

AlphaZero – najlepszy silnik szachowy



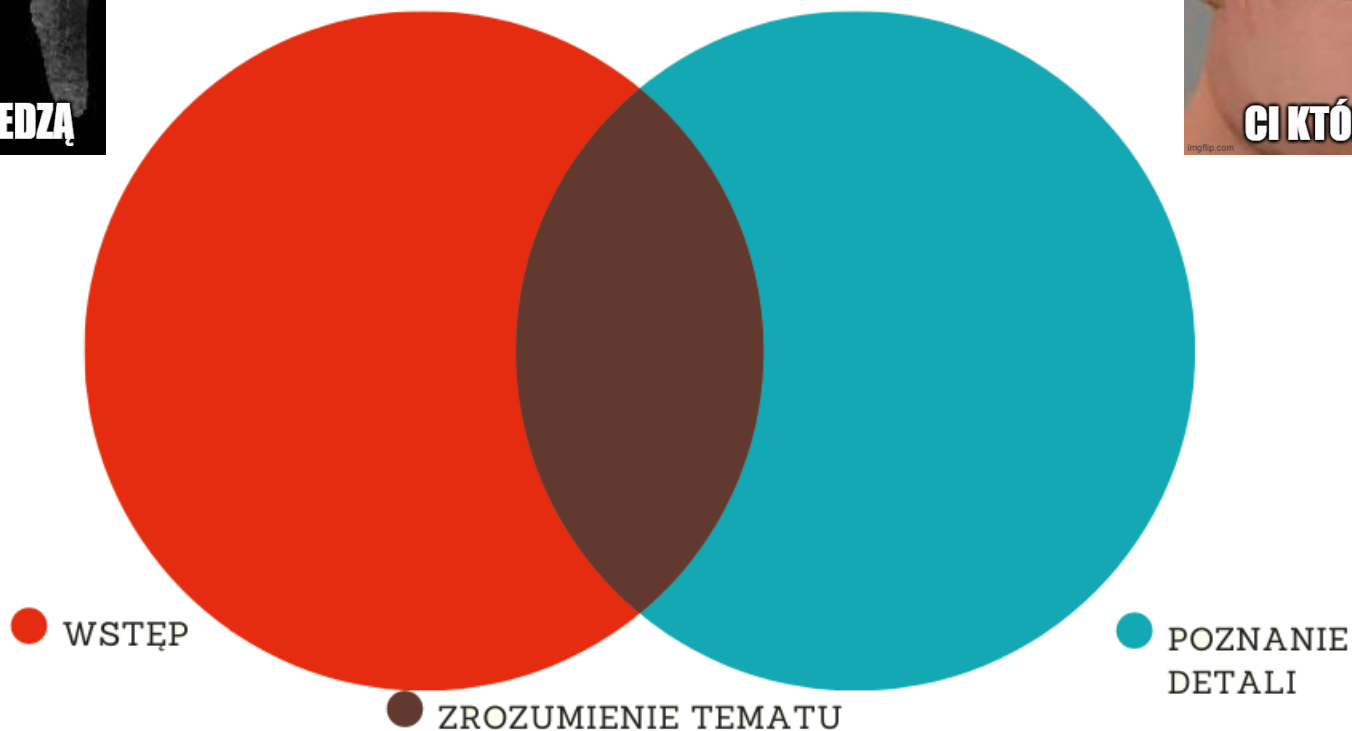
Krystian Kurek
Wydział MiNI
PW

Dlaczego to jest ciekawe?

1. Najlepszy silnik szachowy – pokonał ówczesnego mistrza.
2. Algorytm AlphaZero jest generyczny.
3. Cały trening opiera się tylko na graniu ze sobą.



Dla każdego, coś dobrego



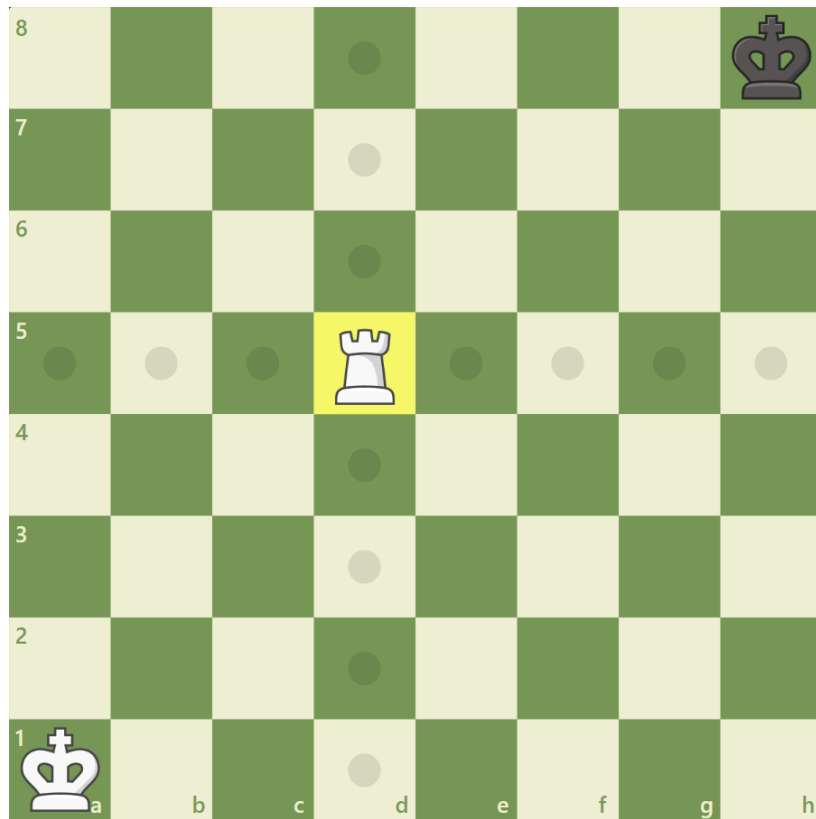
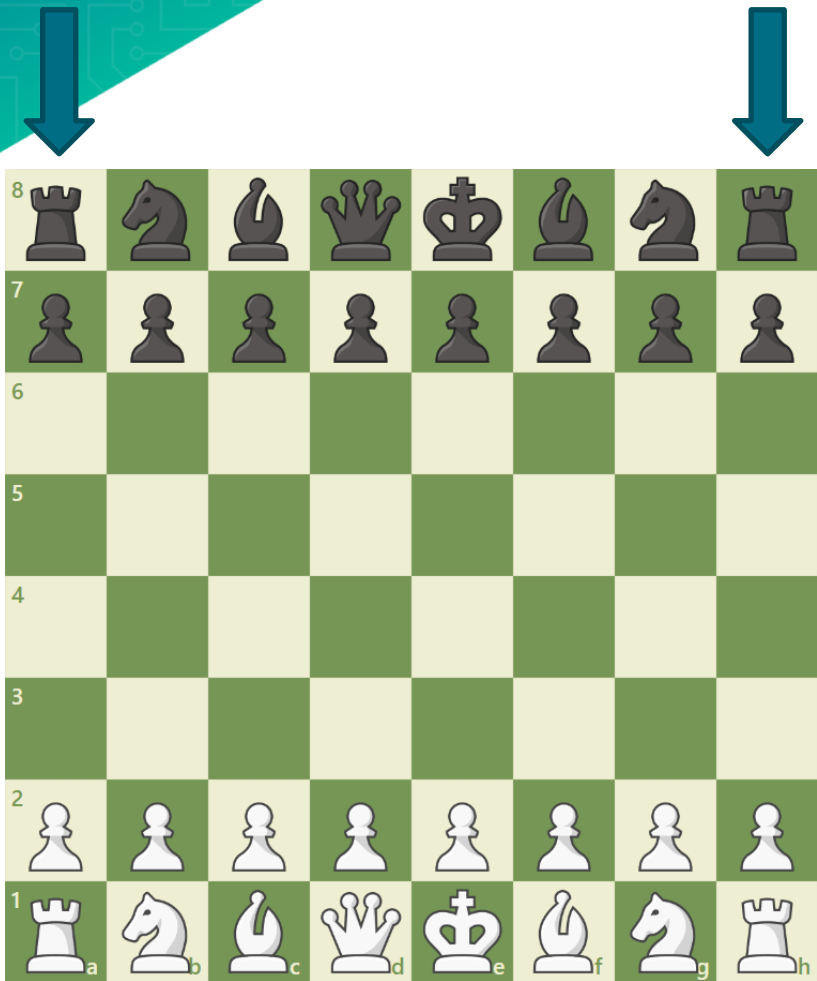
Plan prezentacji

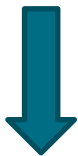
1. Zasady gry w szachy.
2. Monte-Carlo Tree Search.
3. Podstawy sztucznych sieci neuronowych.
4. Zasada działania AlphaZero.
5. Wyniki starcia z mistrzem.

Zasady gry w szachy

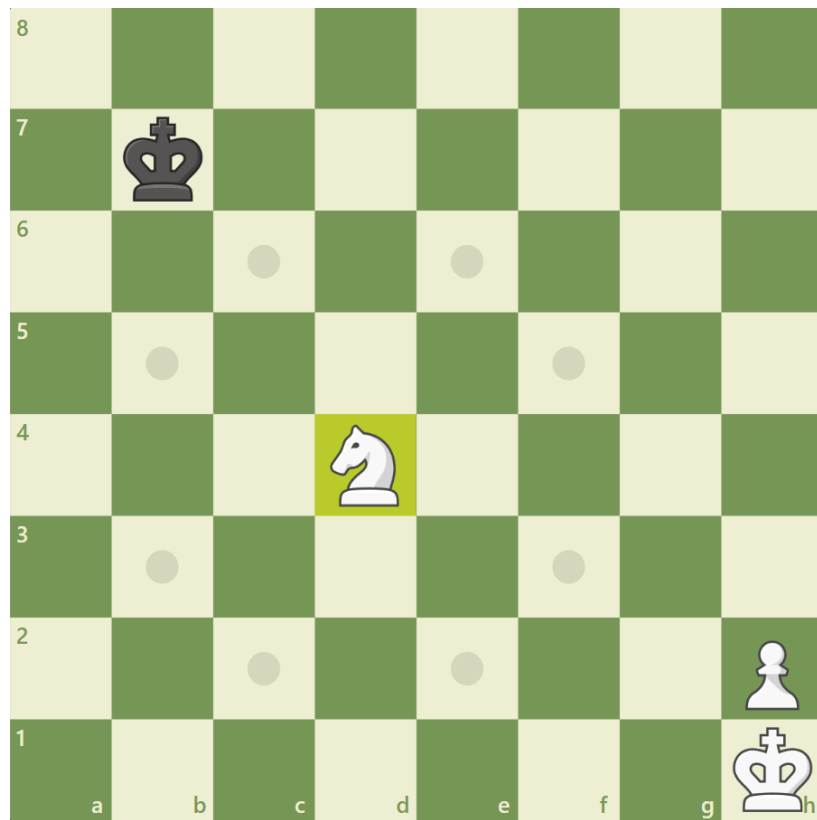
- Dwóch graczy,
- Kwadratowa plansza 8x8.
- 32 bierki, 6 unikalnych rodzajów bierek,
- Celem gry jest zmatowanie króla przeciwnika.

Wieża



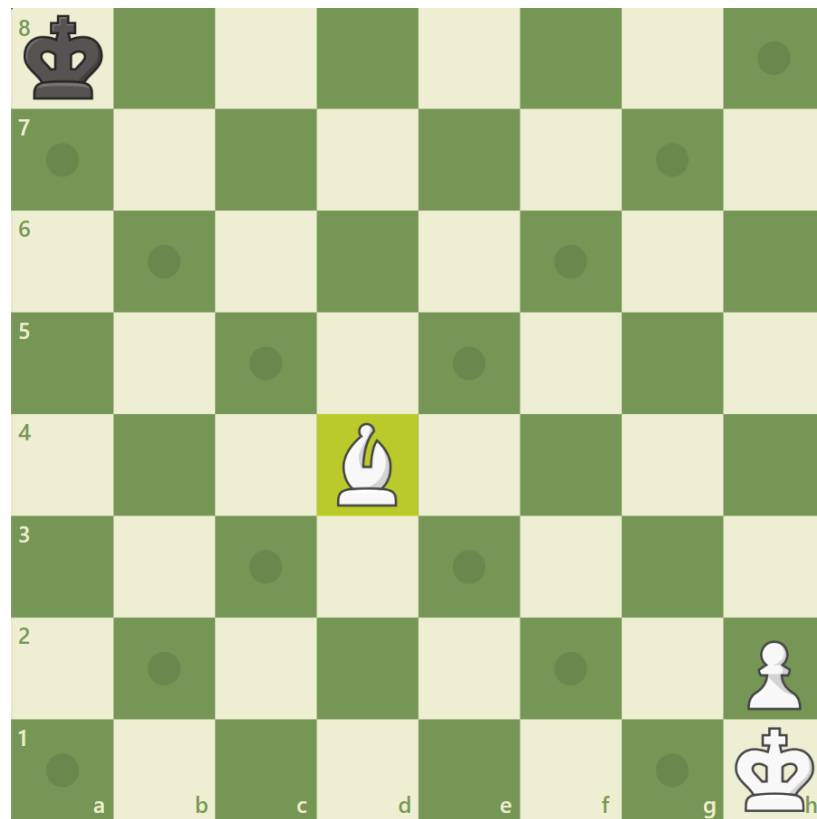


Skoczek



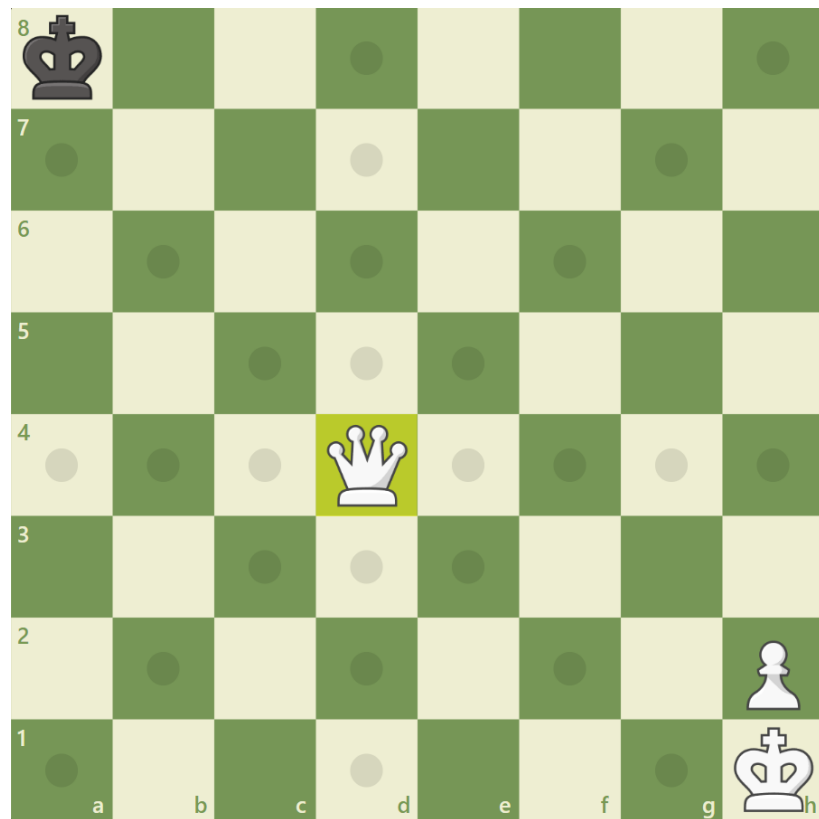


Goniec



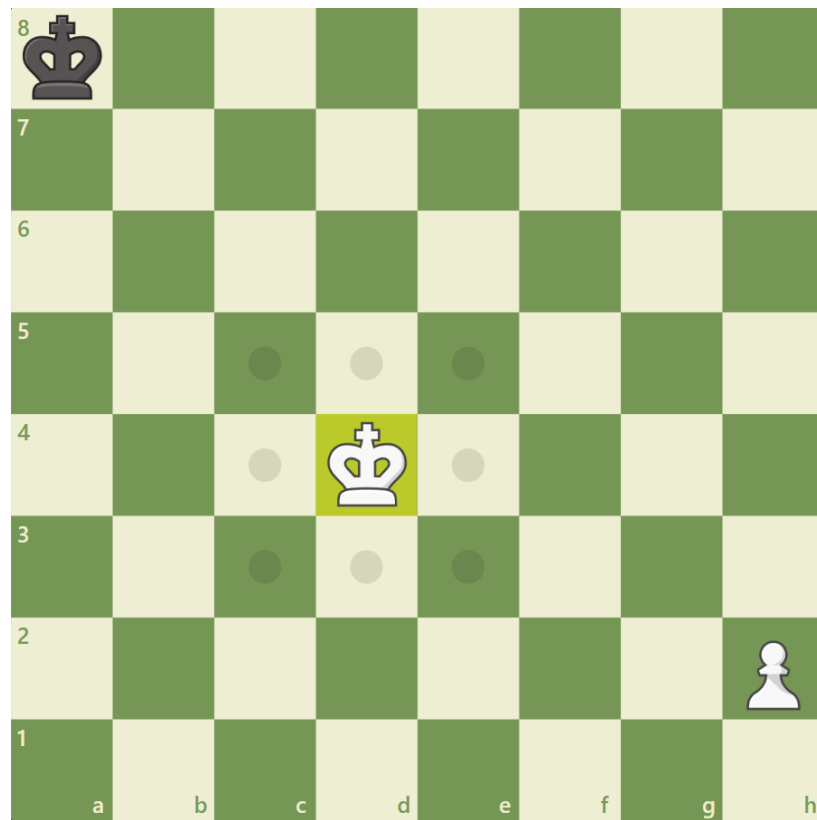


Hetman





Król

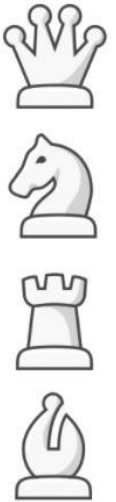




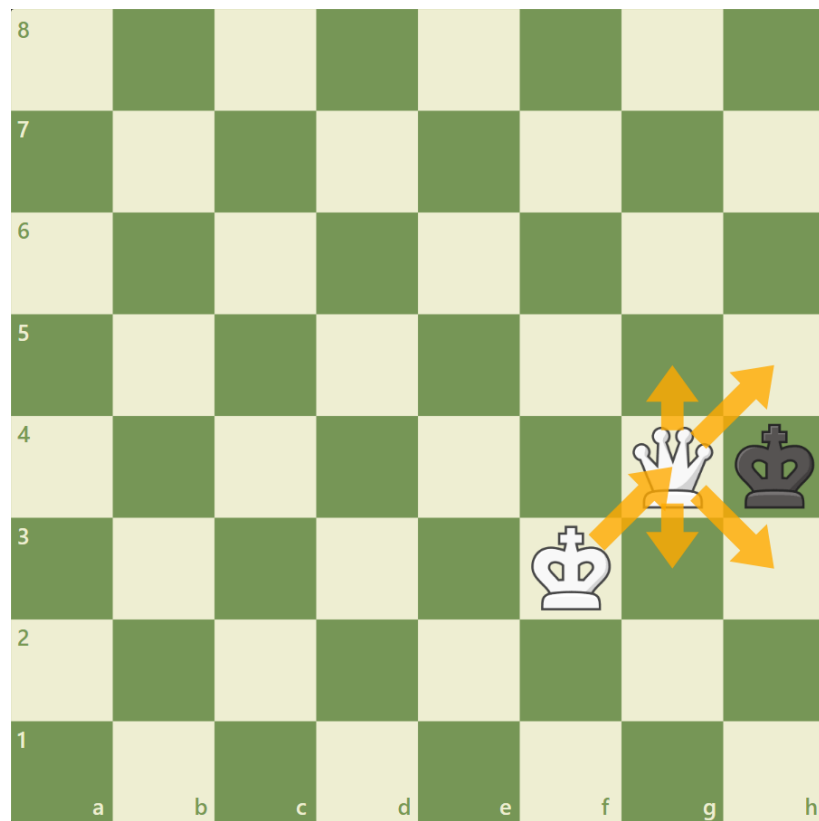
Pionek 1/2



Pionek 2/2



Mat



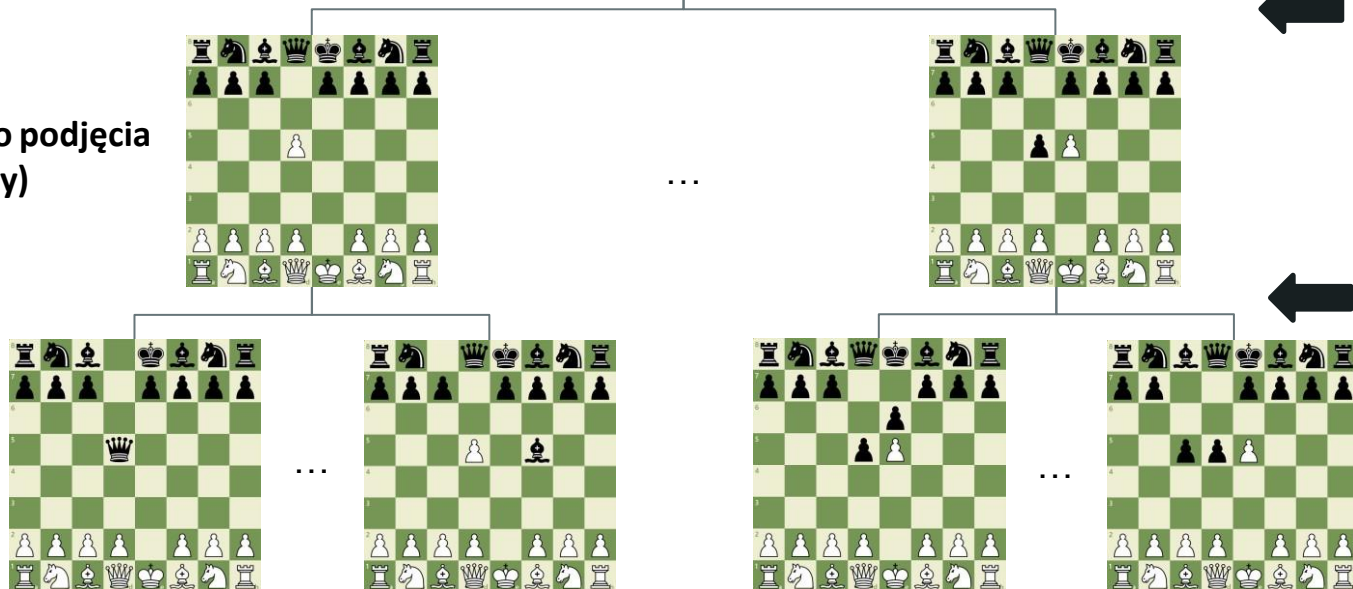
Złożoność gry w szachy

- Liczba możliwych gier ok. 10^{120} – liczba Shannona
- Czynniki rozgałęzienia ok. 35

Obecny stan szachownicy



Możliwe do podjęcia
akcje (ruchy)



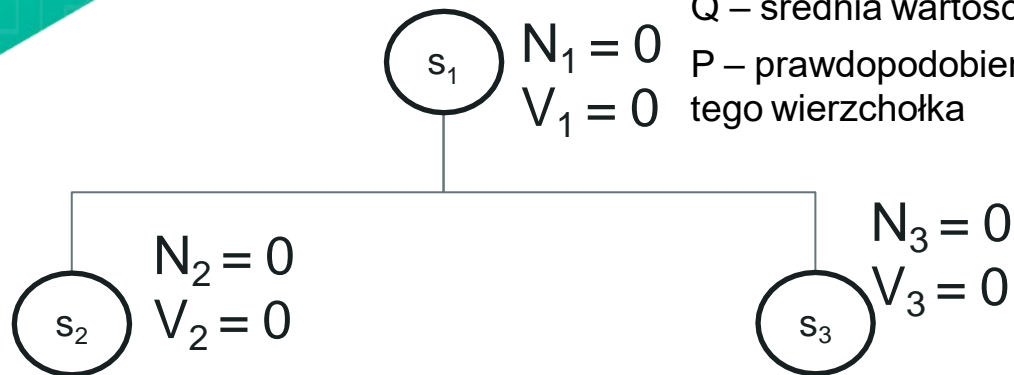
← 35

← $35^2 = 1\,225$

Monte-Carlo Tree Search

- Wybór (ang. *selection*)
- Rozrost (ang. *expansion*)
- Symulacja (ang. *playout*)
- Propagacja wstecz (ang. *backpropagation*)

Wybór



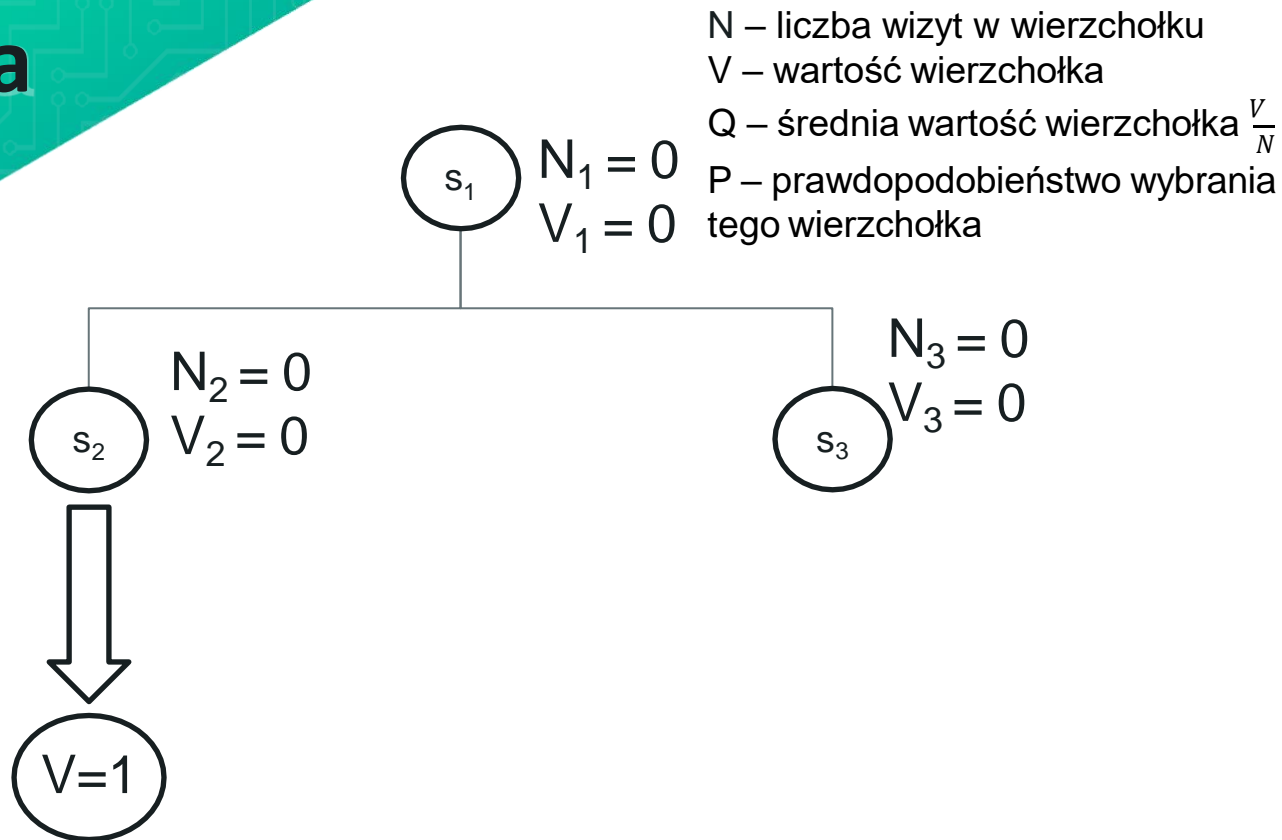
N – liczba wizyt w wierzchołku

V – wartość wierzchołka

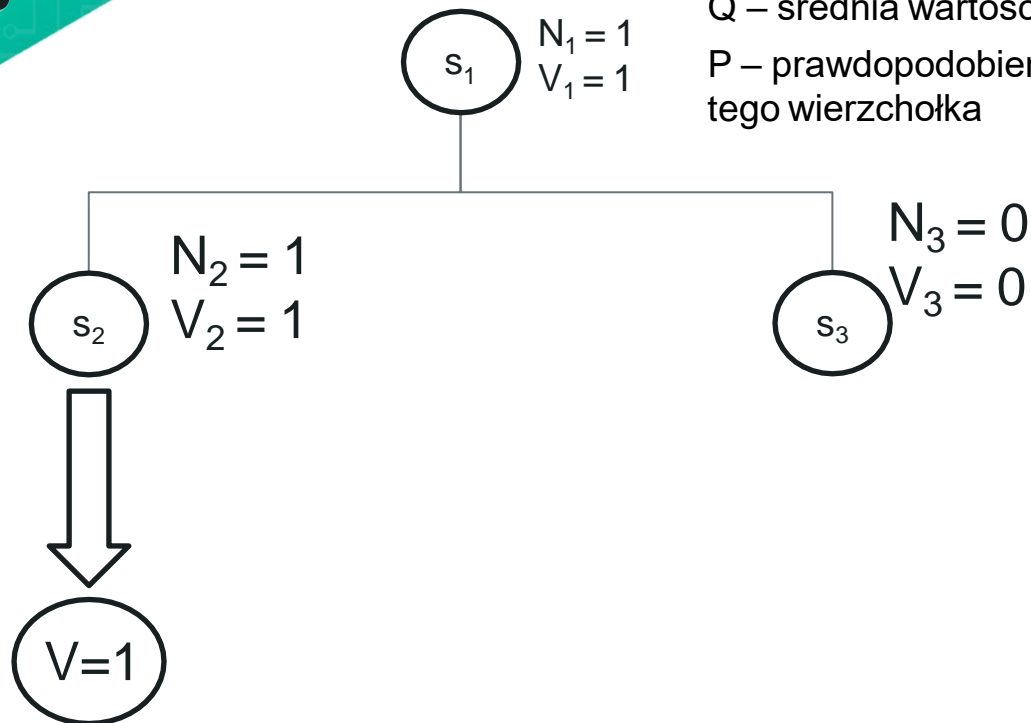
Q – średnia wartość wierzchołka $\frac{V}{N}$

P – prawdopodobieństwo wybrania tego wierzchołka

Symulacja



Propagacja wstecz



N – liczba wizyt w wierzchołku

V – wartość wierzchołka

Q – średnia wartość wierzchołka $\frac{V}{N}$

P – prawdopodobieństwo wybrania tego wierzchołka

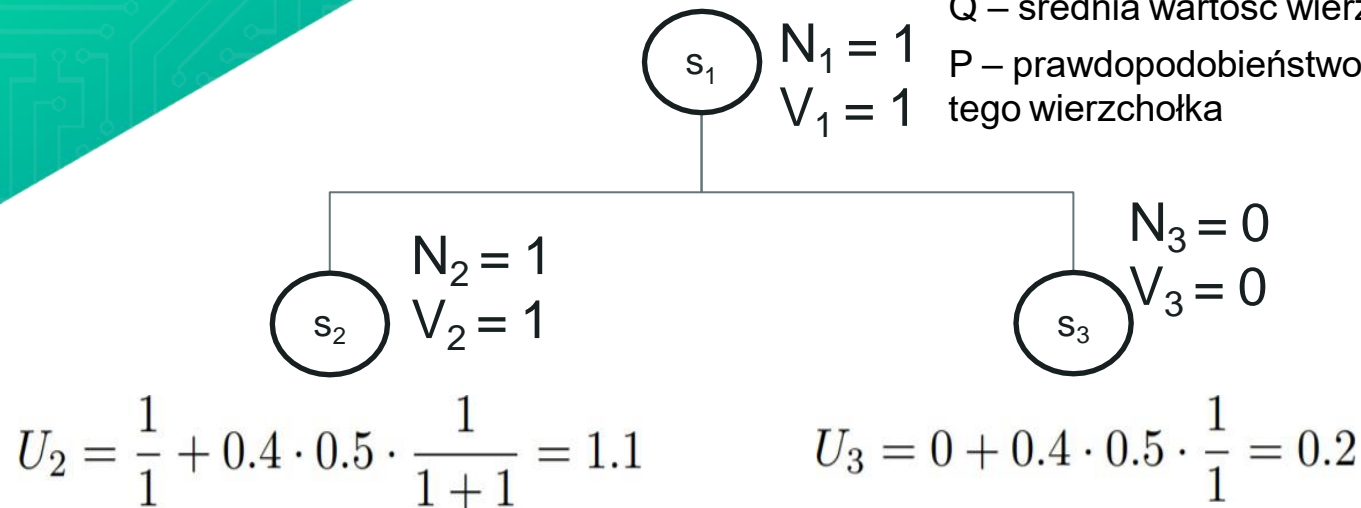
Wybór

N – liczba wizyt w wierzchołku

V – wartość wierzchołka

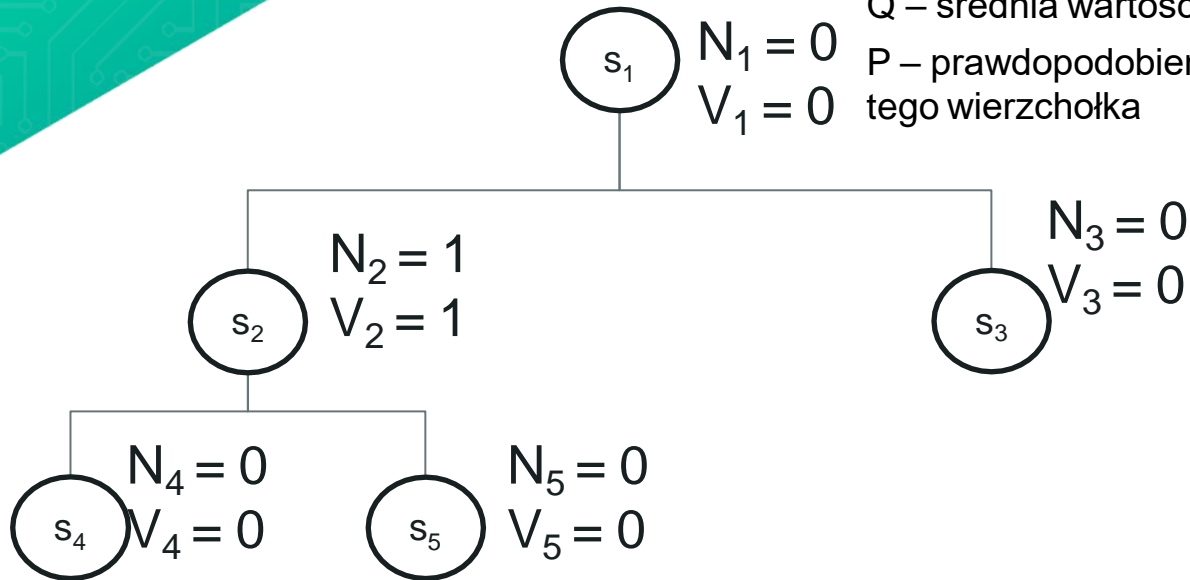
Q – średnia wartość wierzchołka $\frac{V}{N}$

P – prawdopodobieństwo wybrania tego wierzchołka



$$U = \frac{V}{N} + cP(s, a) \frac{\sqrt{\sum_b N(s, b)}}{1 + N(s, a)}$$

Rozrost



N – liczba wizyt w wierzchołku

V – wartość wierzchołka

Q – średnia wartość wierzchołka $\frac{V}{N}$

P – prawdopodobieństwo wybrania tego wierzchołka

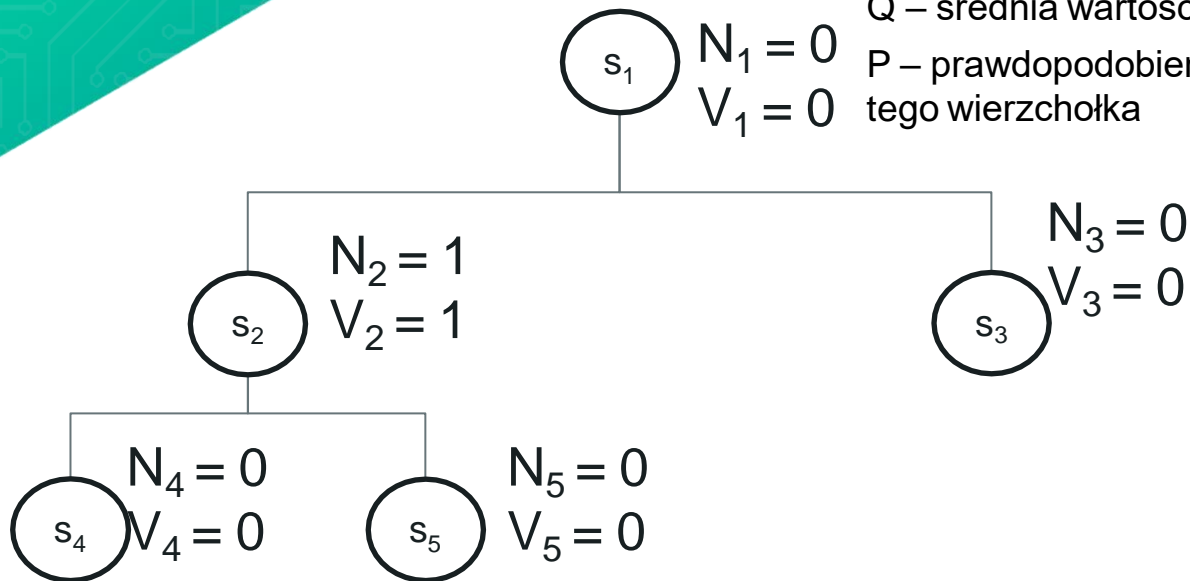
Wybór

N – liczba wizyt w wierzchołku

V – wartość wierzchołka

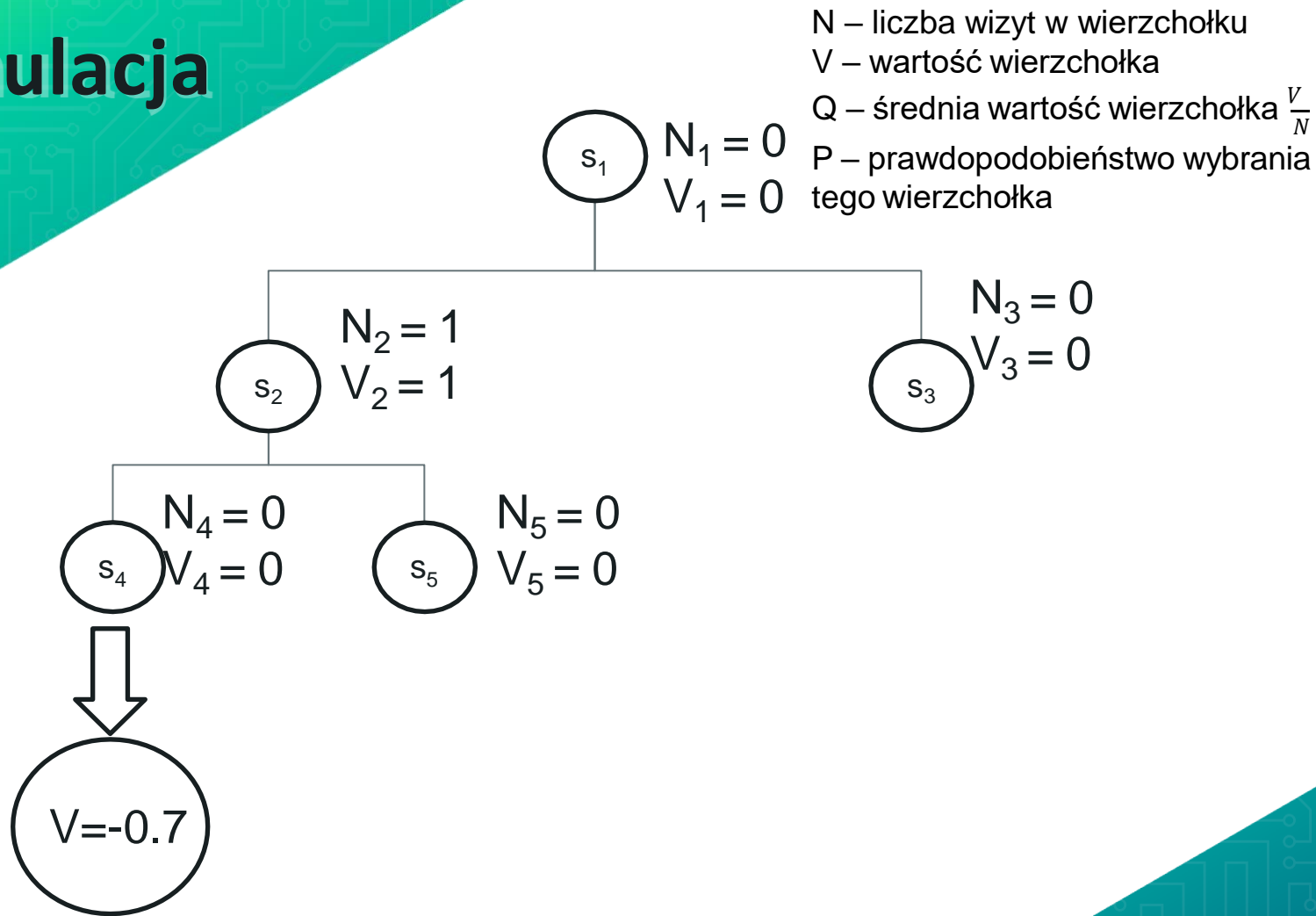
Q – średnia wartość wierzchołka $\frac{V}{N}$

P – prawdopodobieństwo wybrania tego wierzchołka

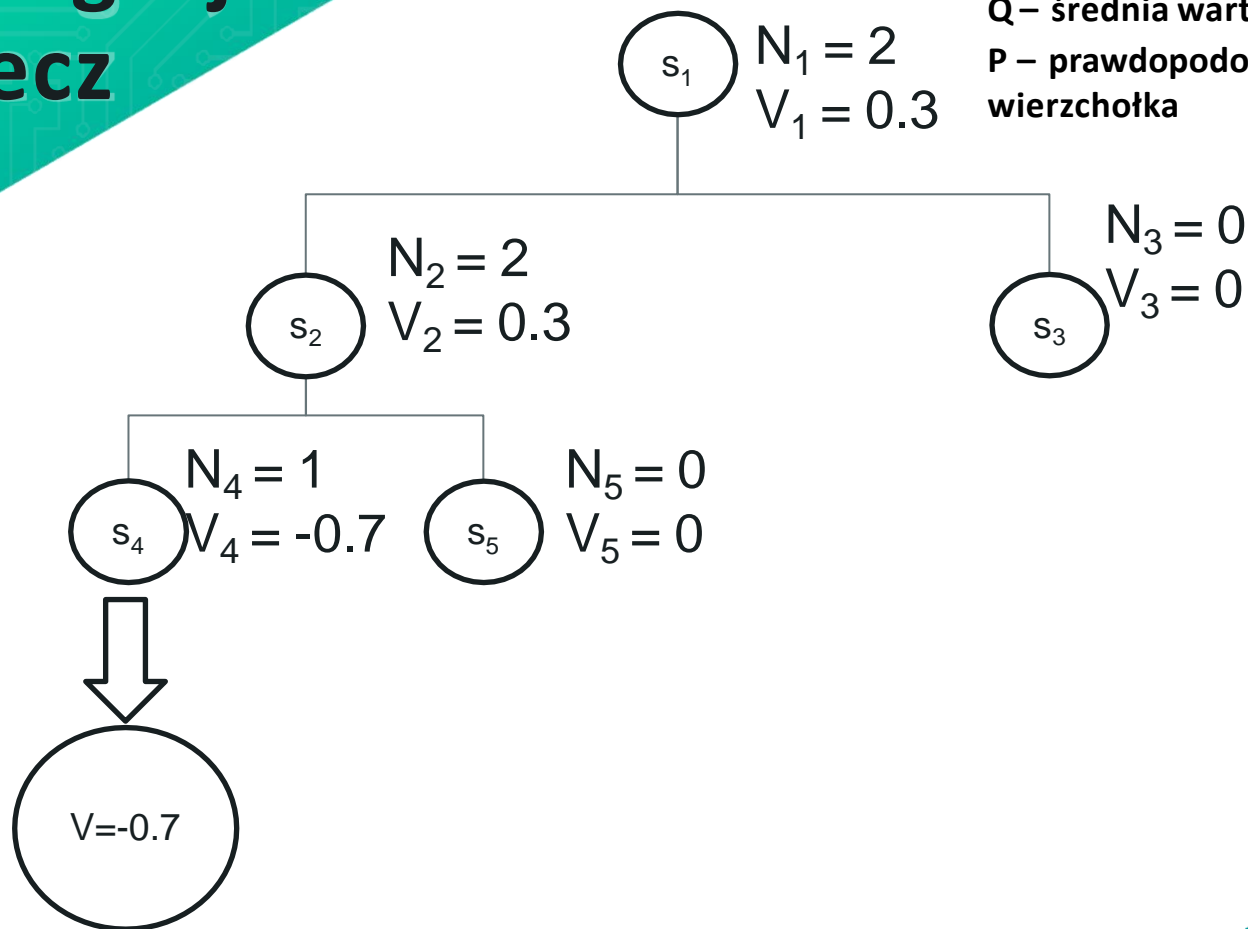


$$U_4 = U_5$$

Symulacja



Propagacja wstecz



N – liczba wizyt w wierzchołku

V – wartość wierzchołka

Q – średnia wartość wierzchołka $\frac{V}{N}$

P – prawdopodobieństwo wybrania tego wierzchołka

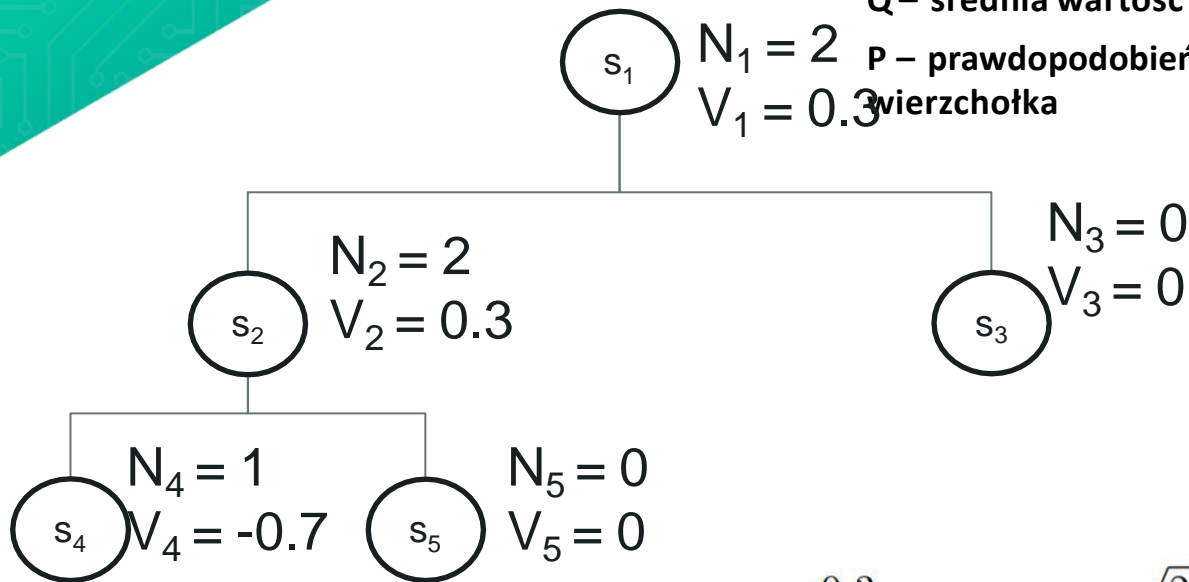
Wybór

N – liczba wizyt w wierzchołku

V – wartość wierzchołka

Q – średnia wartość wierzchołka $\frac{V}{N}$

P – prawdopodobieństwo wybrania tego wierzchołka

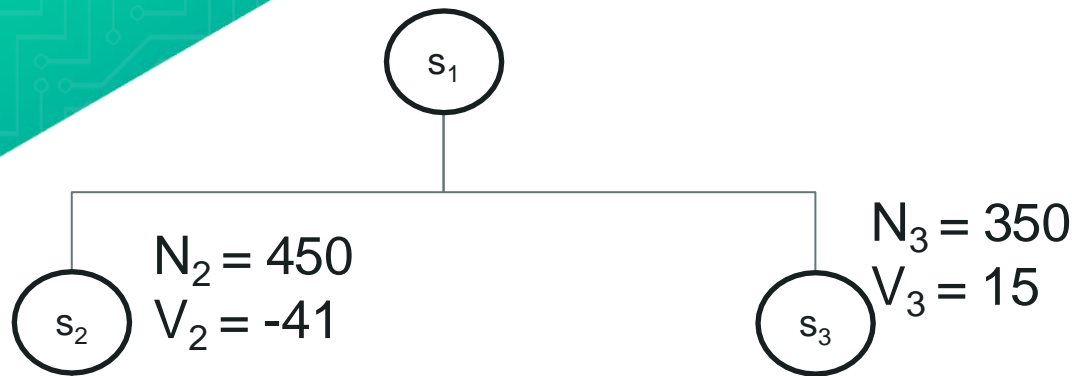


$$U = \frac{V}{N} + cP(s, a) \frac{\sqrt{\sum_b N(s, b)}}{1 + N(s, a)}$$

$$U_2 = \frac{0.3}{2} + 0.4 \cdot 0.5 \cdot \frac{\sqrt{2}}{1 + 2} = 0.24$$

$$U_3 = 0 + 0.4 \cdot 0.5 \cdot \frac{\sqrt{2}}{1 + 0} = 0.28$$

Wybór ruchu



- Podczas treningu ruch wybierany jest losowo z rozkładu:

$$\pi(a|s) = \frac{N(s, a)^{\frac{1}{\tau}}}{\sum_b N(s, b)^{\frac{1}{\tau}}} \quad \pi(s_2|s_1) = \frac{450}{800} = 0.5625$$
$$\pi(s_3|s_1) = \frac{350}{800} = 0.4375$$

- Podczas gry turniejowej wybrany jest ruch z największym N.

Temperatura

- Przez pierwsze 30 ruchów $\tau = 1$ (zachęcenie do eksploracji)

$$\pi(s_2|s_1) = \frac{450}{800} = 0.5625 \quad \pi(s_3|s_1) = \frac{350}{800} = 0.4375$$

- Później $\tau \rightarrow 0$ (wybieranie najlepszego ruchu zawsze)

```
1 x = np.array([450, 350])  
2 calculate_probabilities(x, temperature=1)
```

```
[0.56, 0.44]
```

```
1 calculate_probabilities(x, temperature=0.01)
```

```
[1.0, 0.0]
```

Prosty przykład

Dane wejściowe:

**Numeryczne wartości
dotyczące stanu pacjenta
np. ciśnienie krwi.**

$$x = [x_0, x_1, \dots, x_m]$$

Dane wyjściowe:

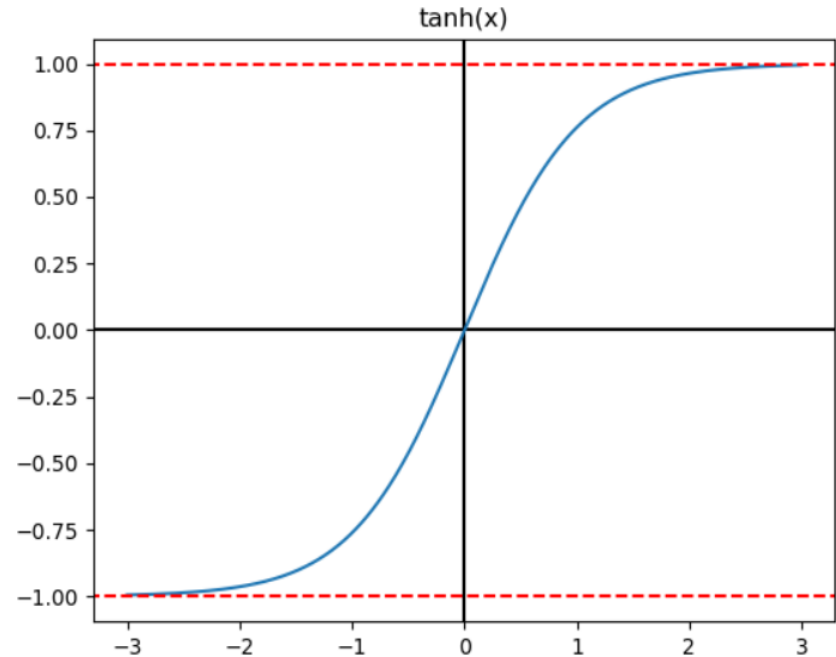
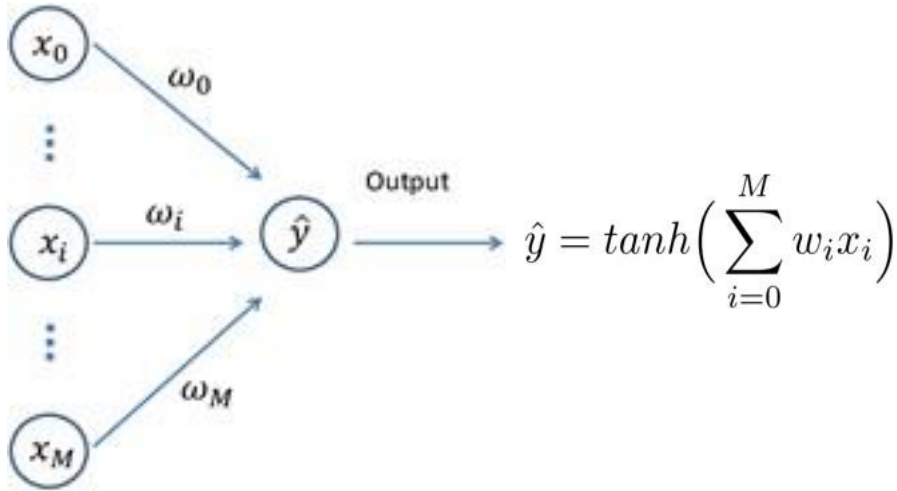
**Zmienna $\{-1, 1\}$ mówiąca czy
dany pacjent miał zawał w
przeciągu miesiącu od
badania.**

$$y = -1$$

Prosty przykład

Dane wejściowe: $x = [x_0, x_1, \dots, x_m]$

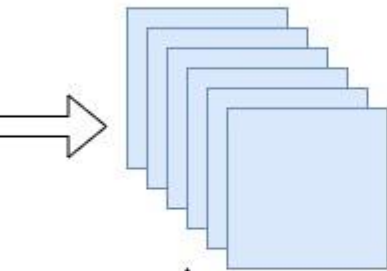
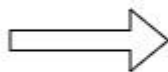
Dane wyjściowe: $y = -1$



Kot czy pies?

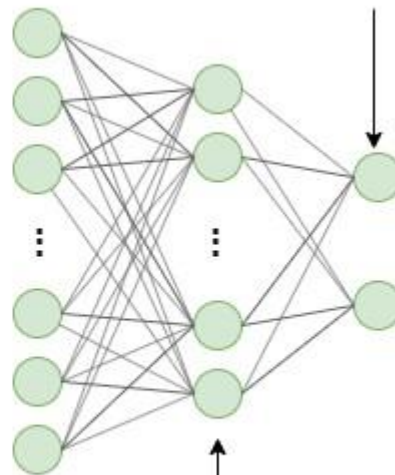
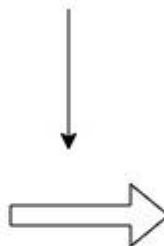


Dane wejściowe



Warstwa konwolucyjna

Spłaszczenie



Warstwy w pełni połączone

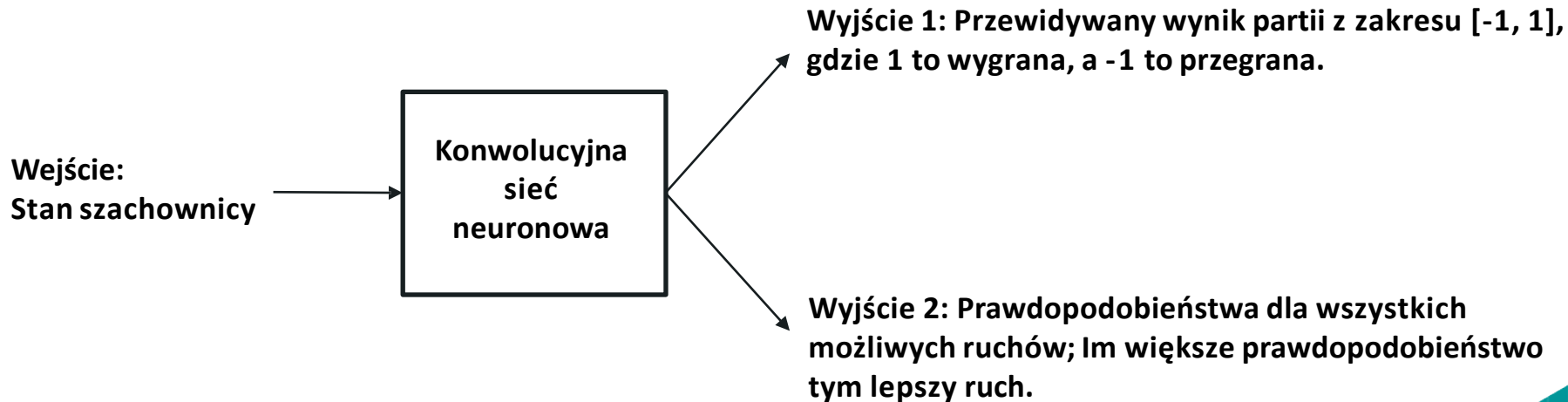
Prawdopodobieństwa
przynależności
do danej klasy obliczone
funkcją aktywacji softmax

Dane wyjściowe:
Kot zakodowany
jako [1, 0]

$$\begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

Zasada działania AlphaZero.

- Sieć neuronowa



Zasada działania AlphaZero, wejście

- Reprezentacja liczbowa stanu szachownicy



Pionki czarnego


$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Zasada działania AlphaZero, wejście 1

- Reprezentacja liczbowa stanu szachownicy

6 rodzajów bierek pierwszego gracza

6 rodzajów bierek drugiego gracza

2 powtórzenia

historia 8 stanów szachownicy

112 macierzy 8x8

Zasada działania AlphaZero, wejście 1

- Reprezentacja liczbowa stanu szachownicy

6 rodzajów bierek pierwszego gracza

6 rodzajów bierek drugiego gracza

2 powtórzenia

historia 8 stanów szachownicy

112 macierzy 8x8

1 Kolor

1 Licznik ruchów

2 Możliwość zrobienia roszady długiej i krótkiej przez pierwszego gracza

2 Możliwość zrobienia roszady długiej i krótkiej przez drugiego gracza

+1 Licznik ruchów bez postępu

119

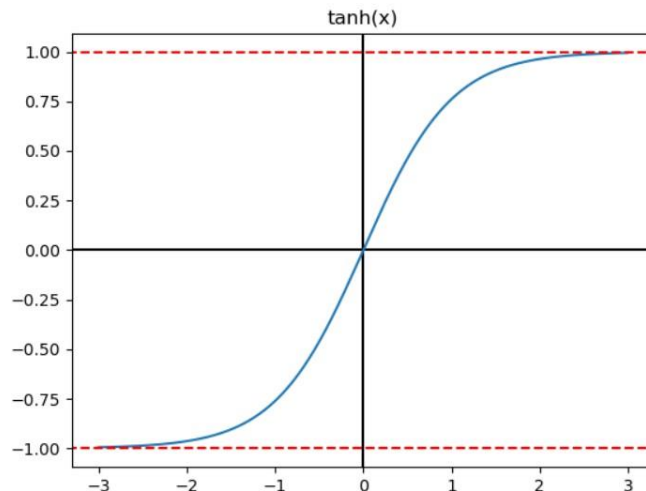
Wejście: macierz 8x8x119

Zasada działania AlphaZero, wyjście 1

- Prawdopodobieństwa dla wszystkich możliwych ruchów;
Im większe prawdopodobieństwo tym lepszy ruch.

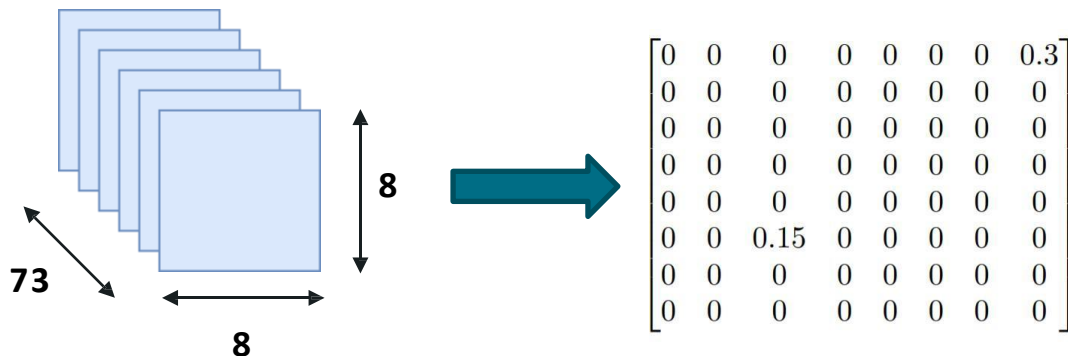
$$v \in [-1, 1]$$

$$MSE = \frac{1}{N} \sum_{j=1}^N (\hat{y}_j - y_j)^2$$



Zasada działania AlphaZero, wyjście 2

- Wyjście 2: Prawdopodobieństwa dla wszystkich możliwych ruchów; Im większe prawdopodobieństwo tym lepszy ruch.



- 56 ruchów hetmana:
 - 8 kierunków: N, W, E, S, NE, NW, SW, SE
 - odległość od 1 do 7
- 8 ruchów skoczka
- 9 słabych promocji
 - 3 figury
 - 3 sposoby na wejścia na ostatnie pole

Nielegalne ruchy są filtrowane, a pozostałe prawdopodobieństwa normowane do 1

Zasada działania AlphaZero, funkcja straty

$$l = (z - v)^2 - \boldsymbol{\pi}^\top \log \mathbf{p} + c ||\boldsymbol{\theta}||^2$$



Błąd średniokwadratowy



Entropia krzyżowa



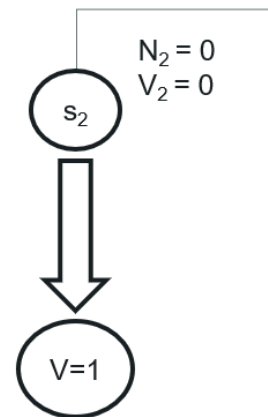
Regularyzacja L2


Użycie danych wyjściowych sieci neuronowej

Konwolucyjna
sieć
neuronowa

Wyjście 1: Przewidywany wynik partii z zakresu $[-1, 1]$,
gdzie 1 to wygrana, a -1 to przegrana.
Wartość zastąpiła wartość uzyskiwaną z symulacji partii
losowymi ruchami.

Wyjście 2: Prawdopodobieństwa dla wszystkich
możliwych ruchów; Im większe prawdopodobieństwo
tym lepszy ruch. Za pomocą tych wartości oblicza się wielkość według,
której wybiera się ruch badany w następnej iteracji.

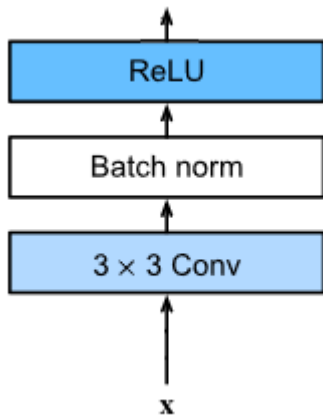



$$U = \frac{V}{N} + cP(s, a) \frac{\sqrt{\sum_b N(s, b)}}{1 + N(s, a)}$$

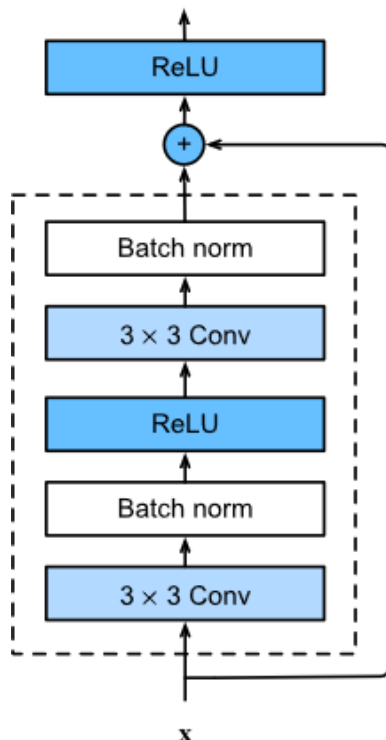
Dodatkowe zachęcenie do eksploracji

$$P(s, a) = P(s, a)(1 - \epsilon) + \epsilon \text{Dir}(\alpha = \frac{1}{35} \approx 0.3)$$
$$\epsilon = 0.25$$

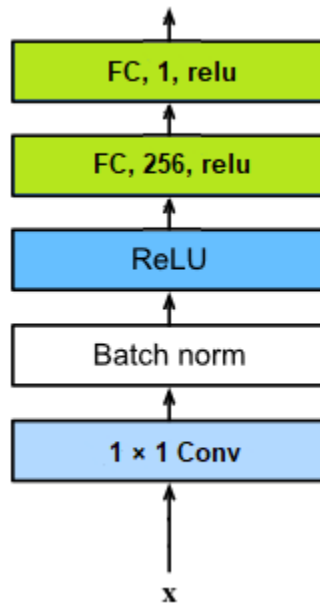
Architektura sieci neuronowej



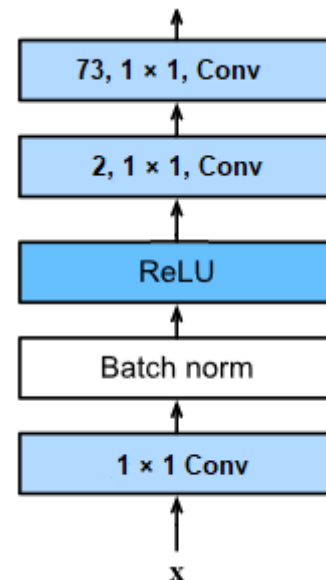
19 bloków:



przewidywanie
wyniku:



przewidywanie
ruchu:



Źródło danych wyjściowych sieci neuronowej

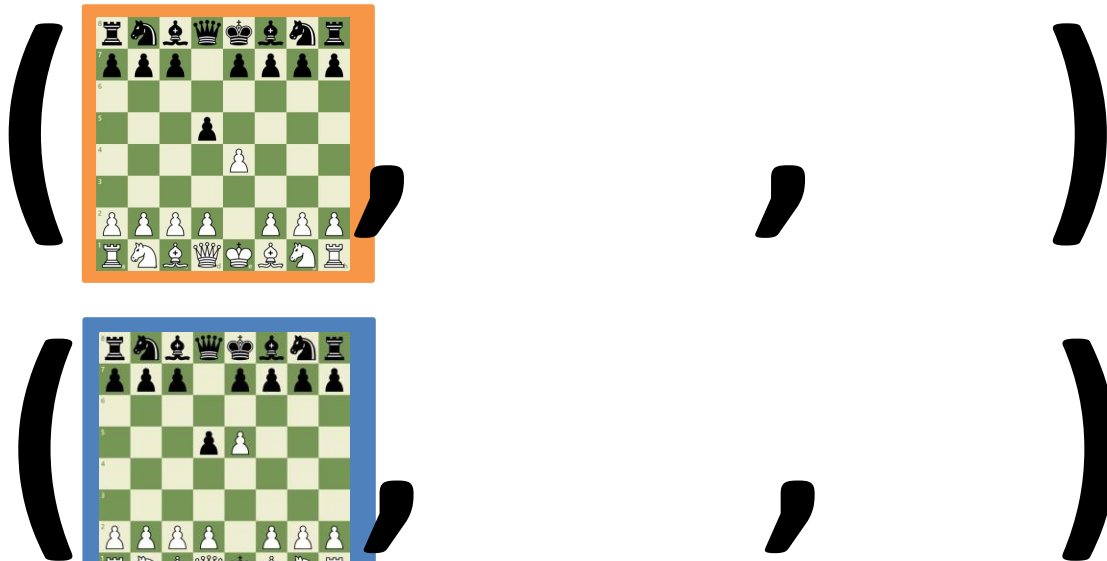
Ruch A



Ruch B



Ruch A



Źródło danych wyjściowych sieci neuronowej

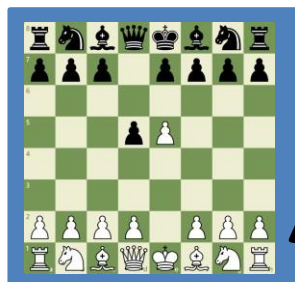
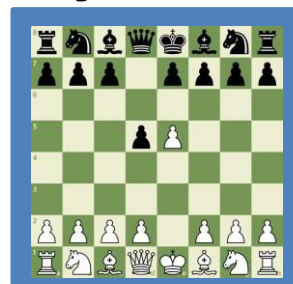
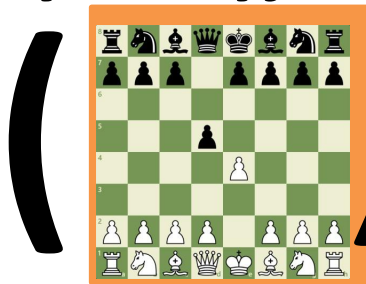
Ruch A



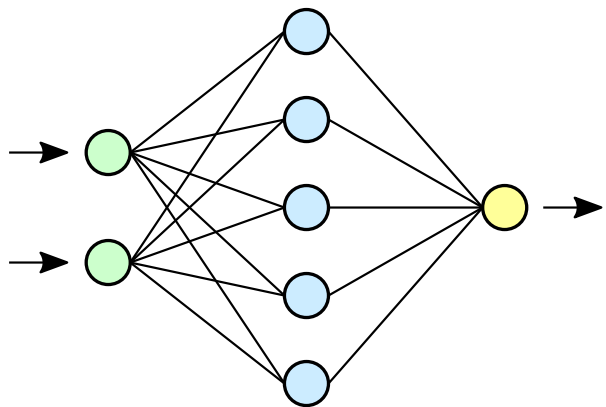
Ruch B



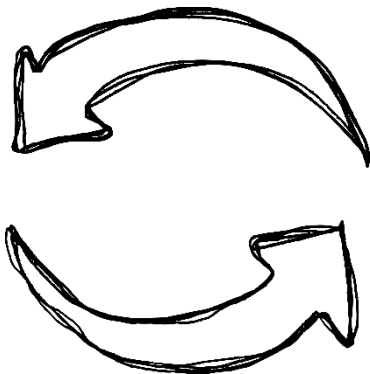
Ruch A



Schemat treningu w pigułce

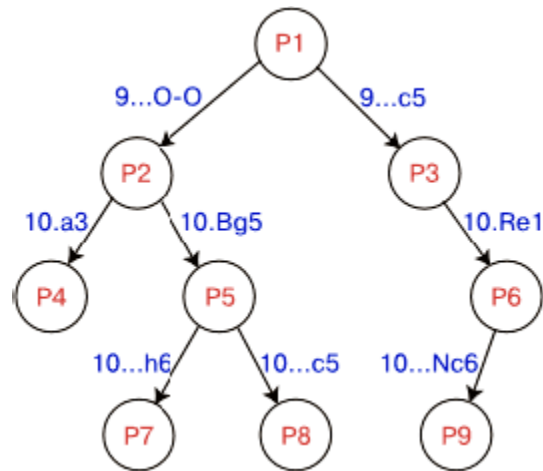


Dane do nauki



V - Ocena pozycji

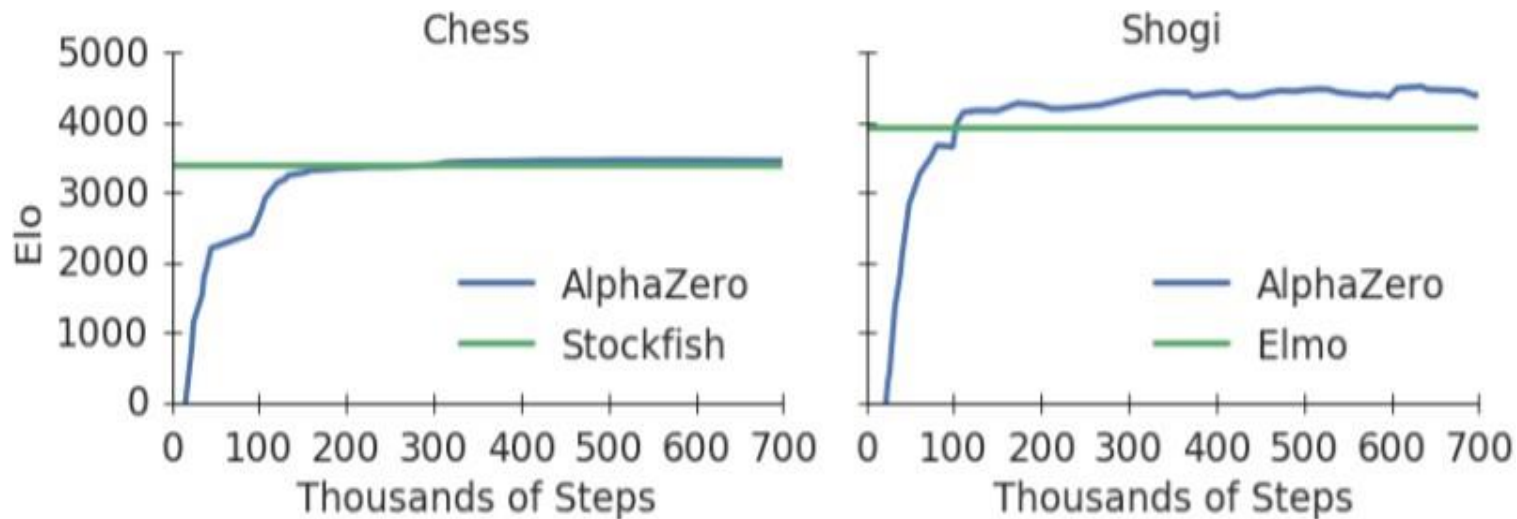
P(s, a) – prawdopodobieństwa
najlepszych ruchów



Wyniki

Game	White	Black	Win	Draw	Loss
Chess	<i>AlphaZero</i>	<i>Stockfish</i>	25	25	0
	<i>Stockfish</i>	<i>AlphaZero</i>	3	47	0
Shogi	<i>AlphaZero</i>	<i>Elmo</i>	43	2	5
	<i>Elmo</i>	<i>AlphaZero</i>	47	0	3
Go	<i>AlphaZero</i>	<i>AG0 3-day</i>	31	–	19
	<i>AG0 3-day</i>	<i>AlphaZero</i>	29	–	21

Wyniki



Wyniki

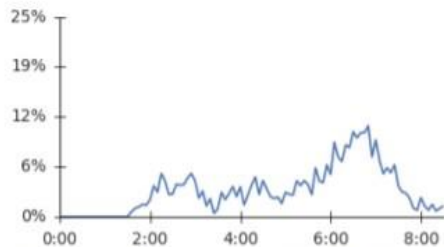
Program	Chess	Shogi	Go
<i>AlphaZero</i>	80k	40k	16k
<i>Stockfish</i>	70,000k		
<i>Elmo</i>		35,000k	

Table S4: Evaluation speed (positions/second) of *AlphaZero*, *Stockfish*, and *Elmo* in chess, shogi and Go.

Ciekawostka: debiuty grane przez AlphaZero



w 27/22/1, b 6/44/0

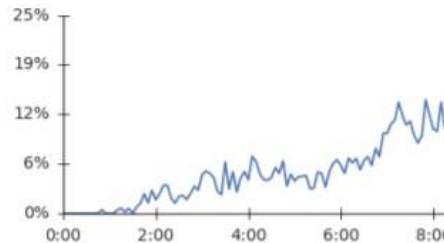


4. ♖a4 ♙e7 O-O ♜f6 ♚e1 b5 ♙b3 O-O

D06: Queens Gambit



w 16/34/0, b 1/47/2



2...c6 ♜c3 ♜f6 ♜f3 a6 g3 c4 a4

AlphaZero – najlepszy silnik szachowy



Krystian Kurek
Wydział MiNI
PW