. Once activated, the secret is removed from the board. Weapon cards are directly equipped to the player's hero and enable him to attack. Their durability value limits the number of attacks till the weapon breaks. Only one weapon can be equipped at the same time.

Hearthstone decks are often created around a common theme. Multiple cards that positively influence each other can create strong synergies and increase the value of each card in context of its deck. For this reason, the value of a single card highly depends on the player's hand, current elements on the board, and the deck in general. Common examples are minion cards of the same type, e.g. "Murloc", which give each other additional advantages, e.g. an attack boost. Each of these minions is comparatively weak, but their value increases when they are played together.

Generated decks can be categorized into three major categories: aggro, mid-range, and control. Aggro decks build on purely offensive strategies, which often include a lot of minions. Control decks try to win in the long run by preventing the opponent's strategy and dominating the game situation. The playing style of mid-range decks is between aggro and control. They try to counter early attacks to dominate the game board with high-cost minions in the middle of the game.

Game length and branching factor can be dependent on the player's decks in the current game. Some decks try to play single high-cost cards, whereas others build on versatile combinations. The complexity of each turn and the uncertainty faced during the game makes Hearthstone a challenging problem for AI research.



Fig. 2: General card types: cards include (1) mana cost, (2) attack damage, (3) health/durability, (4) and special effects.

III. HEARTHSTONE-AI COMPETITION FRAMEWORK

The Hearthstone-AI competition is based on the community driven simulator Sabberstone. This framework is written in C# and the competition extends the original framework by multiple helper classes which provide an agent simple means of accessing the current game state limited to variables that would have been observable to a human player.

More specifically, each agent needs to inherit from the *AbstractAgent* class. The included functions *InitializeAgent* and *FinalizeAgent* can be used to load and store information at the beginning and end of each session. Additionally, *InitializeGame* and *FinalizeGame* are called at the beginning and end of each simulated game. These functions can be used to setup a strategy or updating it based on the games' outcome.

During a game each time the agent needs to choose an action its *GetMove* function is called. The agent is given a *POGame* object representing the partial observation of the current game-state. It contains information about the visible part of the game board, a set of remaining cards in its deck, its hand cards, and the number of cards in their opponents' hand. Furthermore, the opponent's deck as well as its hand cards are replaced by dummy-cards to ensure that this information remains hidden to the agent. The agent is ensured 60s of computation time to step-wise return a set of actions and concluding its turn. In case the turn was not ended by the agent, the returned action will be processed irreversibly and an updated game-state will be returned to the agent while asking for its next action. Actions of both players are applied until a winner can be determined or a maximum number of turns (default = 50) is exceeded. In the latter the game ends with a draw.

The *POGameHandler* class controls the simulation of multiple games and reports the result of these simulations in terms of a *GameStats* object. The number of wins, draws, and loses as well as the total and average response times per agent are tracked and reported at the end of a simulation session.

IV. COMPETITION TRACKS

During the first years of the Hearthstone-AI competition two tracks will be open for entry. We plan to extend this list in the following years to give users some time to accommodate with the framework and the game itself. The following two tracks will be open for submission in the Hearthstone-AI'19:

- **Premade Deck Playing-track** In the Premade Deck Playing-track participants will receive a list of six decks and play out all combinations against each other. Only three of the six decks will be known to the developers before the final submission. Determining and using the characteristics of players and the opponents deck to the players advantage will help in winning the game. The long-term goal of this track will be the development of an agent that is capable of playing any deck.
- **User Created Deck Playing-track** The User Created Deck Playing-track invites all participants to create their own decks or to choose from the vast amount of decks available online. Finding combinations of decks and agents that can consistently beat others will play a key role in this competition track. Additionally, it gives the participants the chance to optimize the agents strategy to the characteristics of their chosen deck.

A round robin tournament will be used to determine the average win-rate of each agent and rank them accordingly. In case this process becomes unfeasible due to a large number of participants, we will split submissions into multiple sub-tournaments to determine the best performing agents among them and use a round robin tournament to determine the winner of the competition. Matches will be repeated multiple times to accommodate for the randomness in the card draw.

In order to support incremental improvements of the agents' performance, we plan to make all submissions publicly available on the competition website after the competition evaluation has been completed.

V. CONCLUSIONS AND FUTURE PLANS

Previous work on collectible card games has been scattered on different games and frameworks. With this competition we want to provide a unified way to develop and compare AI approaches on multiple collectible card game related tasks. While current competition tracks focus on the agents' basic game playing capabilities, we like to cover various different tasks in future installments of this competition. Specifically, we plan to implement the following future tracks:

- **Deck-building:** As soon as agents are reasonably skilled in playing a given deck we want to further explore the deck building process. This is currently considered a highly creative task in which many game characteristics need to be considered to build outstanding decks.
- **Draft mode deck-building:** A special form of deck building is the draft mode. Here, an agent is presented 3 cards at a time of which it needs to choose one to include in its deck. This process is repeated until the deck is filled. Strategic planning and estimating the value of each card are important to exploit synergies and building a competitive deck.
- **Game balancing/Card generation:** During the development process of a collectible card game much of the time is put into the generation of new game mechanics and cards, since these need to fit and complement the current card pool. A future track will aim to explore these balancing tasks in more detail.

We hope that this short introduction of our competition framework motivates further research in this very interesting topic. More information on how to participate in the Hearthstone-AI'19 competition, results of the 2018's installment and additional resources for research on Hearthstone can be found at:

<http://www.ci.ovgu.de/Research/HearthstoneAI.html>

ACKNOWLEDGEMENT

We would like to thank all who contributed to the Sabberstone framework on which this competition is based on. Special thanks goes to *darkfriend77* and *Milva* who are currently organizing the framework's ongoing development process. This competition would not have been possible without them.

REFERENCES

- [1] M. Campbell, a. J. Hoane Jr., and F.-h. Hsu, "Deep Blue," *Artificial Intelligence*, vol. 134, no. 1-2, pp. 57–83, 2002.
- [2] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis, "Mastering the game of Go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, pp. 484–489, 2016.
- [3] N. Brown and T. Sandholm, "Superhuman AI for heads-up no-limit poker: Libratus beats top professionals," *Science*, vol. 1733, no. December, p. eaao1733, dec 2017.
- [4] P. R. Williams, D. Perez-Liebana, and S. M. Lucas, "Ms. pac-man versus ghost team cig 2016 competition," in *2016 IEEE Conference on Computational Intelligence and Games (CIG)*. IEEE, 2016, pp. 1–8.
- [5] S. Ontanón, G. Synnaeve, A. Uriarte, F. Richoux, D. Churchill, and M. Preuss, "RTS AI problems and techniques," *Encyclopedia of Computer Graphics and Games*, pp. 1–12, 2015.
- [6] M. G. Bellemare, Y. Naddaf, J. Veness, and M. Bowling, "The Arcade Learning Environment: An Evaluation Platform for General Agents," *Journal of Artificial Intelligence Research*, vol. 47, pp. 253–279, 2013.
- [7] D. Perez-Liebana, S. Samothrakis, J. Togelius, T. Schaul, S. M. Lucas, A. Couetoux, J. Lee, C. U. Lim, and T. Thompson, "The 2014 General Video Game Playing Competition," *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 8, no. 3, pp. 229–243, 2016.
- [8] "Blizzard Entertainment", "Hearthstone webpage," accessed on 11.04.2019. [Online]. Available: <https://playhearthstone.com>



Alexander Dockhorn is PhD student at the Otto von Guericke University in Magdeburg, Germany. In his research, he combines computational intelligence in games and intelligent data analysis with focus on partial information games and the active learning of rules and strategies from successful plays. He is student member of the IEEE and its Computational Intelligence Society. Currently he is vice-chair of the Student Games-Based Competitions Sub-Committee.



Sanaz Mostaghim is a professor of computer science at the Otto von Guericke University of Magdeburg, Germany. Sanaz has received a PhD degree (2004) in electrical engineering from the University of Paderborn in Germany and worked as a post-doctoral fellow (2004 - 2006) at the Swiss Federal Institute of Technology (ETH) Zurich in Switzerland. Sanaz received her habilitation in applied computer science from Karlsruhe Institute of technology (KIT) in 2012, where she worked as a lecturer (2006 - 2013). Her research interests

are evolutionary multi-objective algorithms, swarm intelligence and their applications in science and industry. She is an active member in several German and international societies such as IEEE, ACM and GI. Currently she is the chair of Task Force Evolutionary Multi-objective Optimization (TFEMO) and is serving as an associate editor for IEEE Transactions on Evolutionary Computation, IEEE Transactions on Emerging Technologies and IEEE Transactions on Cybernetics.