Ruprecht-Karls-Universität Heidelberg
Heidelberg Collaboratory for Image Processing
Summer Term 2019
Seminar: Artificial Intelligence for Games
Docents: Prof. Dr. Ullrich Köthe
Prof. Dr. Carsten Rother

Seminar Report

A new Star is born - Looking into AlphaStar

Name:	Johannes Daub
Enrolment Number:	3145320
Course of Studies:	Master Applied Computer Sciene (1st term)
Email:	bx401@stud.uni-heidelberg.de
Date of Emission:	October 24, 2019

Declaration of Authorship

I, **Johannes Daub** hereby confirm that the report I am submitting on **A new Star is born - Looking into AlphaStar**, in the course **Artificial Intelligence for Games**, is solely my own work. If any text passages or images from books, papers, the Web or other sources have been copied or in any other way used, all references have been acknowledged and fully cited. I declare that no part of the report submitted has been used for any other report, thesis or course in another higher education institution, research institution or educational institution. I am aware of the University's regulations concerning plagiarism, including those regulations concerning disciplinary actions that may result from plagiarism.

Heidelberg, August 08, 2019 J. Daub

Abstract

AlphaStar is a very recent artificial intelligence that combines many novel achievements in machine learning and achieves the performance of a pro player in StarCraft II. Although the details of its neural network are not known, this report sums up what is known of AlphaStar so far. This includes knowledge about AlphaStar of the StarCraft community.

Contents

List of Figures

List of Tables

1 Introduction

The people of Google DeepMind have made great contributions to the development of neural networks and their applications. One of their recent achievements includes AlphaGo[7], an artificial intelligence (AI) that is able to beat the best human players in the game of Go. While AlphaGo still uses recordings of matches played by humans, AlphaGoZero[9] goes a step further and learns the game of Go only by self-play, outperforming the previous AI AlphaGo. After this achievement, the creators were looking for a new challenge to solve. They decided to create an AI for StarCraft, which is not only very popular for its competitive scene and players, but also offers an environment which is very challenging for AI. What makes StarCraft a Challenge? [12]

- **Game theory**: There is no single best strategy to win. More strategic knowledge will yield an advantage during play.
- **Imperfect information**: Players must scout the opponent to get to know their strategy.
- **Long term planning**: Early actions can cause a big effect much later in the game.
- **Real time**: Compared to turn-based games, there is continuous action necessary and not much time to think through choices.
- **Large action space**: Up to 26 possible actions during each game step and hundreds of units that must be controlled.

Creating a StarCraft AI is just an example of what complex problems can be solved by AI. Even though it is just a game, the theoretical knowledge that was gained by creating AlphaStar is really high and can be transferred to other tasks that use machine learning to be solved. The creators themselves say:

> We believe that this advanced model will help with many other challenges in machine learning research that involve long-term sequence modeling and large output spaces such as translation, language modeling and visual representations.[12]

2 Fundamentals

2.1 StarCraft II Learning Environment - SC2LE

The SC2LE [11] is an open-source learning environment for StarCraft based on Python. The game observation will be presented in spatial feature layers, as the game itself is also spatial oriented. Other features that are available to the player are also available, like total unit count, resources, an interpretation of the mini-map and the score. See figure 1 for an example presentation of the in-game state. In contrast to previous environments, the observation is limited to the content the in-game player can see. Being able to observe the whole game state would be cheating.

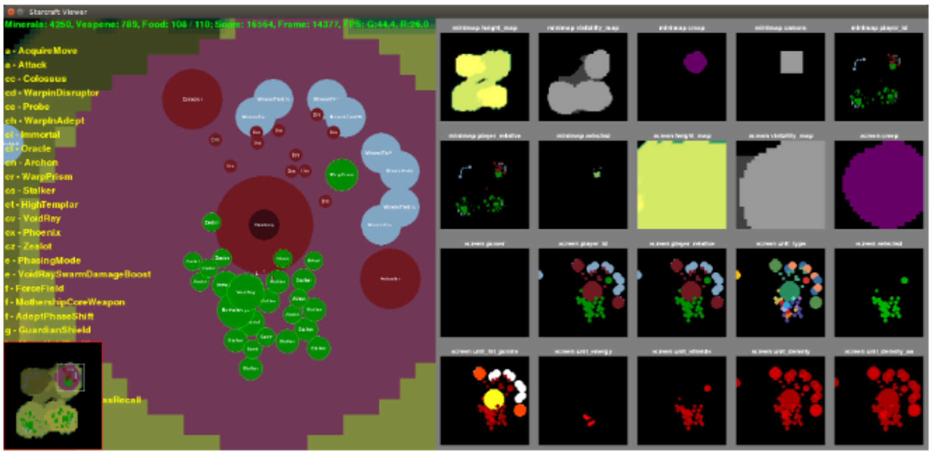

Figure 1: SC2LE Observations. The left side shows a human-understandable image and the right side shows different feature layers of the game.

The actions an agent can take are made available as functions, corresponding to actions a player would make. Together with the environment a collection of mini-games has been released, including some simple agents playing those games. The performance of the agents on the mini-games were compared to human player scores. The SC2LE is the base for AlphaStar.

2.2 Transformer Architecture

The Transformer model [10] is a novel network architecture often used in NLP. Similar to RNNs it works with sequences and can be used for translation tasks. It uses the attention mechanism, which helps the network to look at the important parts of the sequence for a given element in the sequence. A big advantage of the transformer architecture is, that it is parallelizable, saving high magnitudes of training time, compared to LSTMs for example. In StarCraft, it could be used to computes the next action regarding the history of previous actions or to decide which unit or building to create next. This would be very similar to human play as humans learn fixed build orders for the start of the game.

2.3 Nash Average for Evaluation

The Nash average [4] is a new evaluation technique that can be applied to evaluate game agents. In comparison to the Elo rating system [6], it is not biased towards invariants of agents. That means that if there are two agents which are the same it does not make a difference for the evaluation. This is valuable as it encourages adding as many agents as possible to get a more universal evaluation. The Nash equilibrium [8] is a distribution of strategies that are not exploitable, meaning there is no other distribution that is better than the Nash equilibrium. For zero-sum games, this equilibrium is unique. For the evaluation of games agents playing against each other, a meta-game is considered, where two meta-players have to choose agents, that will play against each other. The best distribution of agents chosen is the Nash equilibrium of this meta-game and is called Nash distribution.

3 Training a Star

3.1 AlphaStar League

The training of AlphaStar uses a multi-agent training system. The initial agents are trained on human replays with supervised learning. After this phase, the agents will play against each other. Each of the will go through a reinforcement learning process based on their game experience. After each iteration, the old agents will be kept and the agent with the learned improvements will get a new id. Newer agents learn improved strategies or come up with completely new strategies to counter the previous agents strategies. In addition, not all agents have the same objective. Some of them have to beat a certain opponent, while others may have to beat all others, resulting in diversified agents. To evaluate the performance, the Nash distribution will be used. When an agent has a high probability in the Nash distribution, it generally performs well against most other agents. Figures 2 and 3 show the AlphaStar league and the training progress. The whole training procedure took 14 days, where an agent could receive up to 200 years of real-time StarCraft play.

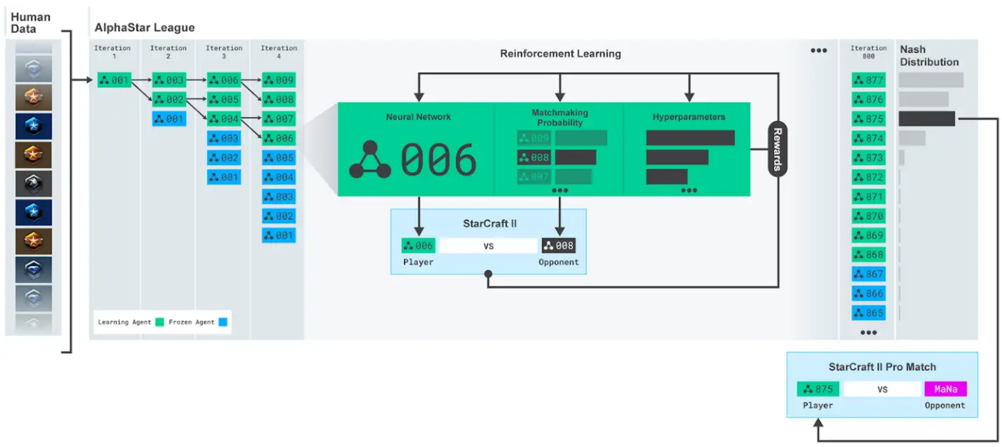

Figure 2: AlphaStar League. Left to right: Agents are first trained with human data. in the AlphaStar League the agents play against each other and after each iteration a new agent is branched from the previous one. Agents that have a high probability in the Nash distribution might be chosen to play against a human pro player.[12]

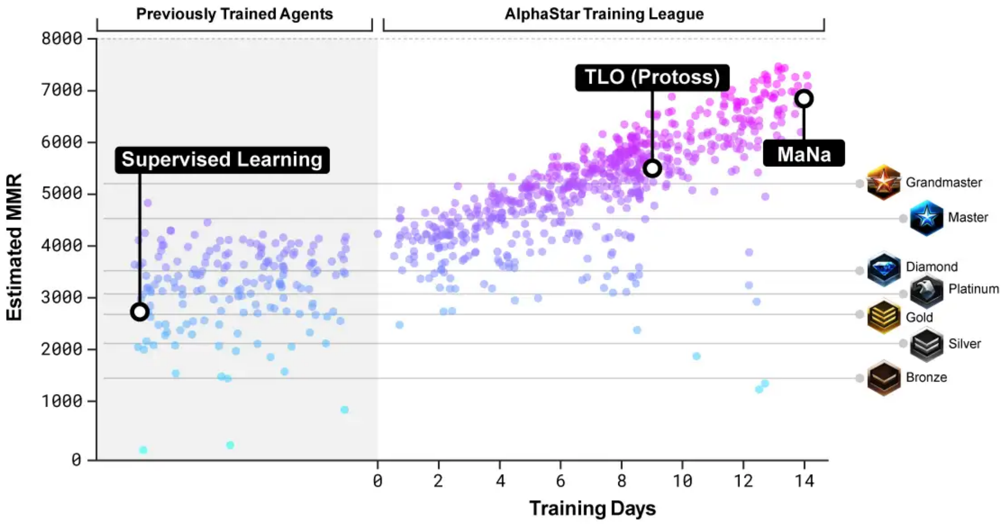

Figure 3: Estimated MMR of AlphaStar agents related to training days. After 10-12 days of training the AlphaStar agents reach an estimated MMR level as the pro player MaNa.[12]

4 Evaluation

4.1 Strategy

The strategy progression of the AlphaStar League is very interesting, as is resembles the discovery of strategies in the early time of human StarCraft play. Early strategies are mostly quick and risky rush strategies as the Cannon Rush. Later on these strategies were replaced by long-term and late-game strategies that focus on economic growth or disturb the enemy economy. Figure 4 shows the average unit counts, its changes indicate changes in the strategies played. In general, the newer agents outperform the older agents, which is indicated by their Nash distribution (fig. 5).

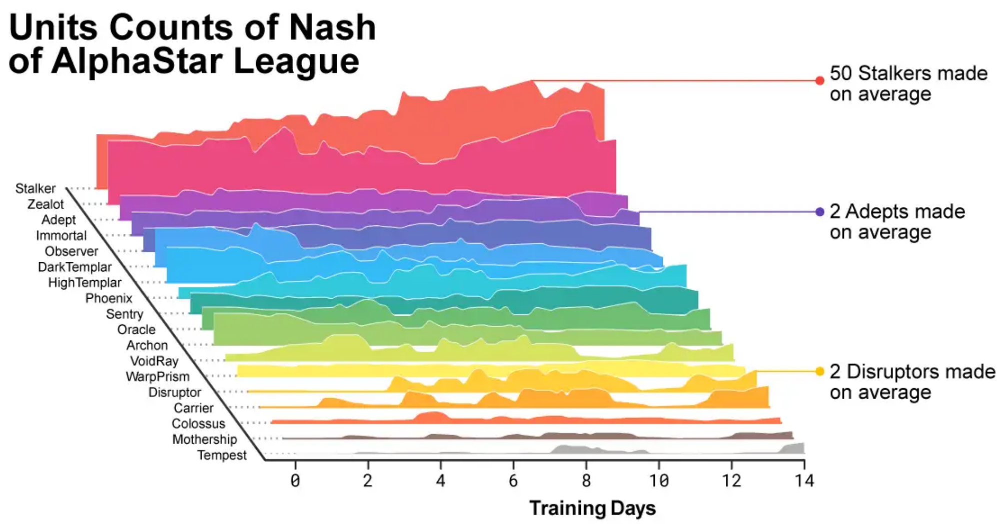

Figure 4: Average of unit counts of agents related to training days. The amount of each unit used in a game varies all the time, meaning the agents try to adapt their strategies to beat their opponents.[12]

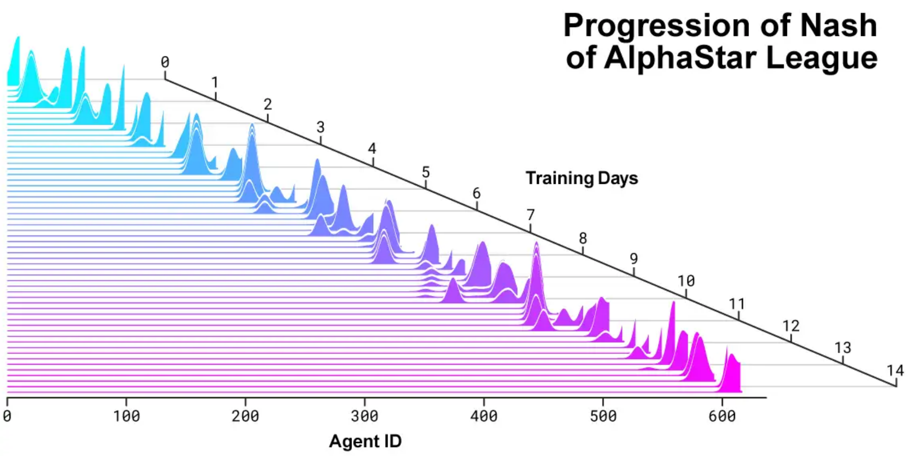

Figure 5: Progress of Nash distribution over training days. The Nash distribution indicates, which agents will be picked by a meta-player (see section 2.3). Latest agents will be picked most times.[12]

4.2 Play comparison to humans

As the AI is a computer, the question rises, whether it can react and control the game quicker than a human does. For previous AI, parallel actions and focus points were possible, which is seen as an advantage over human players. For AlphaStar, the actions-per-minute (APM) were measured (fig. 6). The mean APM is actually lower than that of a pro player, but the AI is still able to burst APM to a superhuman level, which can yield advantages over human players at certain points. One reason for the lower mean APM could be that the AI is more precise in its actions in both terms of mechanical precision as well as future planning and thus needs less total actions.
The first version of AlphaStar used the game interpretation offered by the SC2LE (fig. 1), which is a different representation of what a human observes, although the AI does not have any additional information. To make it even more comparable to human performance, the second version was trained using the raw camera interface (fig. 7).

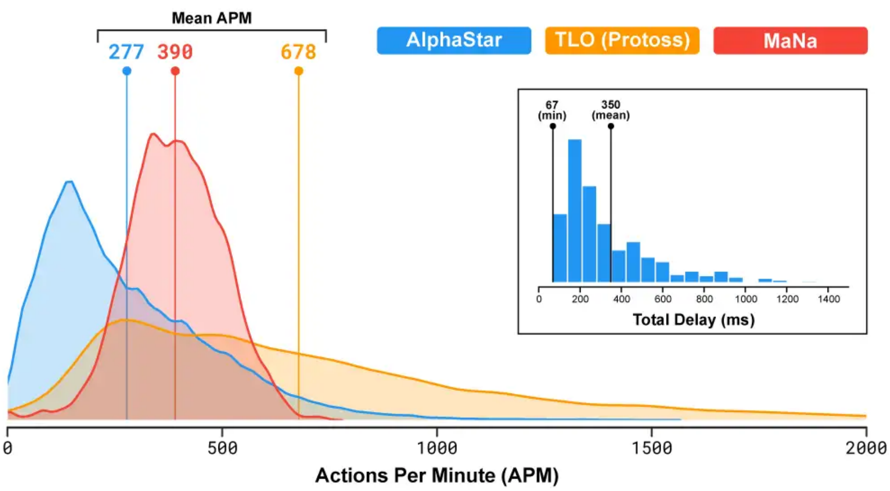

Figure 6: Mean APM of AlphaStar compared to pro players. Note that player TLO uses special hotkeys, which results in much higher APM than normally possible. Also note that AlphaStar can have much higher APM bursts compared to the player MaNa, which means it can outplay humans in some situations, when necessary.[12]

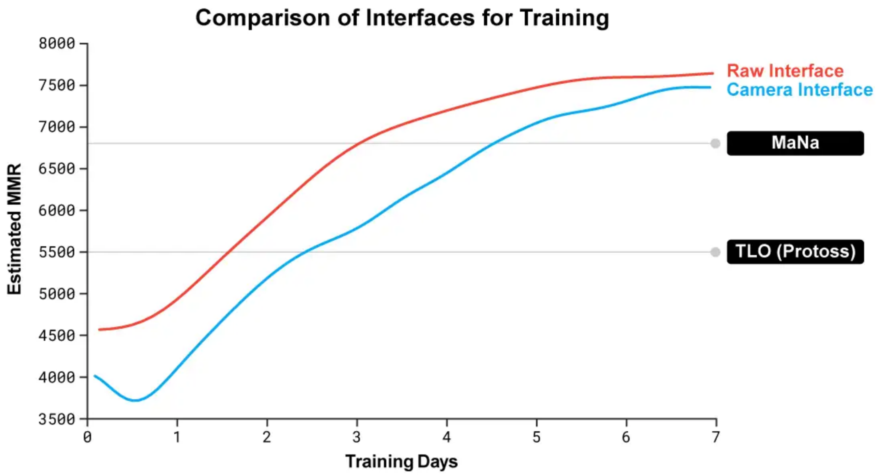

Figure 7: Comparison of MMR progression during training. It can be seen that the camera interface starts with a lower performance but approaches the raw interface.[12]

4.3 Empirical Evaluation

Recently, AlphaStar was launched on Battle.net to play against human players.[5] Although the name of the AI was not be revealed to the public, after a short amount of time, the StarCraft Battle.net community has collected evidence for three certain Battle.net Accounts to be controlled by AlphaStar.

The name of the AlphaStar accounts is really special. It consists of the tiny letter "L" and the big letter "I", which in some fonts looks the same. As there are three accounts, each one of them only plays a certain StarCraft race. While in the game, the names look the same, the name can be decoded by replacing the "L" by 1 and the "I" by 0, resulting in: 000000011111, which is binary and stands for 31. See table 1 for details. The links to the Battle.net Profiles are listed in the bibliography. [1, 2, 3]

Name	Binary	Decimal	Race
IIIIIIIlllll	000000011111	31	Protoss
IIIIIIlIIIII	000000100000	32	Terran
IIIIIIlIIIIl	000000100001	33	Zerg

Table 1: AlphaStar Battle.net names, their binary encoding, decimal equivalent and corresponding race.

It is known, that Protoss, which has the number of 31, was the first race the AI could play, so we can assume that Terran followed and last but not least Zerg was created. This could either be to resource limitations on the training part or due to individual changes to the network for each race, as they come with their own peculiarities respectively.
For a StarCraft player, this name would be very special. But this is not the only indicating for being the AlphaStar AI. One more indicator is that it does not use unit groups in the game, which all human players do to manage their units. The last indicator is the strategies that are observed in the games of these accounts. It is playing different from humans but still has a high win rate.
After this evidence has been collected, many StarCraft commentators on YouTube and Twitch started collecting the replays of the games and creating commented streams, analyzing the play style of AlphaStar.
During the games, it can be observed that the AI has about the same average APM as human players. But it is able to performs more precise actions compared to humans, like selecting and moving units in a special way or switching screen vision for less than a second to perform an action and as already said before, it does not use unit groups. So in some cases it can act faster than a human could in the same situation. On the other hand, it is not perfect, sometimes making not the best decision and thus loosing some units.
When it comes to strategy, AlphaStar does not seem to have a certain best strategy that is always used. Like human players, it will do an early harassment of the enemy base but it does not seem to play any rush strategies as these are very risky.
AlphaStar has lost some games against humans. Some unconventional strategies have had success against it. It seemed to have trouble to adapt to this unconventional play style. Successful strategic elements were a Nydus rush and flying units. AlphaStar has sometimes trouble dealing with flying units, as not every unit can attack a flying unit and AlphaStar probably does not know which unit to build against these.
In total, AlphaStar won around 52 of 60 games with each race, ending up in Master league. This is a solid result, but we can expect a better performance in future. AlphaStar has stopped playing on the Battle.net again and its creators are probably working on improvements.

5 Conclusion

The performance of AlphaStar is already on Master level, but does not outperforms humans completely. This displays the difficulty of StarCraft and the challenge of creating an AI for such a complex game. Many novel techniques have been combined to create AlphaStar and after the evaluation of it on the Battle.net, we can assume that its creators will work on improvements to make it even better. Some of its performance may be due to superhuman possibilities, like really small reaction times or machine-precision where humans have to use a mouse for their inputs. But maybe some restrictions can be made to the AI to limit their APM to a reasonable maximum or adding delay to inputs. When the creators of AlphaStar release new information, its progress can be evaluated easier.

References

[1] Profile Summary AlphaStar (Protoss), 2019. URL `https://starcraft2.com/en-us/profile/2/1/8789661`.

[2] Profile Summary AlphaStar (Terran), 2019. URL `https://starcraft2.com/en-us/profile/2/1/8789863`.

[3] Profile Summary AlphaStar (Zerg), 2019. URL `https://starcraft2.com/en-us/profile/2/1/8789693`.

[4] David Balduzzi. Re-evaluating Evaluation. (NeurIPS), 2018.

[5] Blizzard Entertainment. DeepMind Research on Ladder, 2019. URL `https://starcraft2.com/en-us/news/22933138`.

[6] Arpad E. Elo. *The rating of chessplayers, past and present.* Arco Pub., New York, 1978. ISBN 0668047216 9780668047210. URL `http://www.amazon.com/Rating-Chess-Players-Past-Present/dp/0668047216`.

[7] Thore Graepel. AlphaGo - Mastering the game of go with deep neural networks and tree search. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9852 LNAI(7585):XXI, 2016. ISSN 16113349. doi: 10.1038/nature16961. URL `http://dx.doi.org/10.1038/nature16961`.

[8] David M. Kreps. *Nash Equilibrium*, pages 167–177. Palgrave Macmillan UK, London, 1989. ISBN 978-1-349-20181-5. doi: 10.1007/978-1-349-20181-5_19. URL `https://doi.org/10.1007/978-1-349-20181-5_19`.

[9] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, Yutian Chen, Timothy Lillicrap, Fan Hui, Laurent Sifre, George Van Den Driessche, Thore Graepel, and Demis Hassabis. Mastering the game of Go without human knowledge. *Nature*, 550(7676):354–359, 2017. ISSN 14764687. doi: 10.1038/nature24270. URL `http://dx.doi.org/10.1038/nature24270`.

[10] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention

is all you need. *Advances in Neural Information Processing Systems*, 2017-December(Nips):5999–6009, 2017. ISSN 10495258.

[11] Oriol Vinyals, Timo Ewalds, Sergey Bartunov, Petko Georgiev, Alexander Sasha Vezhnevets, Michelle Yeo, Alireza Makhzani, Heinrich Küttler, John Agapiou, Julian Schrittwieser, John Quan, Stephen Gaffney, Stig Petersen, Karen Simonyan, Tom Schaul, Hado van Hasselt, David Silver, Timothy Lillicrap, Kevin Calderone, Paul Keet, Anthony Brunasso, David Lawrence, Anders Ekermo, Jacob Repp, and Rodney Tsing. StarCraft II: A New Challenge for Reinforcement Learning. *CoRR*, abs/1708.0, 2017. URL `http://arxiv.org/abs/1708.04782`.

[12] Oriol Vinyals, Igor Babuschkin, Junyoung Chung, Michael Mathieu, Max Jaderberg, Wojciech M Czarnecki, Andrew Dudzik, Aja Huang, Petko Georgiev, Richard Powell, Timo Ewalds, Dan Horgan, Manuel Kroiss, Ivo Danihelka, John Agapiou, Junhyuk Oh, Valentin Dalibard, David Choi, Laurent Sifre, Yury Sulsky, Sasha Vezhnevets, James Molloy, Trevor Cai, David Budden, Tom Paine, Caglar Gulcehre, Ziyu Wang, Tobias Pfaff, Toby Pohlen, Yuhuai Wu, Dani Yogatama, Julia Cohen, Katrina McKinney, Oliver Smith, Tom Schaul, Timothy Lillicrap, Chris Apps, Koray Kavukcuoglu, Demis Hassabis, and David Silver. AlphaStar: Mastering the Real-Time Strategy Game StarCraft II, 2019. URL `https://deepmind.com/blog/alphastar-mastering-real-time-strategy-game-starcraft-ii/`.