5. prednáška

Učenie s odmenom a trestom a emergencia stratégie hry

Historický úvod

E. L. Thorndike (1887-1949)

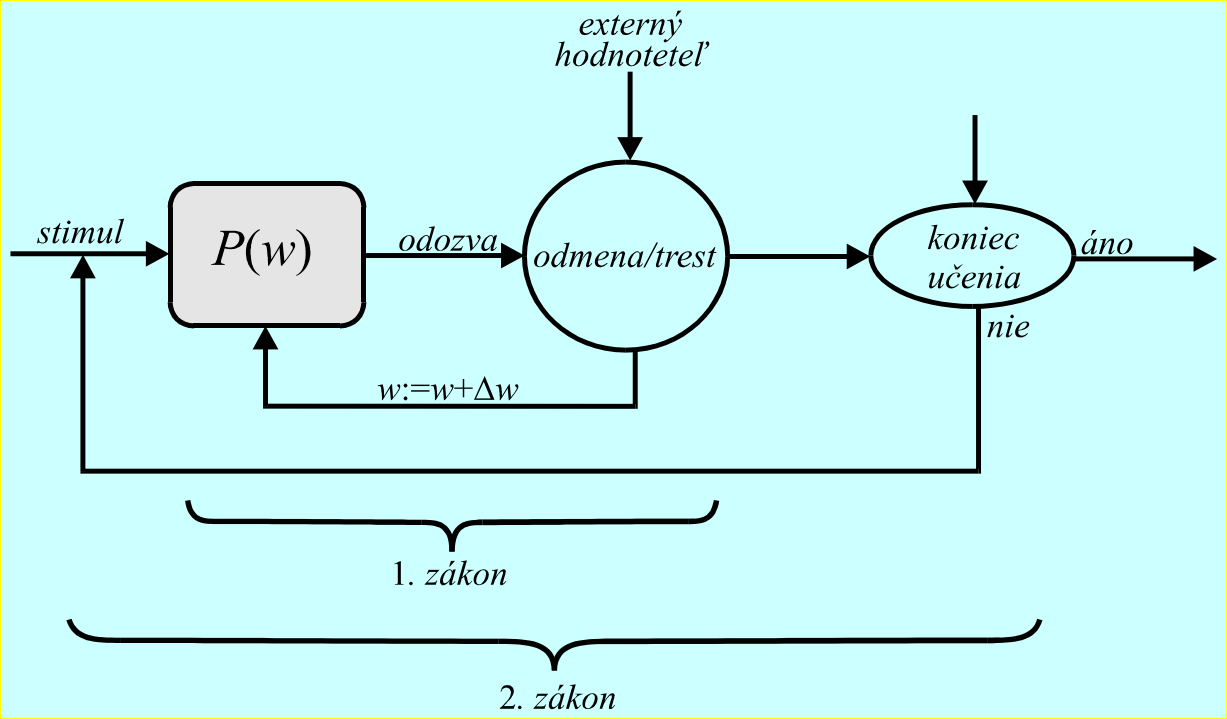
Americký psychológ Thorndike (1887-1949) vo svojej knihe „*The Fundamentals of Learning*“ zaviedol dva zákony:

1. Zákon účinku:
Ak odozva na opakujúci sa stimul je kladná (odmena), potom väzba medzi stimulom a odozvou sa postupne zosilňuje. V opačnom prípade, ak odozva je záporná (trest), potom väzba medzi stimulom a odozvou postupne zaniká.

2. Zákon opakovaného používania:
Požadované správanie je výsledkom častého používania dvojica stimul a odozva

Metóda učenia s odmenou a trestom

Učenie, ktoré je založené na týchto dvoch zákonoch sa nazýva „**učenie s odmenou a trestom**“ („**reinforcement learning**“). Tvorí teoretický základ behavioristického prístupu k učeniu.



Literatúra

(1) R. S. Sutton and A. G. Barto: *Reinforcement Learning:An Introduction,* MIT Press, Cambridge, MA, 1998.

(2) R. S. Sutton: Learning to Predict by the Methods of Temporal Differences. *Machine Learning* **3** (1988), 9-44.

(3) V. Kvasnička: Emergencia stratégie hry. AT&P Journal **12** (1999), 44-46.

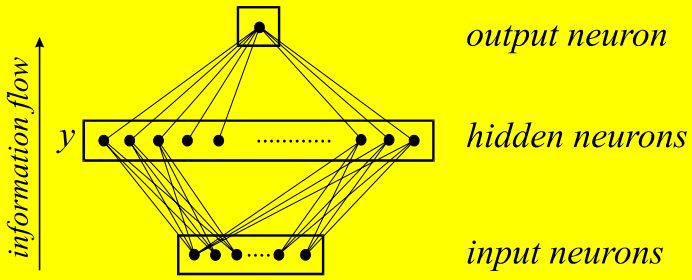
Tieto citácie a mnoho iných je dostupných na internetovskej adrese

`ftp://math.chtf.stuba.sk/pub/vlado/RL_textbook`

Učenia s odmenou a trestom

- Agent, ktorý je predmetom učenia, musí mať **kognitívny orgán, ktorý vykazuje určitú plasticitu**.

- Kognitívny orgán je **modelovaný jednoduchou doprednou neurónovou sieťou** s jednou vrstvou skrytých neurónov.



Poznámka: Plasticita neurónovej siete sa realizuje pomocou zmeny váhových koeficientov.

Teória

Študujme agentov, ktorý sú určený pomocou týchto dvoch množín:

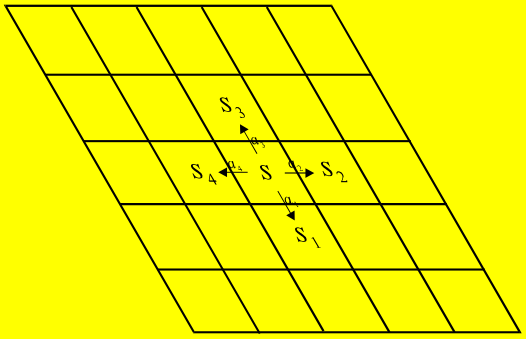
(1) Množina agentových **stavov**

$$\mathbb{S}=\{s_1, s_2, \ldots\}$$

(2) Množina agentových **akcií**

$$\mathbb{A}=\{a_1, a_2, \ldots\}$$

Akcie agentov sú interpretované ako funkcie, ktoré zobrazujú stavy agentov na seba, $s' = a(s)$.



Agent má **kognitívny orgán** (**prediktor**) pomocou ktorého ohodnocuje svoje stavy reálným číslom (**predikcia**)

$$P(w): \mathsf{S} \rightarrow R$$

alebo explicitne

$$z_i = P(s_i; w)$$

Zobrazenie P – prediktor je parametrická funkcia. Hovoríme, že vykazuje **plasticitu** vzhľadom k parametrom w.

Predpoklad: Výber určitej akcie $a \in \mathtt{A}$, ktorá je aplikovaná na stav $s \in \mathtt{S}$ je riadený kognitívnym orgánom.

Postup učenia

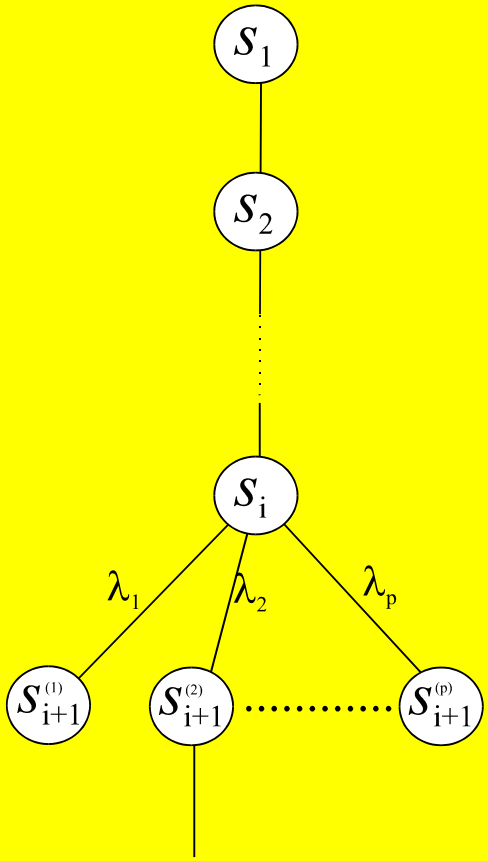
Majme sekvenciu stavov agenta a jej ohodnotenie (pochvala - trest)

$$s_1, s_2, \ldots, s_m, z$$

kde z je ohodnotenie, ktoré vyjadruje skutočnosť, či sekvencia má alebo nemá požadovanú vlastnosť

$$z = \begin{cases} 1 \ (\text{sekvencia má danú vlastnosť}) \\ 0 \ (\text{sekvencia nemá danu vlastnosť}) \end{cases}$$

Sekvencia stavov $s_1, s_2, \ldots, s_m$ je zostrojená **kvázináhodne**, t.j. podsekvencia $s_1, s_2, \ldots, s_i$ je rozšírená o ďalší stav s_{i+1} na základe predikcie $z_i = P(s_i; w)$.



$$s_{i+1} = arg\ \max_j P\left(s_j^{(i)}; w\right)$$

Stratégia učenia

Adaptovať kognitívny orgán $P(w)$ tak, že všetky predikcie

$$z_1 = P_1 = P(s_1; w), \ldots, z_m = P_m = P(s_m; w)$$

sú rovnaká, ako externé ohodnotenie celej sekvencie číslom $z = P_{m+1}$

Poznámka: Používame takú stratégiu učenia, že ak je výsledná sekvencia vyhodnotená ako úspešná (neúspešná), potom všetky stavy z tejto sekvencie sú tiež úspešné (neúspešné).

Adaptácia parametrov kognitívneho orgánu

Uvažujme sekvenciu stavov, ktorá je vyhodnotená ako celok číslom z, potom každý stav tejto sekvencie je tiež ohodnotený týmto číslom

$$(s_1, z), (s_2, z), \ldots, (s_m, z)$$

Kvalita tejto predikcie je určená pomocou účelovej funkcie

$$E(w) = \frac{1}{2} \sum_{t=1}^{m} \left(z - P(s_t; w) \right)^2$$

Účelová funkcia je nezáporná, rovná sa nule len vtedy, ak kognitívny orgán poskytuje také ohodnotenie všetkých stavov sekvencie, ktoré je totožné s externým ohodnotením celej sekvencie

Adaptačná metóda učenia s odmenou a trestom má tieto základné formule

$$w := w + \Delta w$$

$$\Delta w = \sum_{t=1}^{m} \Delta w_t$$

$$\Delta w_t = \alpha(P_{t+1} - P_t) \sum_{k=1}^{t} \lambda^{t-k} grad_w P_k$$

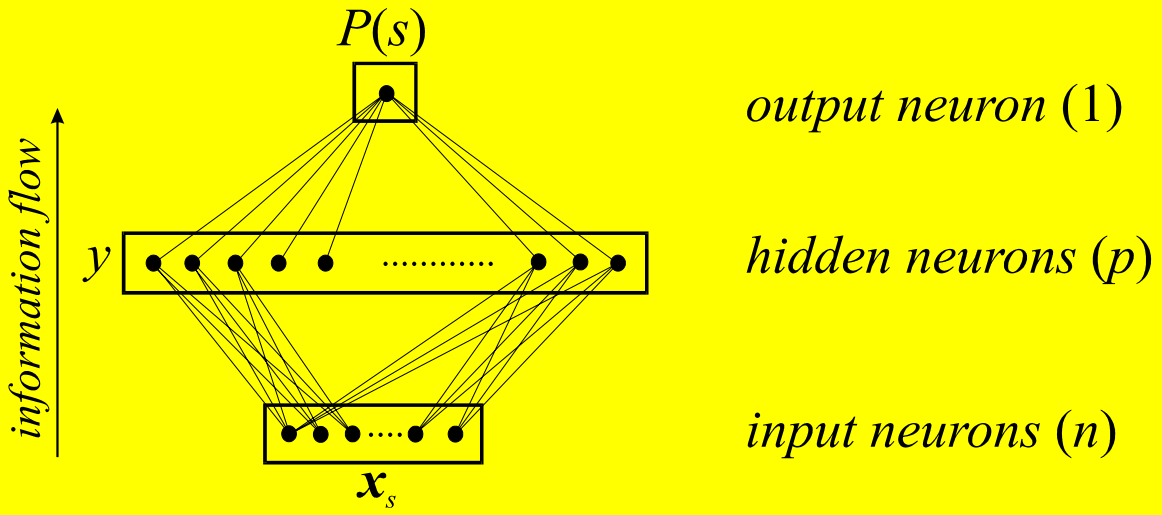
Poznámka. Tieto formule tvoria základ **metódy učenia s odmenou a trestom vo verzii časových rozdielov** *RL-TD*(λ) (**reinforcement learning - temporal differences**)

Architektúra neurónových sietí použitých ako kognitívny orgán

Ohodnotenie stavov je uskutočnené pomocou parametrickej funkcie

$$P(s)=P(\boldsymbol{x}_s;w)$$

Táto funkcia je uskutočnená pomocou doprednej neurónovej siete s jednou vrstvou skrytých neurónov



Algoritmus učenia

Step 1. *Váhové a prahové koeficienty kognitívneho orgánu sú náhodne vygenerované, t:=1.*

Step 2. *Ak kázináhodne vygenerovaná sekvencia stavov spĺňa požadovanú vlastnosť, potom je ohodnotená číslom z=1, v opačnom prípade je ohodnotená číslom z=0.*

Step 3. *Váhové a prahové koeficienty sú obnovené pomocou formúl RL-TD(λ) metódy,t:=t+1.*

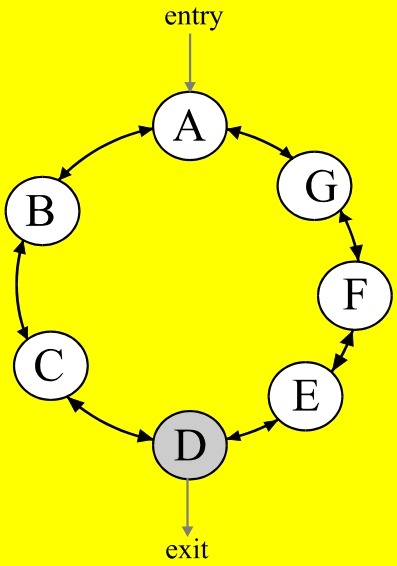
Step 4. *Ak sú splnené konvergenčné kritéria, potom prejdi na krok 5, v opačnom prípade pokračuj krokom 2.*

Step 5. *Stop.*

Prvý ilustratívny príklad

Pohyb v jednoduchom bludisku

Študujme jednoduché kruhové bludisko, ktoré obsahuje 7 "miestností", pričom miestnosť *A* je vstupná a miestnosť *D* je výstupná. V tomto bludisku sa pohybuje agent s kognitívnym orgánom, jeho cieľom je nájsť čo najkratšiu cestu z *A* do *D*. Pri generovaní tejto cesty používa svoj kognitívny orgán ako určitého radcu, ktorým smerom sa má pohybovať.



Príklady ciest v bludisku:

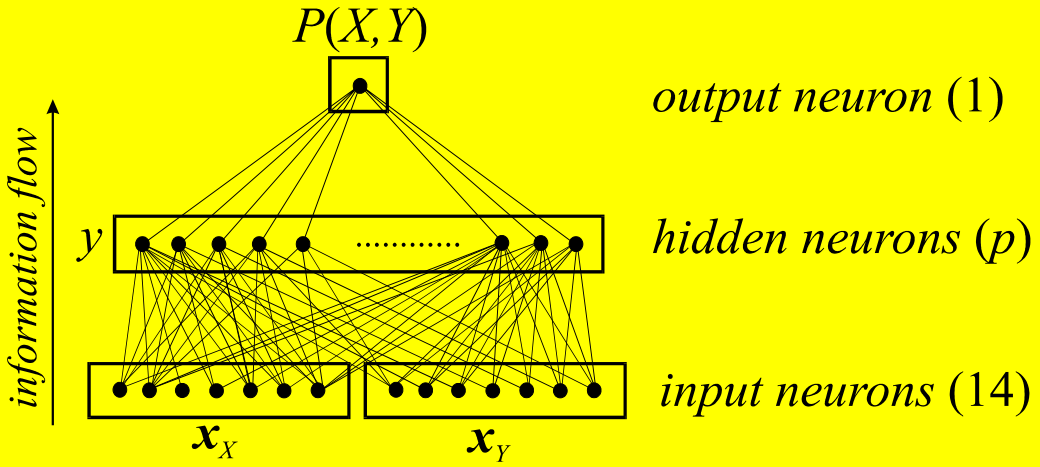
(1) $\mathcal{W}_1$=ABCBAGFED, | $\mathcal{W}_1$|=9

(2) $\mathcal{W}_2$=ABCD, | $\mathcal{W}_2$|=4 (najkratšia cesta)

Stavy sú reprezentované siedmimi binárnymi vektormi

#	state	binary vector ***x***
1	*A*	(1000000)
2	*B*	(0100000)
3	*C*	(0010000)
4	*D*	(0001000)
5	*E*	(0000100)
6	*F*	(0000010)
7	*G*	(0000001)

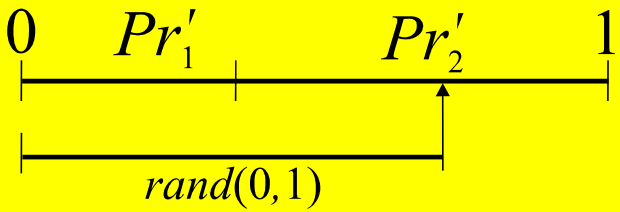
Každá orientovaná hrana (X,Y) je ohodnotená predikciou $P(X,Y)$, čo je realizované pomocou doprednej neurónovej siete so vstupnými aktivitami určenými vektormi $\boldsymbol{x}_X$ a $\boldsymbol{x}_Y$, ktoré sú priradené stavom X a Y.



Predikcia $P(X,Y)$ je interpretovaná ako pravdepodobnosť, že daná cesta $\mathcal{W}=A...UVX$ bude rozšírená na cestu $\mathcal{W}'=A...UVXY$

$$A...UVX \rightarrow \begin{cases} A...UVXV & (Pr_1=P(X,V)) \\ A...UVXY & (Pr_2=P(X,Y)) \end{cases}$$

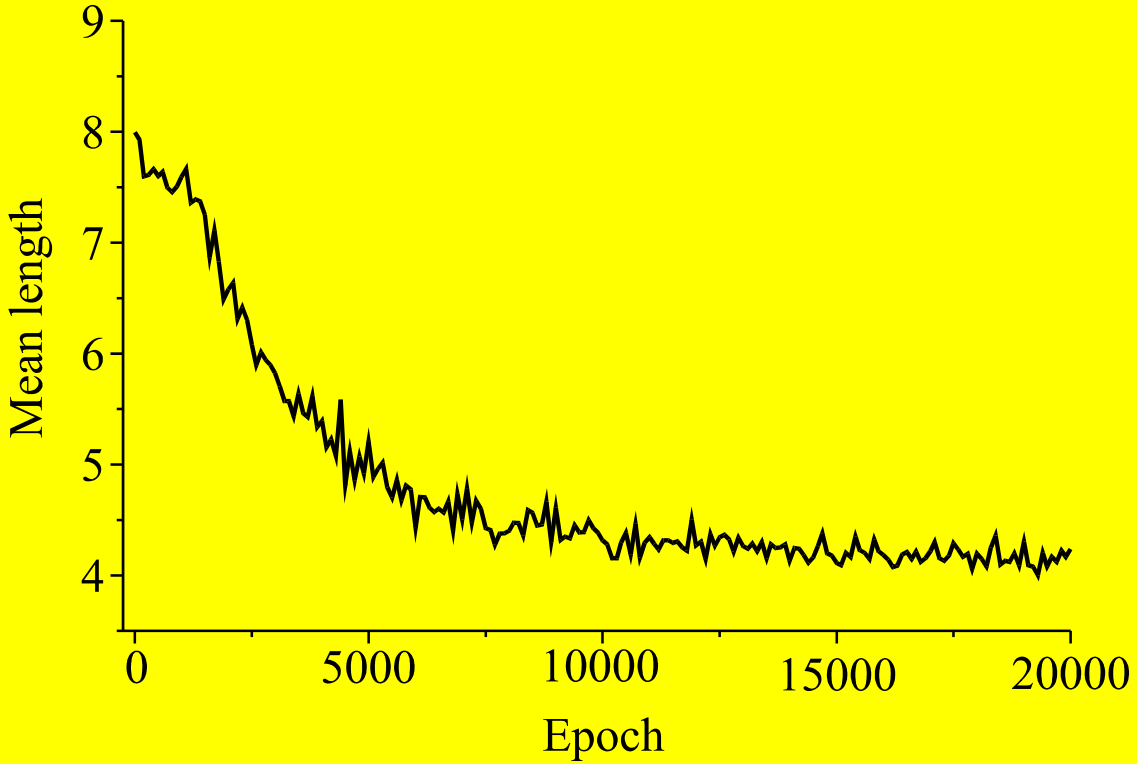
$$Pr_1' = \frac{Pr_1}{Pr_1+Pr_2}, \quad Pr_2' = \frac{Pr_2}{Pr_1+Pr_2}$$

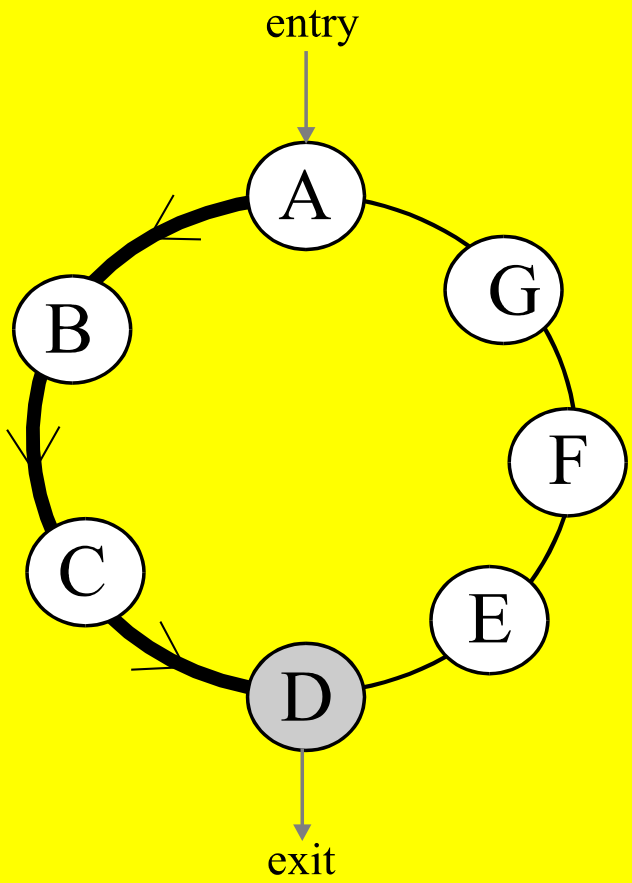


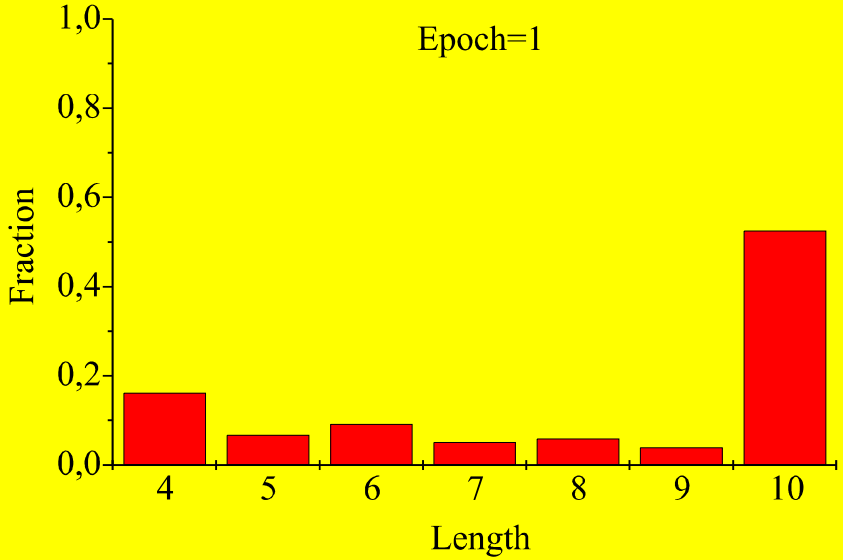
Parametre adaptačného procesu

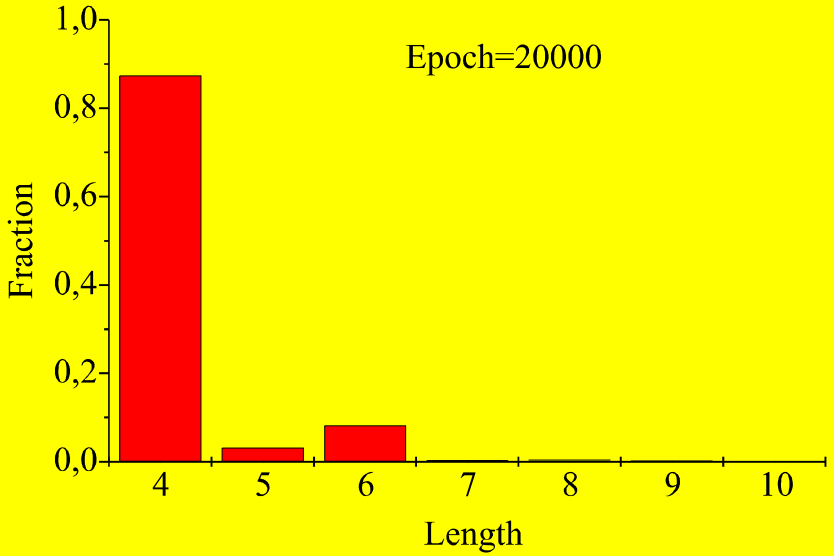
1. α=0.1 (rýchlosť učenia).
2. λ=0.9 (temporal-difference parameter).
3. p=5 (počet skrytých neurónov).
4. Počiatočné hodnoty váhových a prahových koeficientov sú náhodne vyberané z otvoreného intervalu (-2,2).
5. Proces učenia je zastavený po 20000 epochách.
6. Kvázináhodne generované cesty sú ohodnoco-vané podľa formule

$$z = \begin{cases} 1 & \left(ak\,|\mathcal{W}| = 4\right) \\ 0 & \left(ak\,|\mathcal{W}| > 4\right) \end{cases}$$


Mean length
9
8
7
6
5
4
0
5000
10000
15000
20000
Epoch

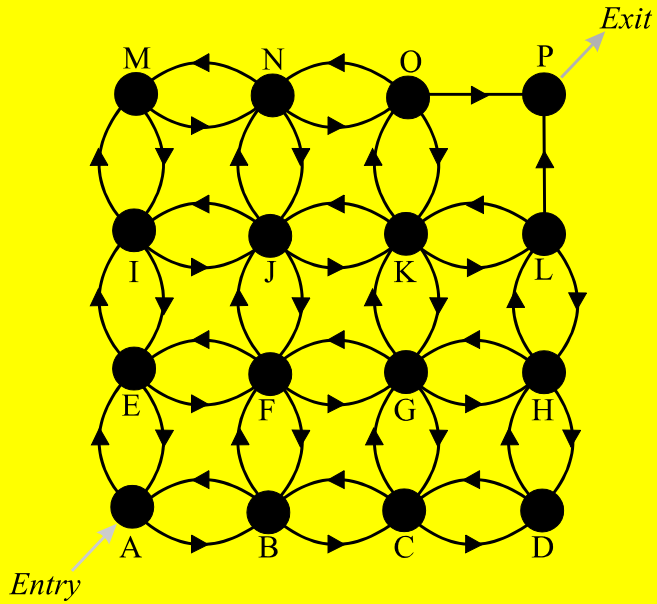

entry
A
G
B
F
C
E
D
exit


Epoch=1
Fraction
1,0
0,8
0,6
0,4
0,2
0,0
4
5
6
7
8
9
10
Length


Epoch=20000
Fraction
1,0
0,8
0,6
0,4
0,2
0,0
4
5
6
7
8
9
10
Length

Second illustrative example

Let us consider a slightly complex example than the previous one, it corresponds to a generator of bounded random walks composed of sixteen states *A, B, ...,O, P*.



Our goal is to construct walks $\mathtt{W}$ that

(1) start in the initial state *A* and end in the terminal state *P*,

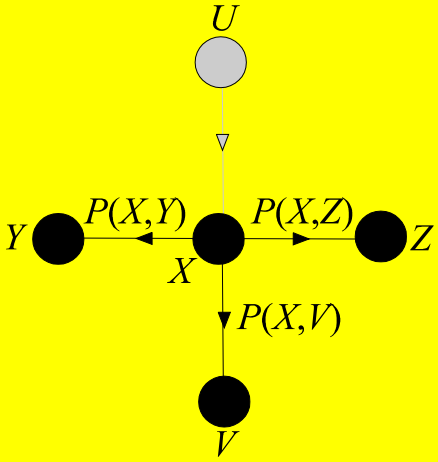
(2) are of shortest length, i.e. $|\mathtt{W}|=6$.

States are represented by fifteen 15-dimensional binary “unit” vectors

#	state	binary vector
1	A	(100000000000000)
2	B	(010000000000000)
3	C	(001000000000000)
4	D	(000100000000000)
5	E	(000010000000000)
6	F	(000001000000000)
7	G	(000000100000000)
8	H	(000000010000000)
9	I	(000000001000000)
10	J	(000000000100000)
11	K	(000000000010000)
12	L	(000000000001000)
13	M	(000000000000100)
14	N	(000000000000010)
15	O	(000000000000001)
16	P	

Each oriented edge (X,Y) is evaluated by a **predictor** $P(X,Y)$ with the following meaning:

Let us have a walk $\mathtt{W}=(A\ldots UX)$ terminated in the state X, and let the last state X has one to three forthcoming neighbor states denoted Y, Z, and V, respectively. The walk is extended by one of them with probability proportional predictors $P(X,Y)$, $P(X,Z)$, and $P(X,V)$.



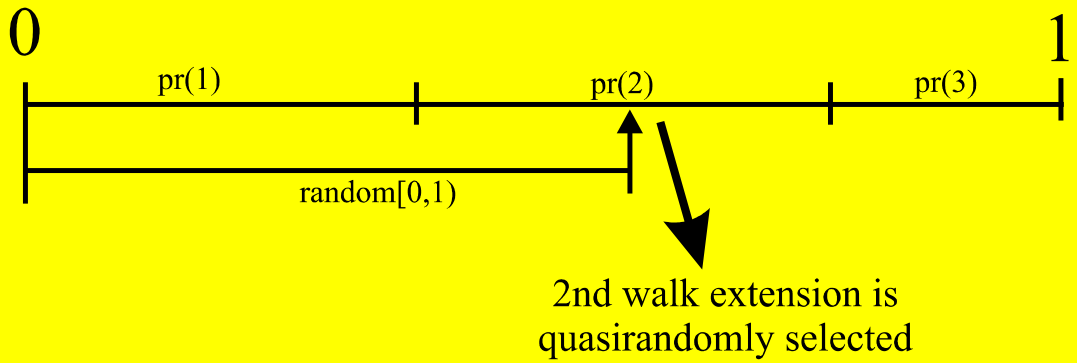
$$\mathcal{W} = A...UX \rightarrow \begin{cases} \mathcal{W}' = A...UXY\left(p_Y \approx P\left(X,Y\right)\right) \\ \mathcal{W}' = A...UXZ\left(p_Z \approx P\left(X,Z\right)\right) \\ \mathcal{W}' = A...UXV\left(p_V \approx P\left(X,V\right)\right) \end{cases}$$

This type of quasirandom selection is numerically realized by the **"roulette wheel"** (see Goldberg's implementation of GA)

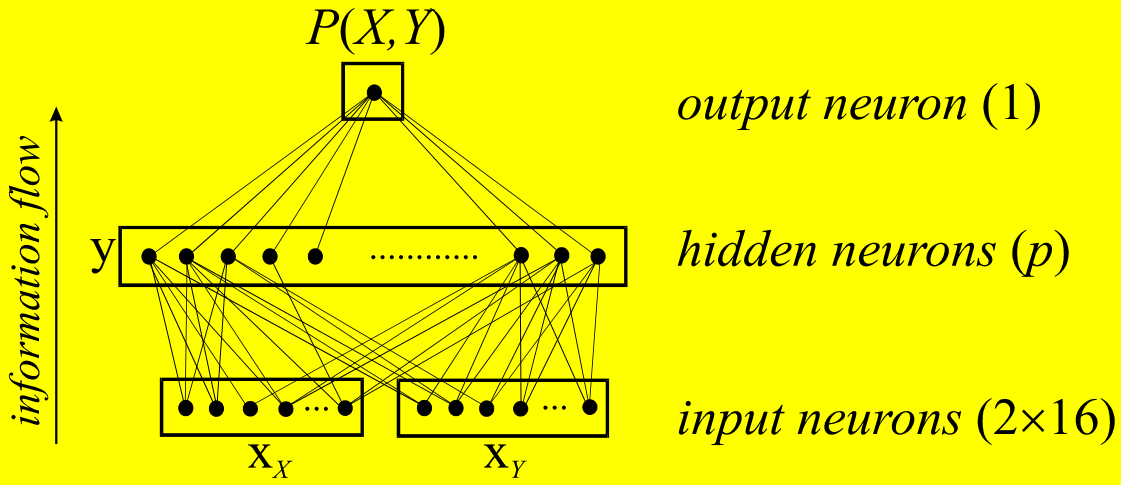
$$pr_1 = \frac{p_Y}{p_Y + p_Z + p_V}$$

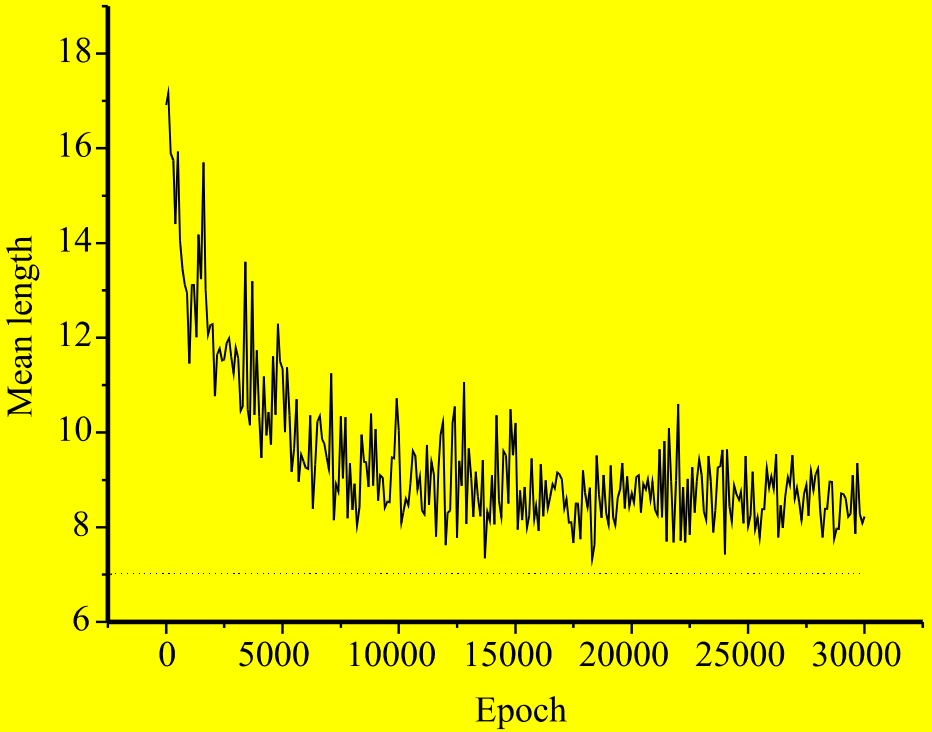
$$pr_2 = \frac{p_Z}{p_Y + p_Z + p_V}$$

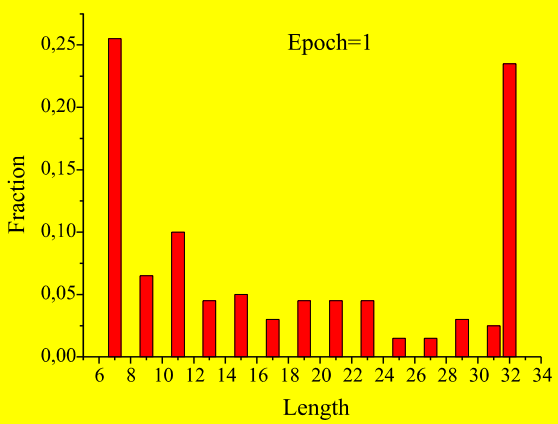
$$pr_3 = \frac{p_V}{p_Y + p_Z + p_V}$$

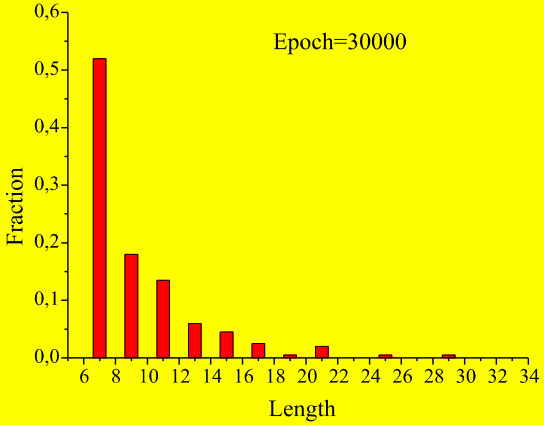


The predictor $P(X,Y)$ is numerically realized by the feed-forward neural network with input-neuron activities specified by the vector representation of states X.

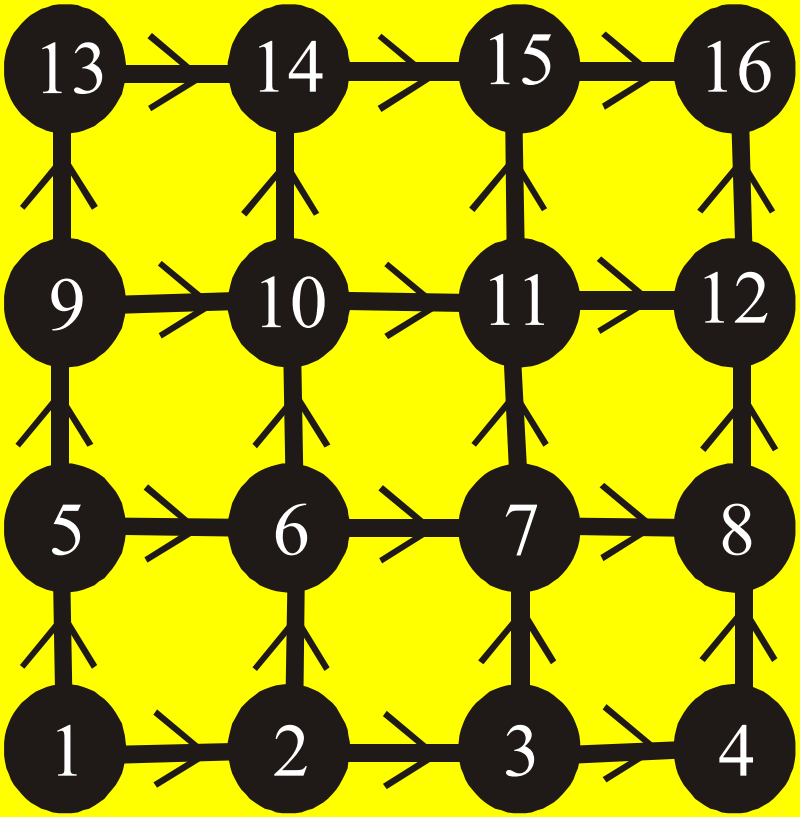








Priesvitka 30

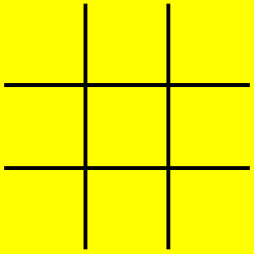


Priesvitka 31

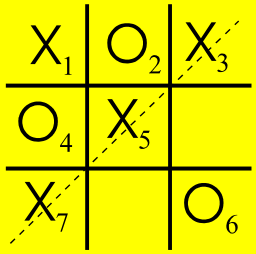
Tretí ilustratívny príklad

Hra piškvorky (tic-tac-toe)

The American Heritage Disctionary: *A game played by two people, each trying to make a line of three X's or three O's in a boxlike figure with nine spaces*.



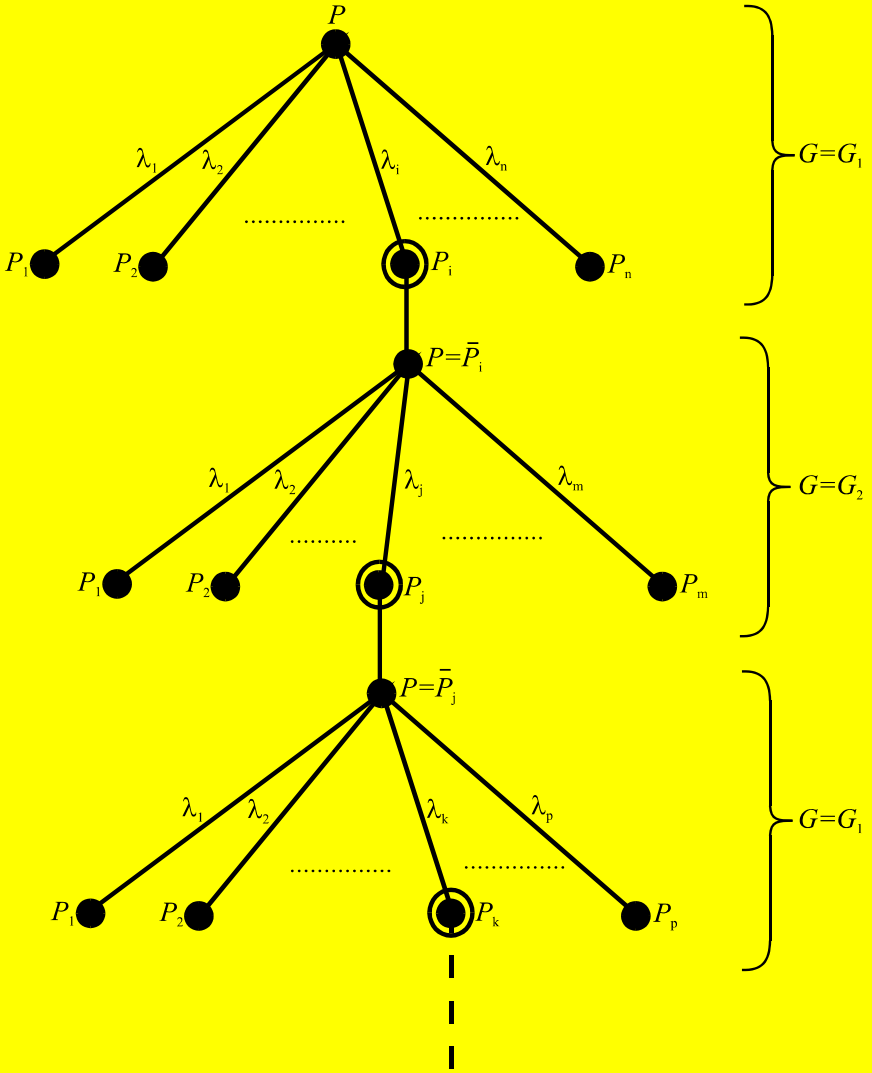
X-hráč (prvý)
O-hráč (druhý)



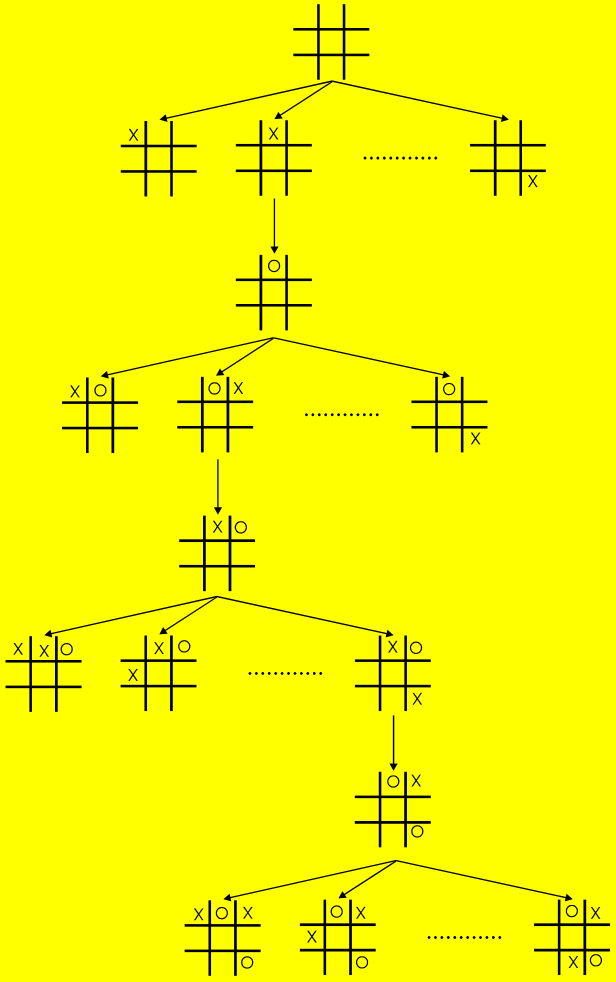
X-hráč zvíťazil

Hra je zahájená prvým hráčom (X), na jeho ťah odpovedá druhý hráč (O), toto striedanie hráčov sa opakuje až do konca hry. **Koniec hry** nastáva **víťaztvom** hráča, ktorý dosiahol „riadkovu“ pozíciu troch svojich znakov, alebo **remízou**, ak po deviatich ťahoch ani jeden z hráčov nedosiahool víťaznú pozíciu.

Reprezentácia algoritmu pomocou stromu riešení



Vrchná časť stromu riešení



Dimenzia stavového priestoru je určená pomocou jednoduchých kombinatorických úvach takto

$$N = \sum_{p=1}^{5} \binom{9}{p} \left[\binom{9-p}{p-1} + \binom{9-p}{p} \right] - 2 \times 6 \times 13 = 5889$$

Pomocou metódy spätného prehľadávanie je možné zostrojiť celý strom riešení, kde počty koncových pozícií sú uvedené v tabuľke

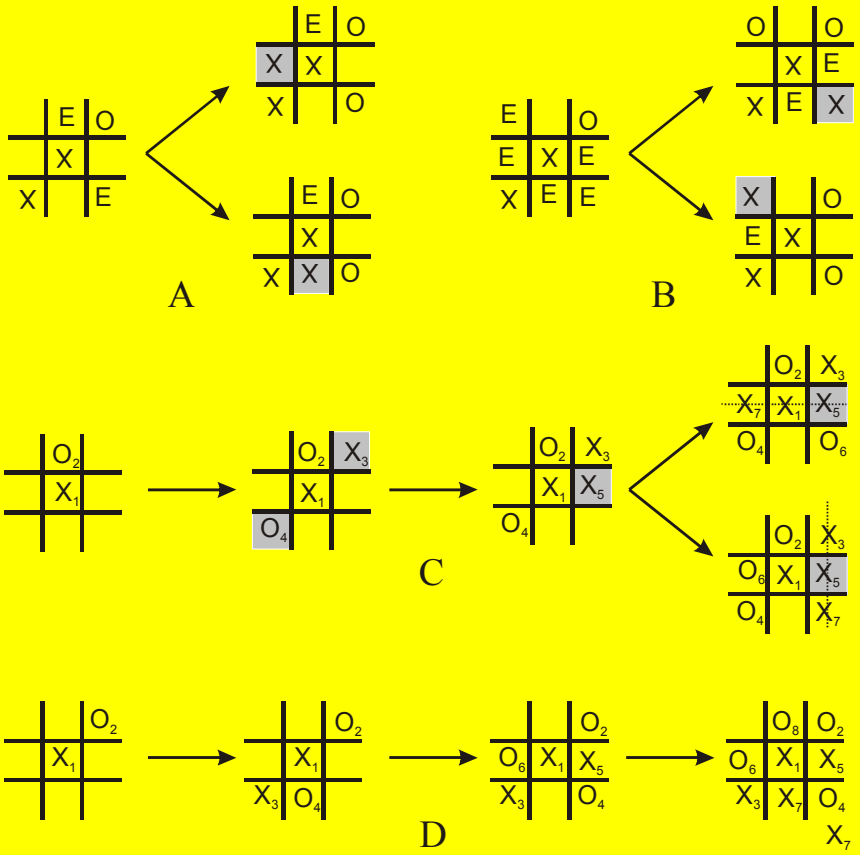
No.	Počet	Typ
1	131184	víťazstvo hráča X
2	77904	víťazstvo hráča O
3	46080	remíza hráčov X a O
	255168	celkový počet

Z tejto tabuľky vyplýva, že prvý hráč X má väčšiu šancu hru vyhrať. Podrobnou analýzou sa dá ukázať, že aj hráč O môže hru forsírovať tak, že remizuje. Celkový počet koncových vetví v strome riešení možno jednoducho odhadnúť ako 9!=362880.

Model hry

Model obsahuje 6 pravidiel s klesajúcou prioritou:

1. *Hráč vykoná ťah, ktorý vedie k jeho víťazstvu.*
2. *Hráč vykoná ťah, ktorý zabráni víťazstvu oponenta v nasledujúcom ťahu.*
3. *Hráč vykoná ťah, ktorým si pripraví možnosť dvojitého použitia 1. pravidla* (*tzv. vidlička*).
4. *Hráč vykoná ťah, ktorým zabráni oponentovi pripraviť "vidličku"*
5. *Hráč obsadí stredové pole.*
6. *Hráč obsadí rohové pole, ktorého proti-poloha je obsadená oponentom*
7. *Hráč obsadí rohové pole.*
8. *Hráč obsadí volné pole.*



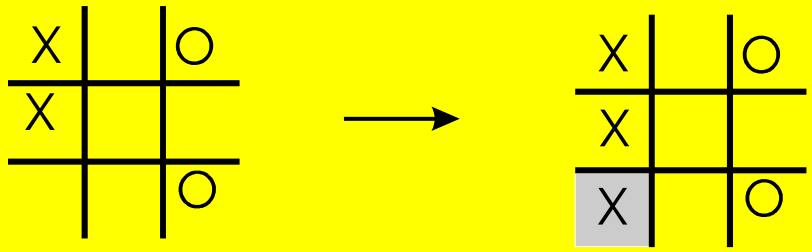
Diagramy A-B znázorňujú základné typy vidličkových pozícií, ktoré sú aplikovateľné použitím pravidiel 3 a 4. Tak napríklad, diagram A znázorňuje východiskovú pozíciu pre prípravu "vydličky"; písmena E špecifikujú prázdne bunky. Z diagramu A môžeme vytvoriť dve "vydličkové" pozície. Diagram C znázorňuje hru, ktorá je prehraná pre hráča O už po druhom ťahu. Druhý hráč urobil „fatálnu“ chybu, keď umiestnil svoju figuru O v druhom ťahu v druhom stĺpci hore, podľa pravidla mal obsadiť rohové pole. Diagram D znázorňuje hru, ktorá prebieha podľa pravidiel, končí remízou. .

Pozícia je reprezentovaná 9-rozmerným vektorom

$$x(P) = (x_1, x_2, \ldots, x_9) \in \{0, 1, -1\}^9$$

kde jednotlivé zložky určujú jednotlivé políčka v pozícii P

$$x_i = \begin{cases} 0 \ (i - \text{té pole je neobsadené}) \\ 1 \ (i - \text{té pole je obsadené X}) \\ -1 \ (i - \text{té pole je obsaden O}) \end{cases}$$



P=(1,0,-1,1,0,0,0,0,-1) ⟶ P'=(1,0,-1,1,0,0,1,0,-1)

Prvý model - agent hrá proti modelu hry piškvorky

Algoritmus modelu

1. krok. *Váhové koeficienty neurónovej siete sú náhodne vygenerované z intervalu [-1,1].*

2. krok. *Polož* t:=1.

3. krok. *S 50% pravdepodobnosťou deklaruj agenta ako prvého* X*-hráča a model ako druhého* O*-hráča (v opačnom prípade je agent deklarovaný ako druhý* O*-hráč a model ako prvý* X*-hráč). Na záver hry pomocou metódy TD(λ) opraví váhové koeficienty kognitívneho orgánu agenta.*

4. krok. *Polož* t:=t+1.

5. krok. *Ak* $t<t_{max}$*, potom pokračuj krokom 3, v opačnom prípade prejdi na krok 6.*

6. krok. *Koniec algoritmu.*

Parametre adaptačného procesu

2. α=0.1 (rýchlosť učenia).

7. λ=0.3 (temporal-difference parameter).

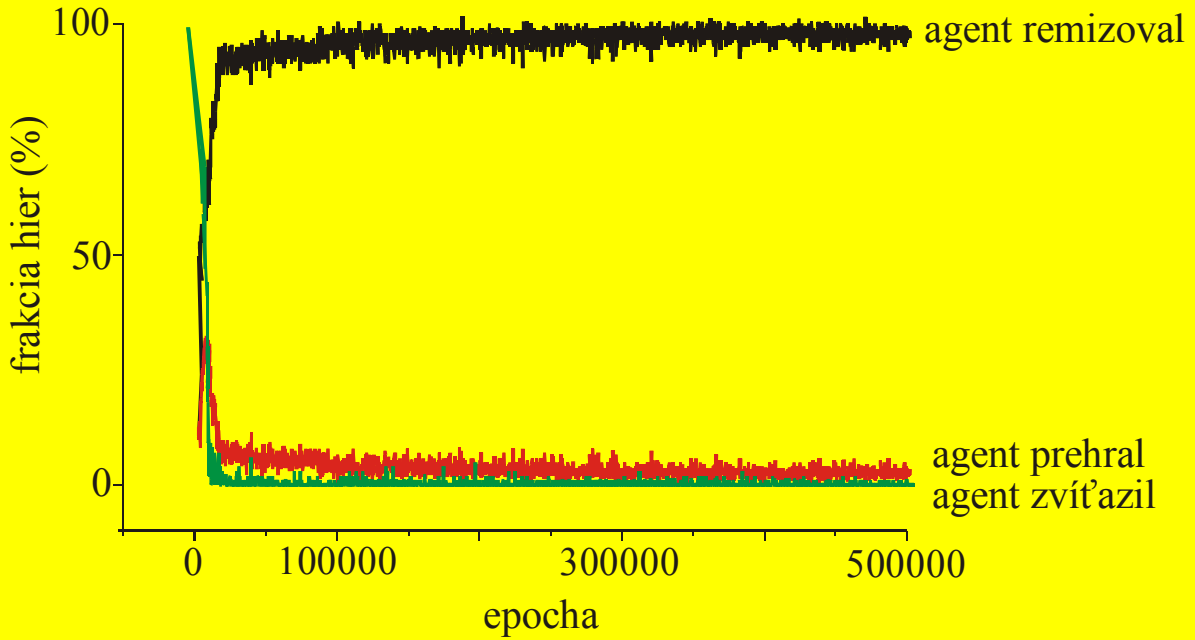
8. p=30 (počet skrytých neurónov).

9. Počiatočné hodnoty váhových a prahových koeficientov sú náhodne vyberané z otvoreného intervalu (-2,2).

10. Učenia je zastavené po 500000 epochách.

11. Pozície sú ohodnocované podľa formule

$$z = \begin{cases} 1 & (\text{prvý hráč vyhral}) \\ 0.5 & (\text{hráči remizovali}) \\ 0 & (\text{prvý hráč prehral}) \end{cases}$$



Priebeh frakcií hier (zo 100), ktoré hral agent proti modelu hry, pričom polovicu hier (50) hral ako prvý a druhú polovicu hier hral ako druhý. Z priebehu jednotlivých prípadov jasne vyplýva, že neurónová sieť je schopná tak kvalitnej spontánnej adaptácie, že dokáže neprehrávať s modelom.

Záver

- Uvedený subsymbolický prístup k riešeniu úloh, pre ktoré je len veľmi obtiažne zostrojiť ich efektívny model, poskytuje povzbudzujúce výsledky.
- V priebehu evolúcie agentov dochádza k emergencii stratégie hry.
- Pre hru backgamon získal Tesauro neurónovú sieť, ktorá je schopná hrať túto hru na veľmajstrovskej úrovni.
- Získané výsledky sú vynikajúcou ilustráciou subsymbolického prístupu k riešeniu zložitých úloh, ktoré sú ťažko riešiteľné technikami klasickej (symbolickej) umelej inteligencie.
- V rámci subsymbolického prístupu, založenom na neurónových sieťach s dopredným šírením signálu a adaptačnej metóde "učenia s odmenou a trestom" (reinforcement learning), je možné riešiť rôzne úlohy z robotiky, riadenia zložitých systémov, komplexných strategických hier, atď. bez nutnosti poznať ich model alebo databázu ich známych realizácií.

A cognitive joke

After D. J. Chalmers a 'cognitive joke' is a joke whose humour seems to rely on higher-level, more abstract cognitive processing in the brain, where offered solution is highly unexpected

The End