BLOG POST

23 DEC 2020

MuZero: Mastering Go, chess, shogi and Atari without rules

In 2016, we introduced AlphaGo, the first artificial intelligence (AI) program to defeat humans at the ancient game of Go. Two years later, its successor – AlphaZero – learned from scratch to master Go, chess and shogi. Now, in a paper in the journal Nature, we describe MuZero, a significant step forward in the pursuit of general-purpose algorithms. MuZero masters Go, chess, shogi and Atari without needing to be told the rules, thanks to its ability to plan winning strategies in unknown environments.

For many years, researchers have sought methods that can both learn a model that explains their environment, and can then use that model to plan the best course of action. Until now, most approaches have struggled to plan effectively in domains, such as Atari, where the rules or dynamics are typically unknown and complex.

MuZero, first introduced in a preliminary paper in 2019, solves this problem by learning a model that focuses only on the most important aspects of the environment for planning. By combining this model with AlphaZero's powerful lookahead tree search, MuZero set a new state of the art result on the Atari benchmark, while simultaneously matching the performance of AlphaZero in the classic planning challenges of Go, chess and shogi. In doing so, MuZero demonstrates a significant leap forward in the capabilities of reinforcement learning algorithms.

Generalising to unknown models

The ability to plan is an important part of human intelligence, allowing us to solve problems and make decisions about the future. For example, if we see dark clouds forming, we might predict it will rain and decide to take an umbrella with us before we venture out. Humans learn this ability quickly and can generalise to new scenarios, a trait we would also like our

Researchers have tried to tackle this major challenge in AI by using two main approaches: lookahead search or model-based planning.

Systems that use lookahead search, such as AlphaZero, have achieved remarkable success in classic games such as checkers, chess and poker, but rely on being given knowledge of their environment's dynamics, such as the rules of the game or an accurate simulator. This makes it difficult to apply them to messy real world problems, which are typically complex and hard to distill into simple rules.

Model-based systems aim to address this issue by learning an accurate model of an environment's dynamics, and then using it to plan. However, the complexity of modelling every aspect of an environment has meant these algorithms are unable to compete in visually rich domains, such as Atari. Until now, the best results on Atari are from model-free systems, such as DQN, R2D2 and Agent57. As the name suggests, model-free algorithms do not use a learned model and instead estimate what is the best action to take next.

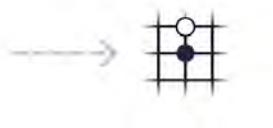
MuZero uses a different approach to overcome the limitations of previous approaches. Instead of trying to model the entire environment, MuZero just models aspects that are important to the agent's decision-making process. After all, knowing an umbrella will keep you dry is more useful to know than modelling the pattern of raindrops in the air.

Specifically, MuZero models three elements of the environment that are critical to planning:

- The **value:** how good is the current position?
- The **policy:** which action is the best to take?
- The **reward:** how good was the last action?

These are all learned using a deep neural network and are all that is needed for MuZero to understand what happens when it takes a certain action and to plan accordingly.

ILLUSTRATION OF HOW MONTE CARLO TREE SEARCH CAN BE USED TO PLAN WITH THE MUZERO NEURAL NETWORKS. STARTING AT THE CURRENT POSITION IN THE GAME (SCHEMATIC GO BOARD AT THE TOP OF THE ANIMATION), MUZERO USES THE REPRESENTATION FUNCTION (H) TO MAP FROM THE OBSERVATION TO AN EMBEDDING USED BY THE NEURAL NETWORK (S0). USING THE DYNAMICS FUNCTION (G) AND THE PREDICTION FUNCTION (F), MUZERO CAN THEN CONSIDER POSSIBLE FUTURE SEQUENCES OF ACTIONS (A), AND CHOOSE THE BEST ACTION.



FROM THE ENVIRONMENT, AS WELL AS THE RESULTS OF SEARCHES PERFORMED WHEN DECIDING ON THE BEST ACTION.

DURING TRAINING, THE MODEL IS UNROLLED ALONGSIDE THE COLLECTED EXPERIENCE, AT EACH STEP PREDICTING THE PREVIOUSLY SAVED INFORMATION: THE VALUE FUNCTION V PREDICTS THE SUM OF OBSERVED REWARDS (U), THE POLICY ESTIMATE (P) PREDICTS THE PREVIOUS SEARCH OUTCOME (Π), THE REWARD ESTIMATE R PREDICTS THE LAST OBSERVED REWARD (U).

This approach comes with another major benefit: MuZero can repeatedly use its learned model to improve its planning, rather than collecting new data from the environment. For example, in tests on the Atari suite, this variant - known as MuZero Reanalyze - used the learned model 90% of the time to re-plan what should have been done in past episodes.

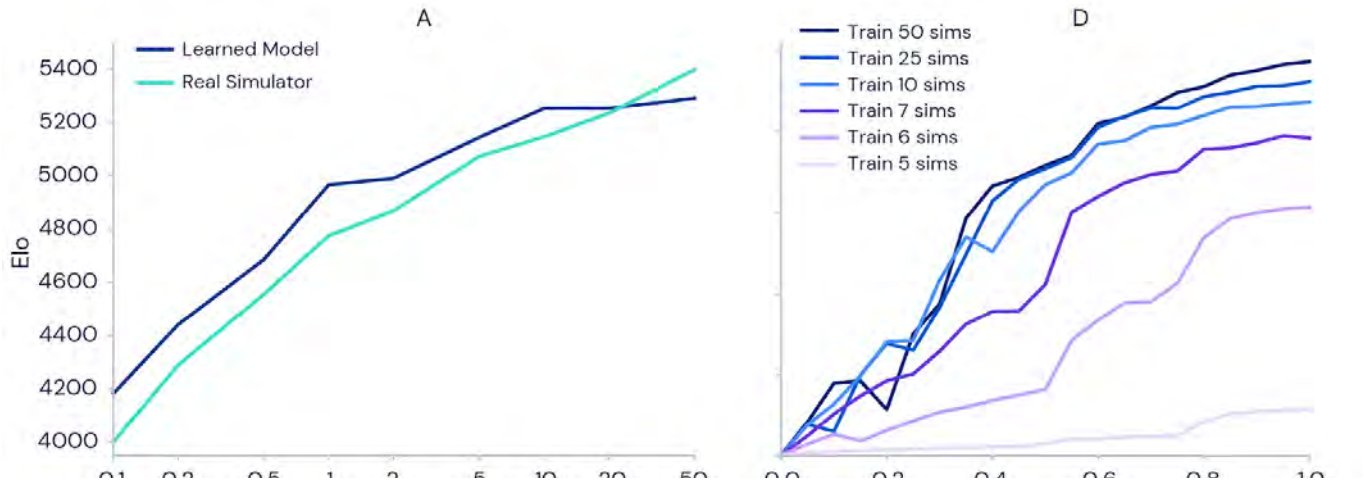
MuZero performance

We chose four different domains to test MuZeros capabilities. Go, chess and shogi were used to assess its performance on challenging planning problems, while we used the Atari suite as a benchmark for more visually complex problems. In all cases, MuZero set a new state of the art for reinforcement learning algorithms, outperforming all prior algorithms on the Atari suite and matching the superhuman performance of AlphaZero on Go, chess and

Agent	Median	Mean	Env. Frames
Ape-X	434.1%	1695.6%	22.8B
R2D2	1920.6%	4024.9%	37.5B
MuZero	**2041.1%**	**4999.2%**	20.0B
IMPALA	191.8%	957.6%	200M
Rainbow	231.1%	-	200M
UNREAL	250%	880%	200M
LASER	431%	-	200M
MuZero Reanalyse	**731.1%**	**2168.9%**	200M

PERFORMANCE ON THE ATARI SUITE USING EITHER 200M OR 20B FRAMES PER TRAINING RUN. MUZERO ACHIEVES A NEW STATE OF THE ART IN BOTH SETTINGS. ALL SCORES ARE NORMALISED TO THE PERFORMANCE OF HUMAN TESTERS (100%), WITH THE BEST RESULTS FOR EACH SETTING HIGHLIGHTED IN BOLD.

We also tested how well MuZero can plan with its learned model in more detail. We started with the classic precision planning challenge in Go, where a single move can mean the difference between winning and losing. To confirm the intuition that planning more should lead to better results, we measured how much stronger a fully trained version of MuZero can become when given more time to plan for each move (see left hand graph below). The results showed that playing strength increases by more than 1000 Elo (a measure of a player's relative skill) as we increase the time per move from one-tenth of a second to 50 seconds. This is similar to the difference between a strong amateur player and the strongest professional player.



LEFT: PLAYING STRENGTH IN GO INCREASES SIGNIFICANTLY AS THE TIME AVAILABLE TO PLAN EACH MOVE INCREASES. NOTE HOW MUZERO'S SCALING ALMOST PERFECTLY MATCHES THAT OF ALPHAZERO, WHICH HAS ACCESS TO A PERFECT SIMULATOR. RIGHT: THE SCORE IN THE ATARI GAME MS PAC-MAN ALSO INCREASES WITH THE AMOUNT OF PLANNING PER MOVE DURING TRAINING. EACH PLOT SHOWS A DIFFERENT TRAINING RUN WHERE MUZERO WAS ALLOWED TO CONSIDER A DIFFERENT NUMBER OF SIMULATIONS PER MOVE.

To test whether planning also brings benefits throughout training, we ran a set of experiments on the Atari game Ms Pac-Man (right hand graph above) using separate trained instances of MuZero. Each one was allowed to consider a different number of planning simulations per move, ranging from five to 50. The results confirmed that increasing the amount of planning for each move allows MuZero to both learn faster and achieve better final performance.

Interestingly, when MuZero was only allowed to consider six or seven simulations per move - a number too small to cover all the available actions in Ms Pac-Man - it still achieved good performance. This suggests MuZero is able to generalise between actions and situations, and does not need to exhaustively search all possibilities to learn effectively.

New horizons

MuZero's ability to both learn a model of its environment and use it to successfully plan demonstrates a significant advance in reinforcement learning and the pursuit of general purpose algorithms. Its predecessor, AlphaZero, has already been applied to a range of complex problems in chemistry, quantum physics and beyond. The ideas behind MuZero's powerful learning and planning algorithms may pave the way towards tackling new challenges in robotics, industrial systems and other messy real-world environments where the "rules of the game" are not known.

Design by Adam Cain, Jim Kynvin and Aleksandrs Polozuns

- MuZero: Nature Paper
- MuZero talks: NeurIPS (9 mins, Dec 2019), ICAPS (30 mins, Oct 2020)

- AlphaGo: Blog | Paper
- AlphaGo Zero: Blog | Paper
- AlphaZero: Blog | Paper

Further reading