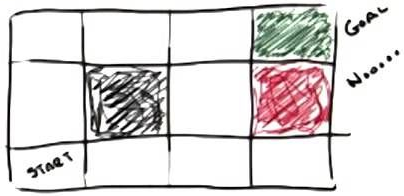

AlphaGo Zero
Starting from scratch

1) MDP
2) MCTS
3) Q-Learning
4) Residual neural networks

Markov Decision Processes



STATES : S

MODEL : $T(s, a, s') \sim Pr(s' | s, a)$ UP, DOWN, LEFT, RIGHT

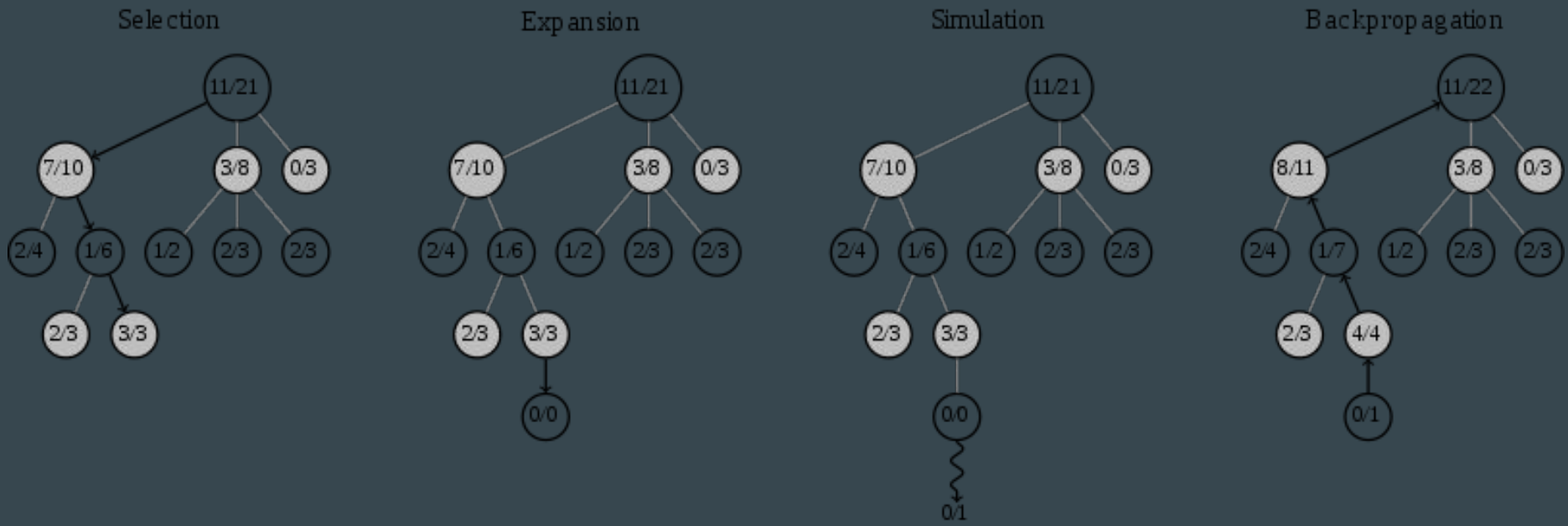
ACTIONS : $A(s)$, A

REWARD : $R(s)$, $R(s, a)$, $R(s, a, s')$

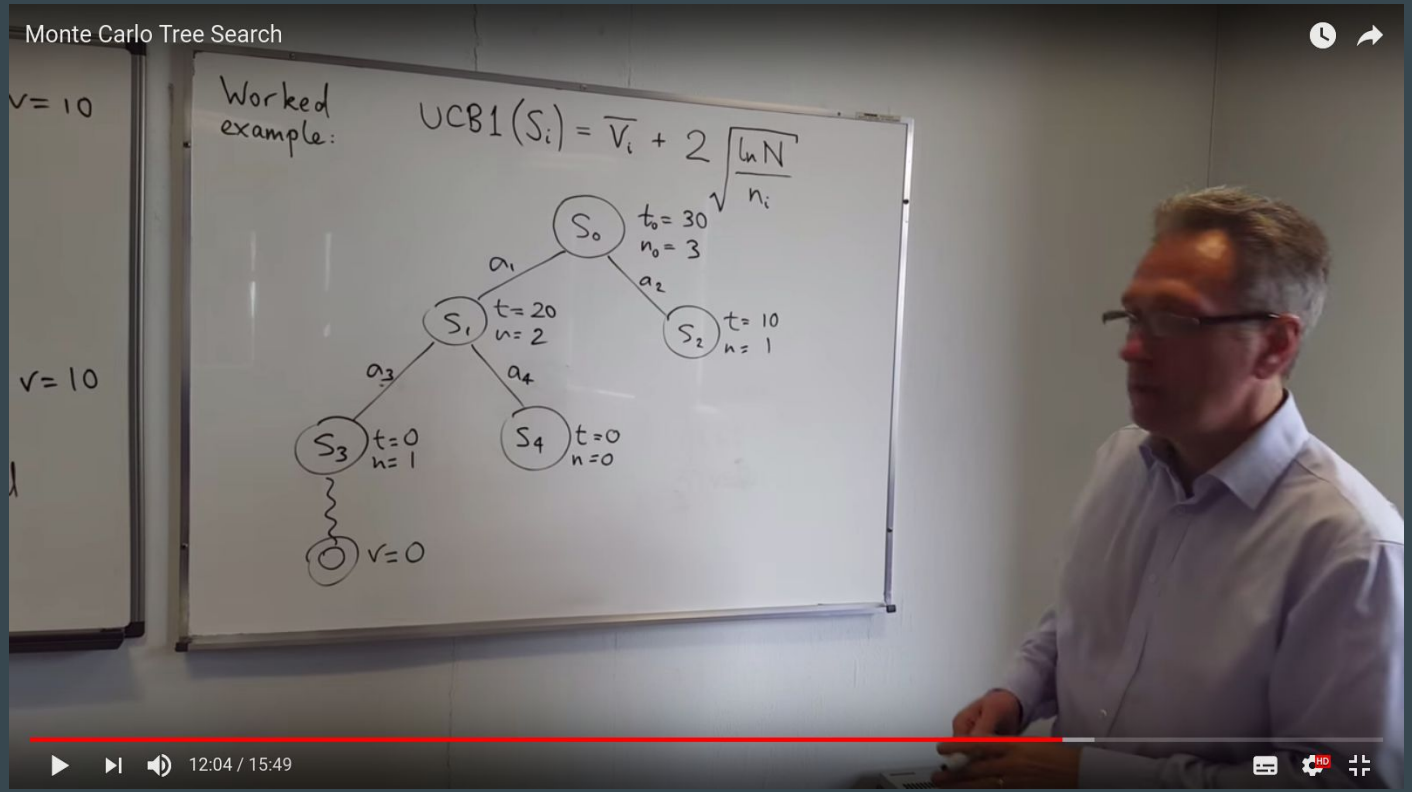
POLICY : $\pi(s) \rightarrow a$

π^*

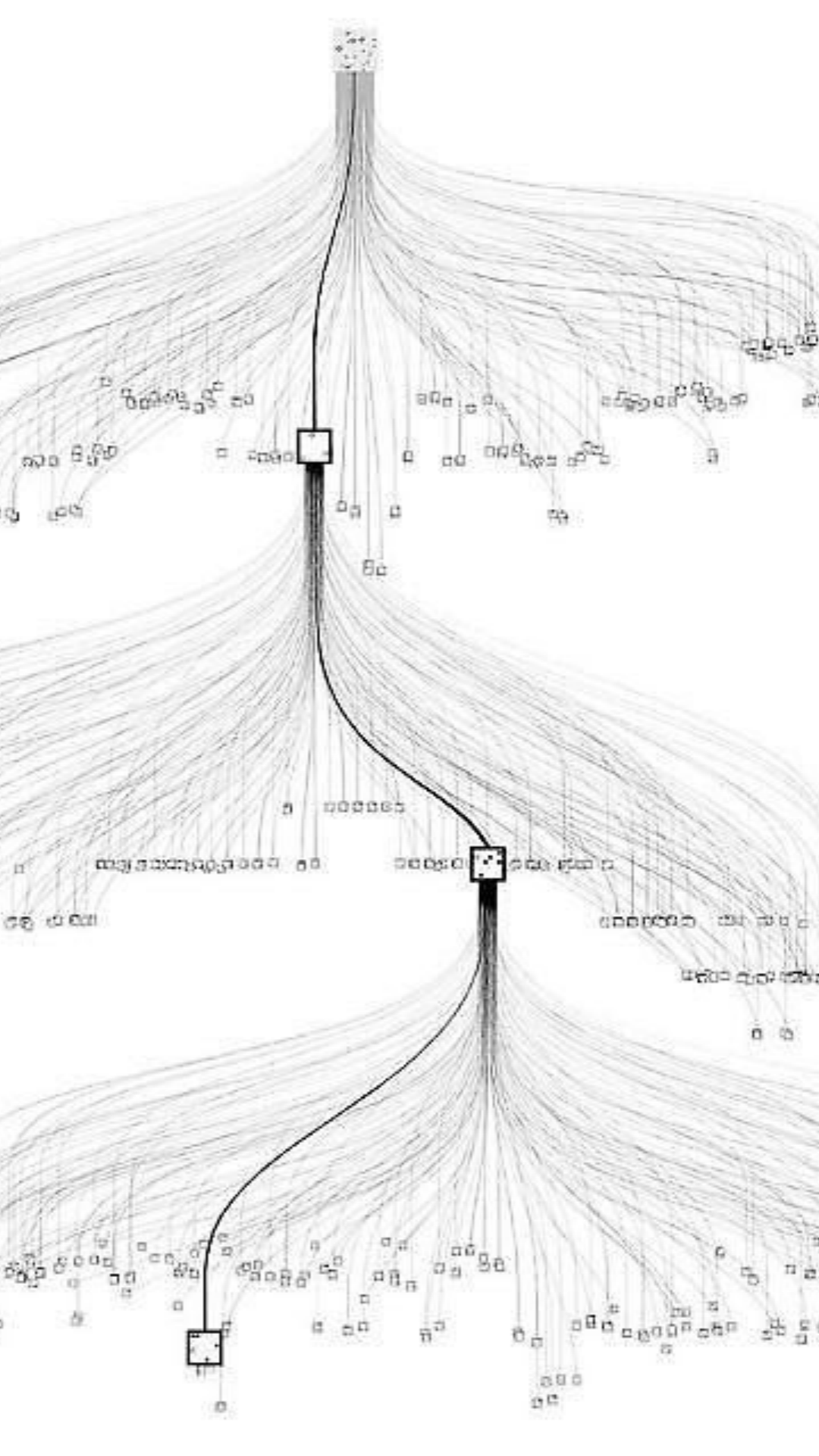
Monte Carlo Tree Search



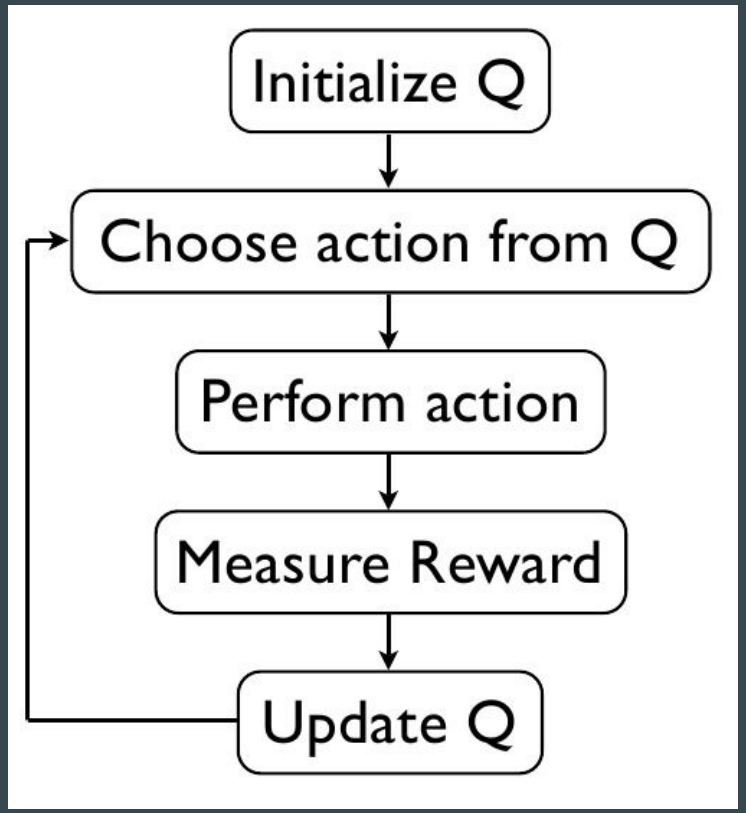
Monte Carlo Tree Search



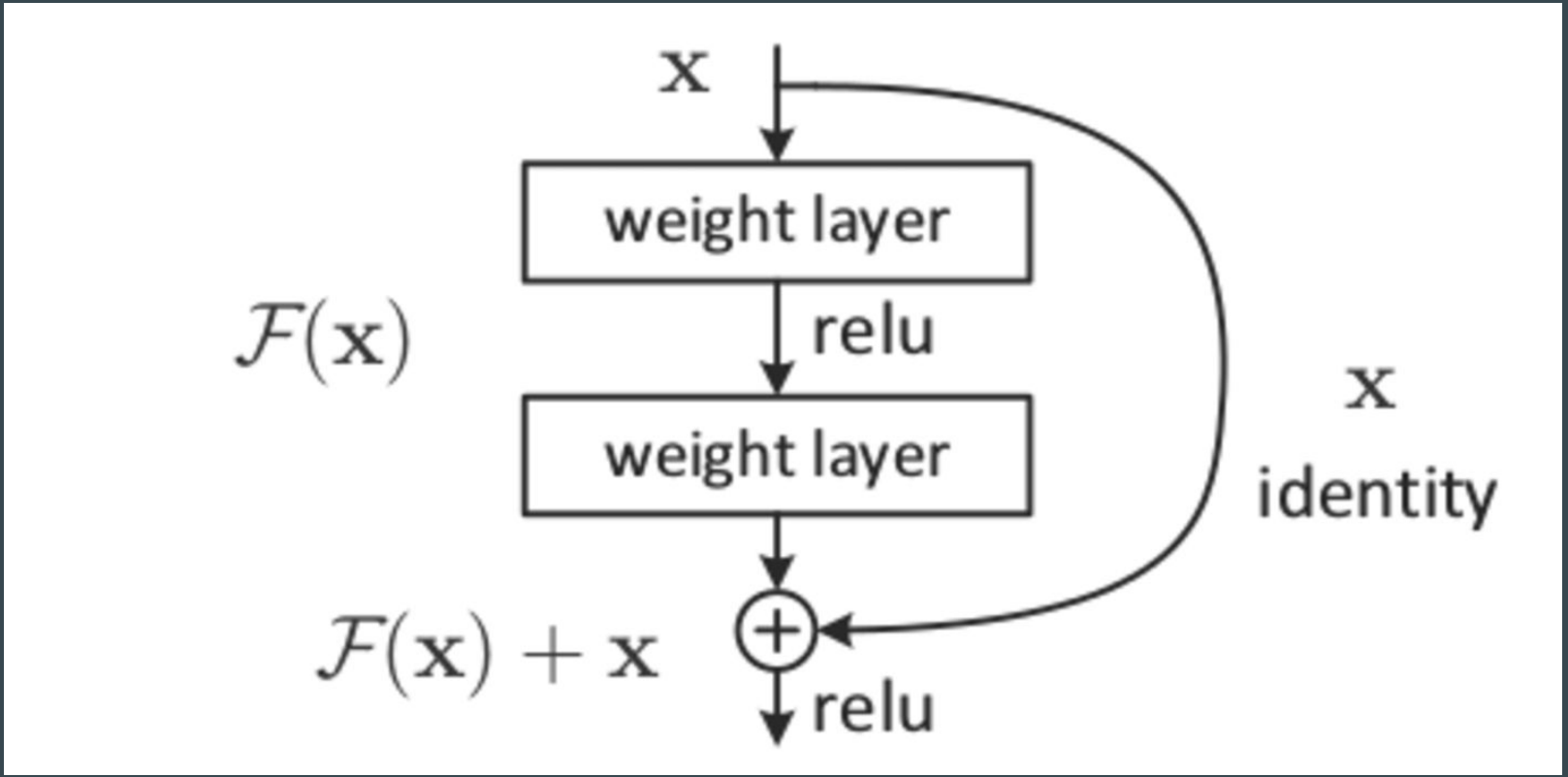
source : https://www.youtube.com/watch?v=UXW2yZndl7U&t=127s



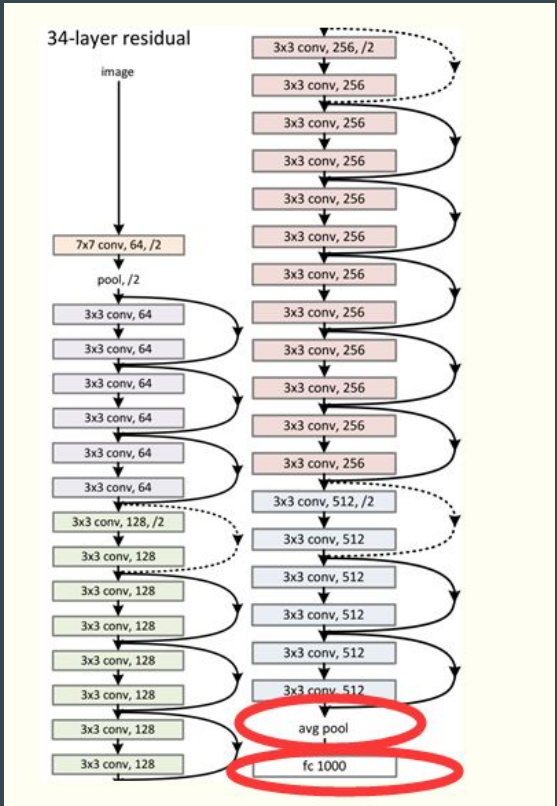
Q-learning


Initialize Q
Choose action from Q
Perform action
Measure Reward
Update Q

Residual Neural Networks



Residual Neural Networks



1) To accelerate the speed of training of the deep networks

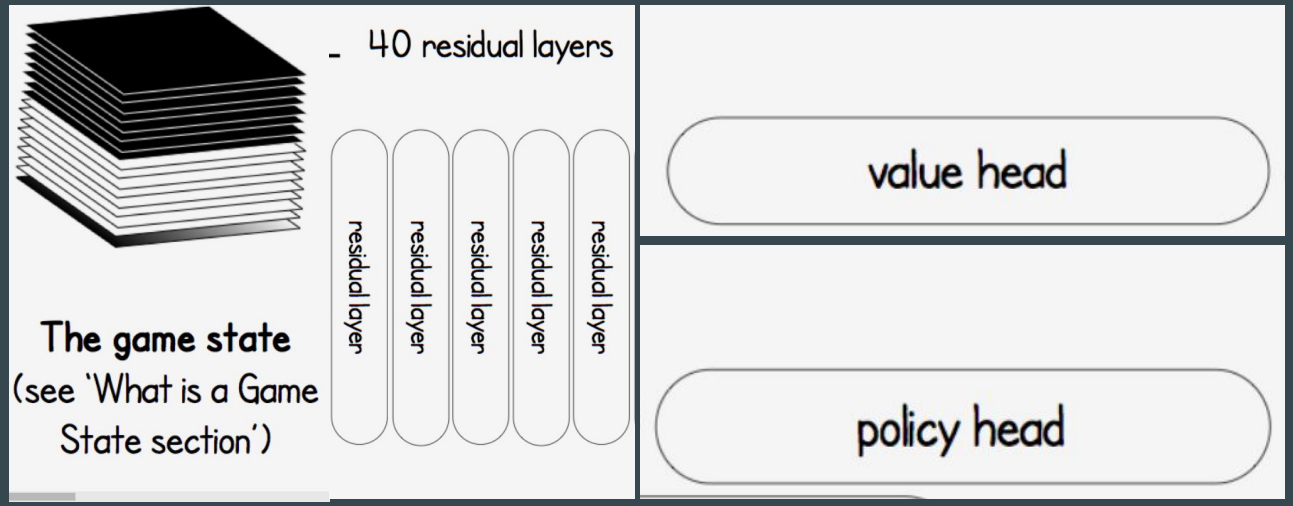
2) Instead of widen the network, increasing depth of the network results in less extra parameters

3) Reducing the effect of Vanishing Gradient Problem

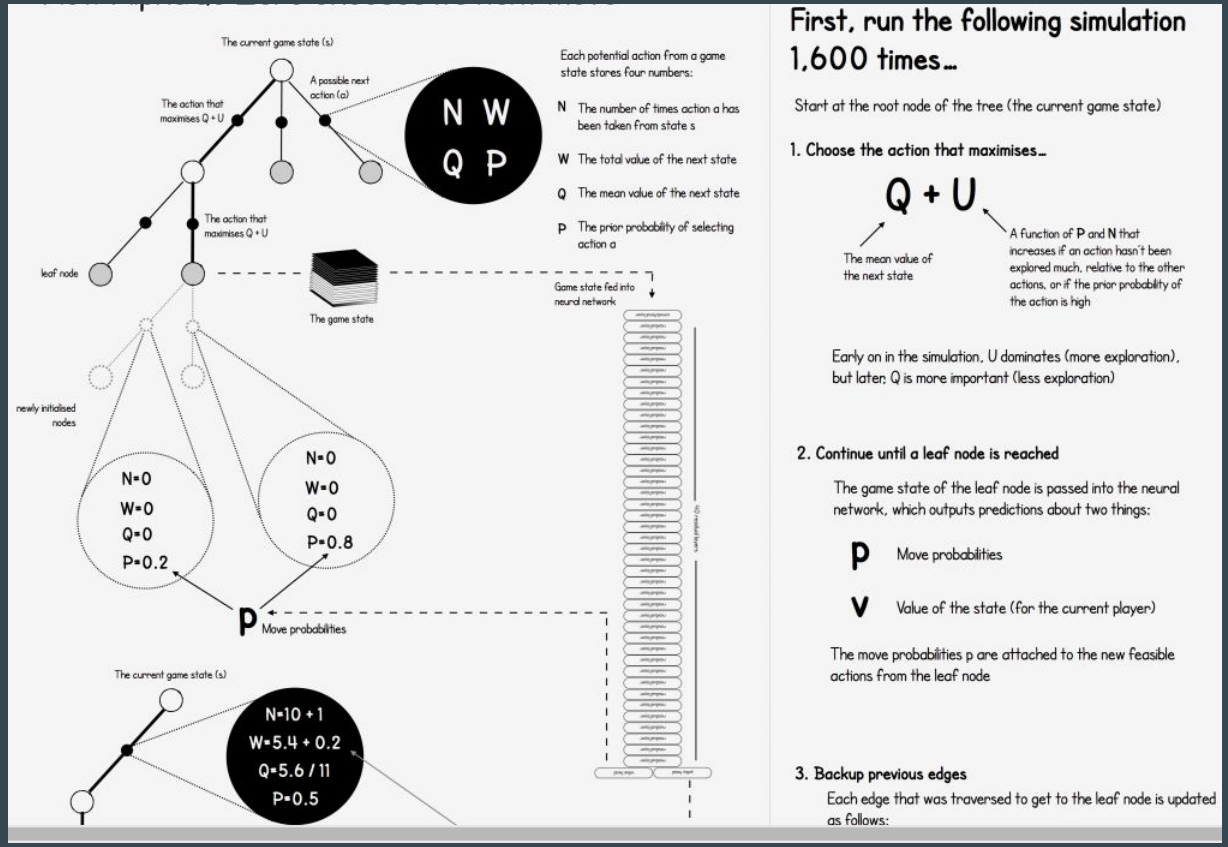
4) Obtaining higher accuracy in network performance especially in Image Classification

Residual Neural Networks

2 output :



Neural network in MCTS



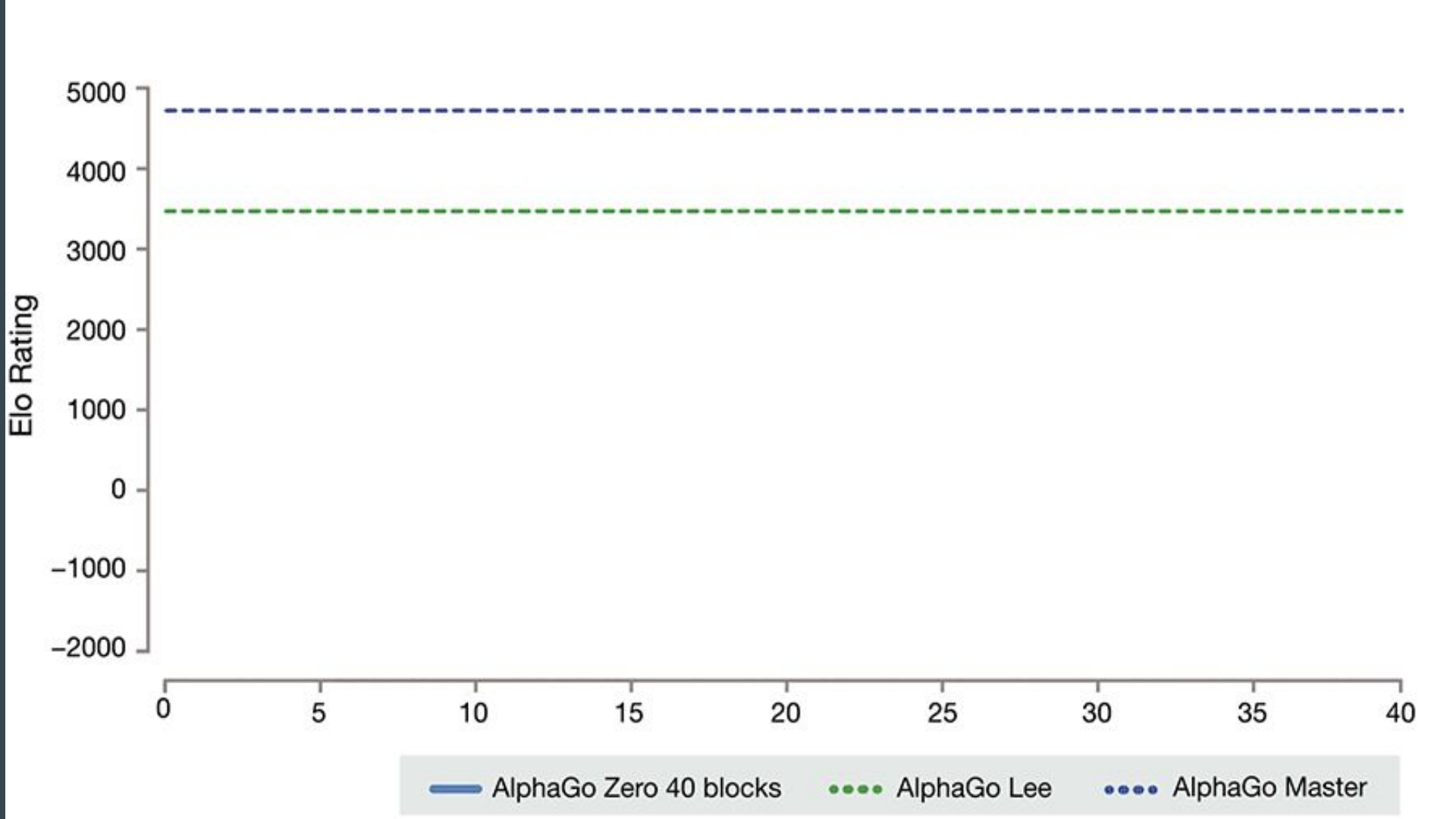
How Residual neural networks Learn ?

1600 simulations for a moove
25000 games against itself => Training site
1000 training loops : 2048 positions takes randomly in 500 000 games => Train
400 games against between the latest best neural network => Evaluate
if the new neural network is better than the previous one he becomes the best

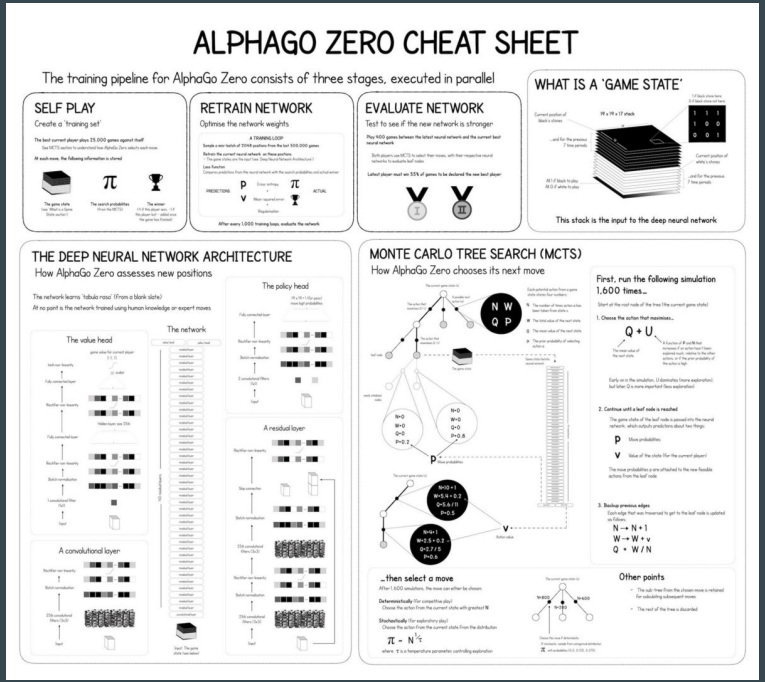
How many GPU ?

Learning :
19 CPU
64 GPU

Inference :
4 TPU


5000
4000
3000
2000
1000
0
−1000
−2000
Elo Rating
0
5
10
15
20
25
30
35
40
AlphaGo Zero 40 blocks
AlphaGo Lee
AlphaGo Master

ALPHAGO ZERO CHEAT SHEET



https://applied-data.science/static/main/res/alpha_go_zero_cheat_sheet.png