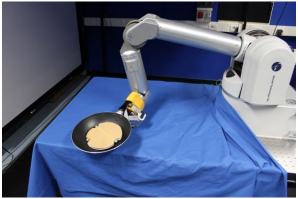
Aproximácia funkcie ohodnotení v algoritmoch Q-learning

Ing. Michal CHOVANEC
Fakulta riadenia a informatiky

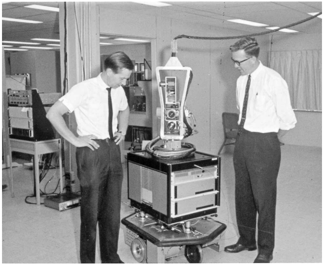
Marec 2016

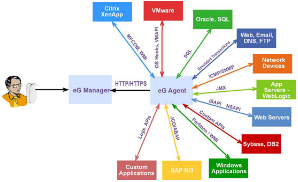
Obsah

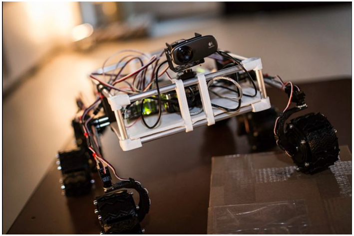
Využite q-learning algoritmu



Agentové systémy



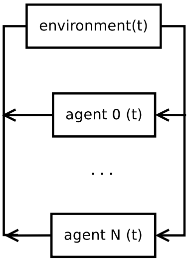




Agentové systémy

Racionálny agent :

- Schopný vnímať prostredie
- Robiť rozhodnutia
- Pre každú možnú postupnosť vstupov vyberá akciu maximalizujúcu očakavaný výkon



Obr. : Multiagentný systém

Adaptívne a učiace sa systémy

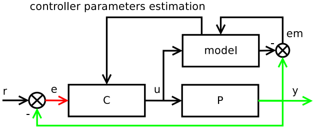
Adaptívny systém

- reaktívne správanie
- malá pamäť - bez očakávaní
- rychlá dynamika

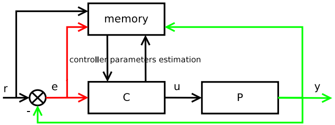
Učiaci sa systém

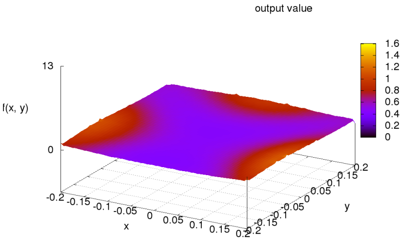
- konštruktívne správanie
- veľká pamäť - očakávania
- pomalá dynamika

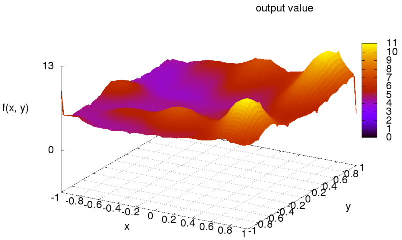
Adaptívny systém



Učiaci sa systém

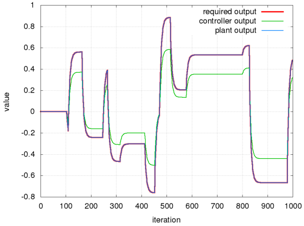


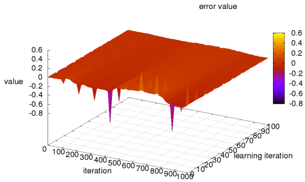




Adaptívny systém
PID regulátor

Učiaci sa systém
Iterative learning controll





$$u(n) = u(n-1) + b_0(n)e(n) + b_1(n)e(n-1) + b_2(n)e(n-2)$$

$$u_k(n) = u_{k-1}(n) + \gamma e_{k-1}(n) + \Gamma(e_{k-1}(n) - e_{k-2}(n))$$

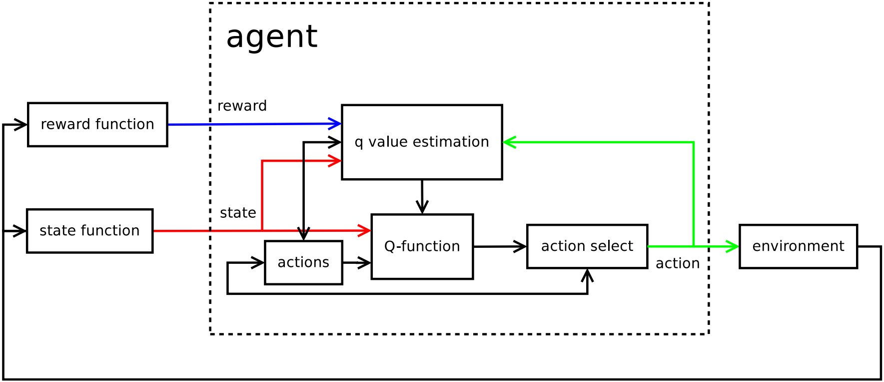
Q-learning algoritmus

Daná je množina stavov $\mathbb{S}$ a akcií $\mathbb{A}$, kde $\mathbb{S} \in \mathbb{R}^{n_s}$ a $\mathbb{A} \in \mathbb{R}^{n_a}$, kde n_s a n_a sú rozmery stavového vektora a vektora akcií. Je známa podmnožina východiskových stavov $\mathbb{S}_0$.
Existuje prechodová funkcia

$$s(n+1) = \lambda(s(n), a(n)) \tag{1}$$

zo stavu $s(n)$ použitím akcie $a(n)$ - táto funkcia je ale agentovi neznáma.

Agent začlenený do prostredia



Odmeňovacia funkcia

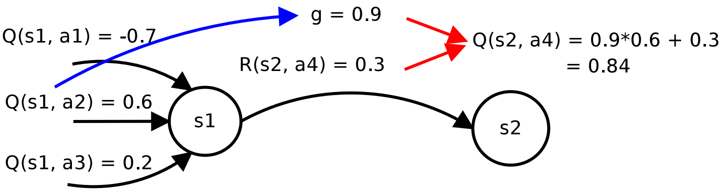
Daná je funkcia ohodnotení

$$Q(s(n), a(n)) = R(s(n), a(n)) + \gamma \max_{a(n-1) \in \mathbb{A}} Q(s(n-1), a(n-1))$$

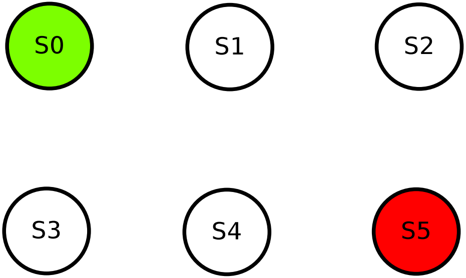
kde

- $R(s(n), a(n))$ je odmeňovacia funkcia s hodnotami v $\langle -1, 1 \rangle$,
- $Q(s(n-1), a(n-1))$ je odmeňovacia funkcia v stave $s(n-1)$ pre akciu $a(n-1)$,
- γ je odmeňovacia konštanta a platí $\gamma \in (0, 1)$.

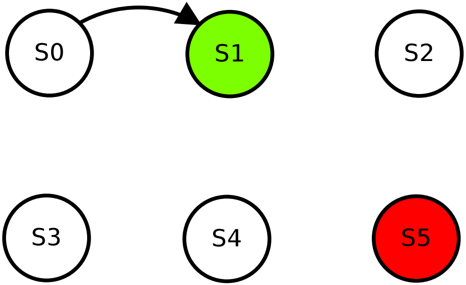
Odmeňovacia funkcia



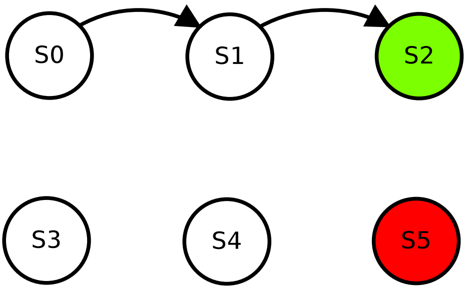
Ilustračný príklad - inicializácia



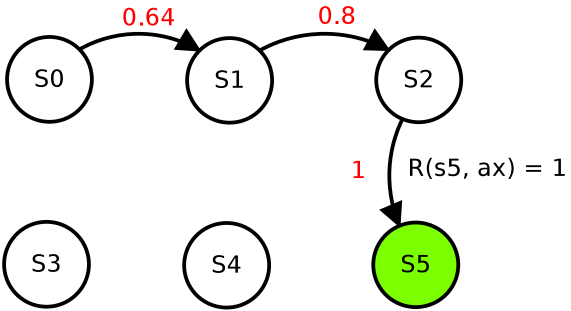
Ilustračný príklad - prechod do ďalšieho stavu



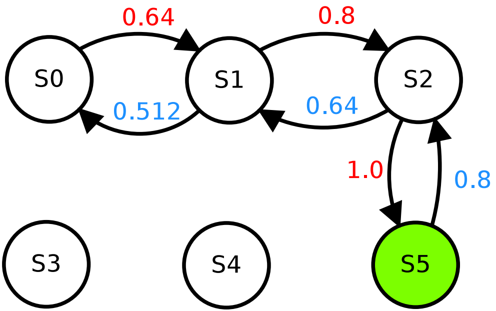
Ilustračný príklad - prechod do ďalšieho stavu



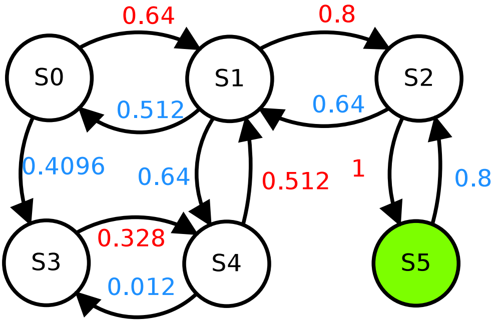
Ilustračný príklad - prechod do cieľového stavu



Ilustračný príklad - ďalšie prechody



Ilustračný príklad - konečný stav algoritmu



R(s4, ay) = -0.5

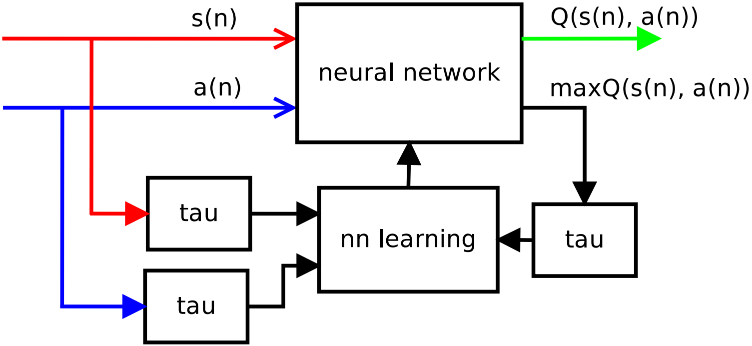
Implementačné problémy

Problémy tabuľkovej interpretácie $Q(s(n), a(n))$:

- pre veľké n_s alebo n_a narastajú pamäťové nároky,
- o nevyplnených $Q(s(n), a(n))$ nevieme povedať nič,
- pre rozsiahle stavové priestory ťažko vypočítateľné,
- ako aproximovať $Q(s(n), a(n))$?

Aproximácia neurónovou sieťou

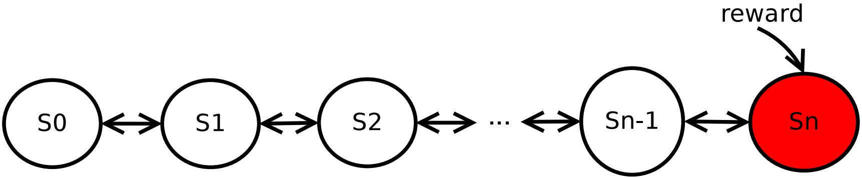
Utopická predstava :



Prečo nedáva správne výsledky?

Hypotéza

Na základe experimetov



Pre korektné vyplnenie hodnôt v s_{n-1} sa vyžaduje korektá hodnota v s_n

$$Q(s(1),a(1)) = R(s(1),a(1)) + \gamma \max_{a(0)\in\mathbb{A}} Q(s(0),a(0))$$

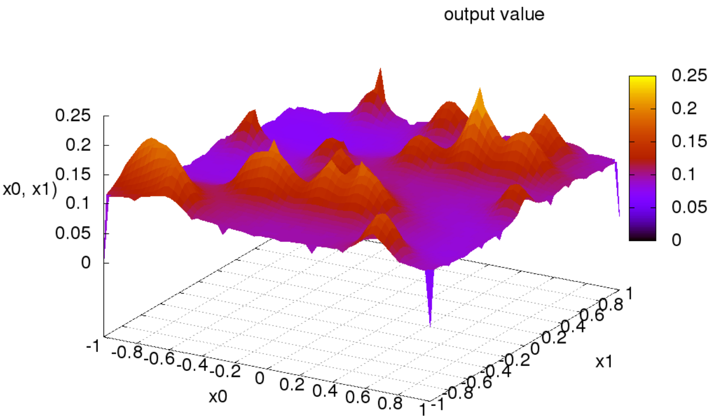
$$Q(s(2),a(2)) = R(s(2),a(2)) + \gamma \max_{a(1)\in\mathbb{A}} Q(s(1),a(1))$$

$$\ldots$$

Učenie doprednej siete

- Nie je homogénne!
- V priebehu učenia $Q(s(n), a(n))$ chaoticky osciluje okolo požadovanej hodnoty.
- Ani po 10-mil. iteráciach sa hodnota neustáli na požadovanej hodnote.

Je možné zostaviť neurónovú sieť, ktorá sa dá naučiť lokálne?



Rozklad $Q(s(n), a(n))$ na bázické funkcie

$$f_j^1(s(n), a(n)) = e^{-\sum_{i=1}^{n_s} \beta_{aji}(n)(s_i(n) - \alpha_{aji}(n))^2}$$

$$f_j^2(s(n), a(n)) = \frac{1}{1 + \sum_{i=1}^{n_s} \beta_{aji}(n)(s_i(n) - \alpha_{aji}(n))^2}$$

$$f_j^3(s(n), a(n)) = e^{-\sum_{i=1}^{n_s} \beta_{aji}(n)|s_i(n) - \alpha_{aji}(n)|}$$

$$f_j^4(s(n), a(n)) = \sum_{k=1}^{m} \sum_{i=1}^{n_s} \beta_{aji}(n)(s_i(n) - \alpha_{aji}(n))^k$$

Voľba bázických funkcií

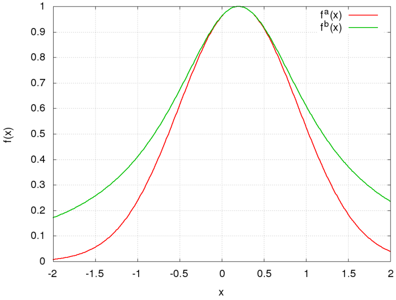
Vzhľadom na charakter učiaceho algoritmu

$$Q(s(n), a(n)) = R(s(n), a(n)) + \gamma \max_{a(n-1) \in \mathbb{A}} Q(s(n-1), a(n-1))$$

boli zvolené bázické funkcie (parameter n pre prehľadnosť vynechaný)

$$f_j^1(s,a) = e^{-\sum_{i=1}^{n_s} \beta_{aji}(s_i - \alpha_{aji})^2}$$

$$f_j^2(s,a) = \frac{1}{1 + \sum_{i=1}^{n_s} \beta_{aji}(s_i - \alpha_{aji})^2}$$



Q-learning algoritmus - aproximácia

Pre symetrické prechody medzi stavmi možno zjednodušiť na

$$f_j^1(s,a) = e^{-\beta_{aj} \sum\limits_{i=1}^{n_s} (s_i - \alpha_{aji})^2}$$

$$f_j^2(s,a) = \frac{1}{1 + \beta_{aj} \sum\limits_{i=1}^{n_s} (s_i - \alpha_{aji})^2}$$

a ich lineárna kombinácia

$$Q^x(s,a) = \sum_{j=1}^{l} w_{ja} f_j^x(s,a)$$

kde l je počet bázických funkcií a x je voľba typu bázickej funkcie.

Voľba parametrov

- bázicke funkcie musia rovnomerne pokryť stavový priestor,
- parameter $\alpha_{ji}(n)$ reprezentuje posunutie bázickej funkcie - bod s najväčou funkčnou hodnotou,
- parameter $\beta_j(n)$ reprezentuje strmosť bázickej funkcie.

Určenie parametrov $\alpha_{jia}(n)$

Paramatre $\alpha_{jia}(n)$ - pokrytie stavového priestoru do oblastí podľa veľkosti $R(s(n), a(n))$. Využije sa princíp Kohonenovej siete - najbližšie vzory $\alpha_{jia}(n)$ sa posunú podľa vstupných vektorov tak aby vrchol Gaussovej krivky ležal v ťazisku.

- na začiatku sa zvolia $\alpha_{jia}(n)$ náhodne
- spočítajú sa vzdialenosti od predloženého vstupu $d_{ja}(n) = \mid s(n) - \alpha_{ja}(n) \mid$
- nájde sa také ka kde pre $\forall j : d_{ka}(n) \leq d_{ja}(n)$
- spočíta sa krok učenia $\eta'_a(n) = \eta_1 \mid Q_r(s(n), a(n)) \mid$
- upravia sa parametre $\alpha_{aki}(n+1) = (1 - \eta')\alpha_{aki}(n) + \eta' s_i(n)$

kde
$Q_r(s(n), a(n))$ je požadovaný výstup
η_1 je konštanta učenia

Určenie parametrov $\beta_{ja}(n)$

Paramatre $b_{ja}(n)$ - určuje strmosť krivky

- stanoví sa chyba $e(n) = Q_r(s(n), a(n)) - Q(s(n), a(n))$
- pre každú bázickú funkciu $\beta_{ja}(n+1) = \beta_{ja}(n) + \eta_2 e(n) w_{ja}(n)$
- skontroluje sa $\beta_{ja}(n) \in (0, \infty)$

kde
$Q_r(s(n), a(n))$ je požadovaný výstup
η_2 je konštanta učenia

Určenie parametrov $w_{ja}(n)$

Paramatre w_j - váhové parametre

- stanoví sa chyba $e(n) = Q_r(s(n), a(n)) - Q(s(n), a(n))$
- pre každé w_{ja} : $w_{ja}(n+1) = w_{ja}(n) + \eta_3 e(n) y_j(n)$
- skontroluje sa $w_{ja}(n) \in (-r, r)$

kde
η_3 je konštanta učenia
r je maximálny rozsah váh

Bloková schéma syntézy testovaného riešenia

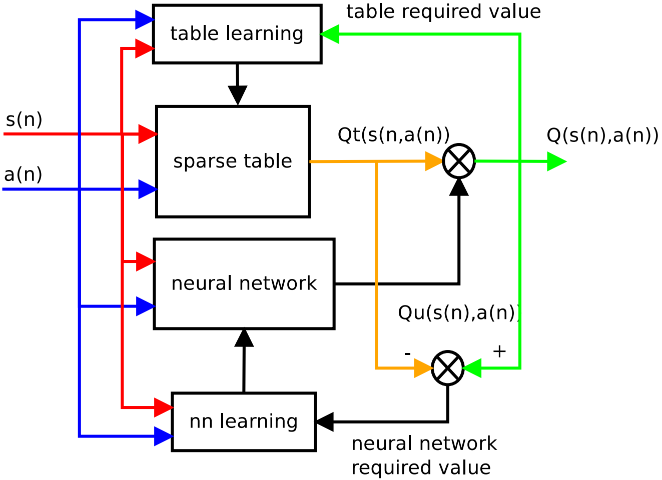
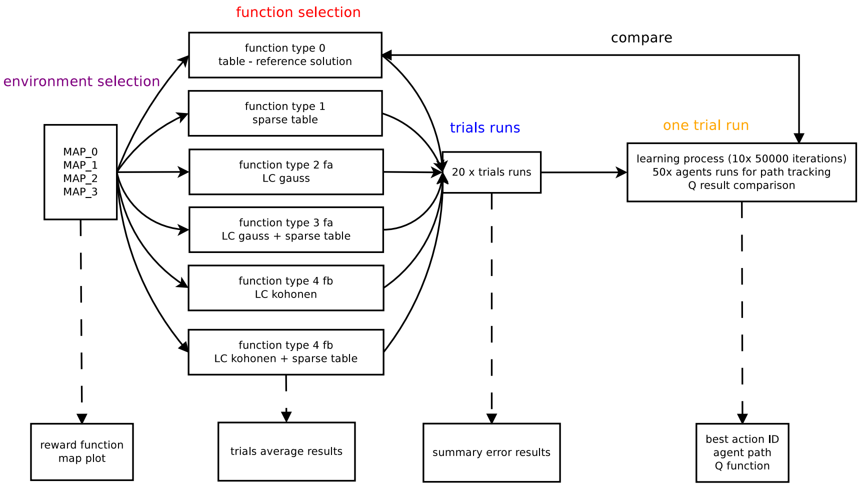


Schéma priebehu experimentov



Návrh experimentov - podmienky

- 50000 iterácií učenia
- rozmer s je $n_s = 2$, rozmer a je $n_a = 2$
- predpis funkcie ohodnotení

$$
\begin{aligned}
&Q(s(n), a(n)) = \\
&\alpha Q(s(n-1), a(n-1)) \\
&(1-\alpha)(R(s(n), a(n)) + \gamma \max_{a(n-1) \in \mathbb{A}} Q(s(n-1), a(n-1))
\end{aligned}
$$

- $R(s(n), a(n)) \in \langle -1, 1 \rangle$ náhodná mapa s 1 cieľovým stavom
- $\gamma = 0.98$ a $\alpha = 0.7$
- hustota referenčného riešenia = 1/32 (4096 stavov)
- počet akcií v každom stave = 8
- hustota riedkej tabuľky = 1/8 (1:16 pomer)
- počet bázických funkcií $l = 64$
- rozsah parametrov
  - $\alpha_{ja}(n) \in \langle -1, 1 \rangle$
  - $\beta_{ja}(n) \in \langle 0, 200 \rangle$
  - $w_{ja}(n) \in \langle -4, 4 \rangle$

Návrh experimentov - podmienky

$Q_{rt}(s(n), a(n))$ referenčná funkcia Q (funkcia 0), kde $t \in \langle 0, 19 \rangle$ je číslo trialu
$Q_{jt}(s(n), a(n))$ testované funkcie Q a $j \in \langle 1, 5 \rangle$.
Celková chyba behu trialu t je

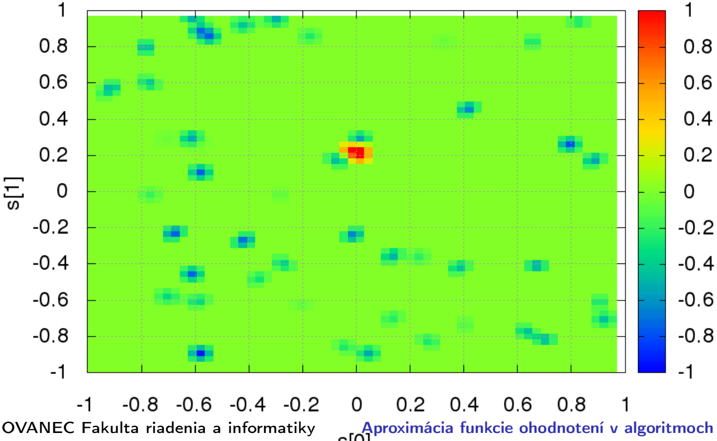
$$e_{jt} = \sum_{s,a} (Q_{rt}(s, a) - Q_{jt}(s, a))^2$$

priemerná, minimálna, maximálna chyba a smerodatná odchylka

$$\bar{a}_j = \frac{1}{20} \sum_t e_{jt}$$
$$e_j^{min} = \min_t e_{jt}$$
$$e_j^{max} = \max_t e_{jt}$$
$${\sigma_j}^2 = \frac{1}{20} \sum_t (\bar{a}_j - e_{jt})^2$$

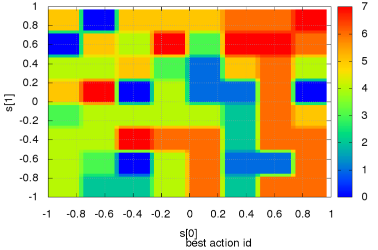
Funkcia R(s, a), mapa 1 - Výsledky experimentov

pre každý stav je zvolená rovnaka množina akcií.
Ďalej platí $s = (s[0], s[1])$.

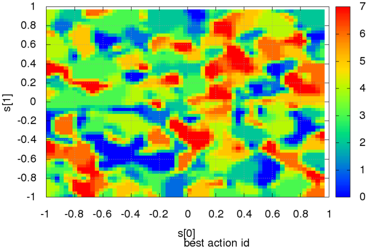


Mapa najlepších akcií - Výsledky experimentov

Funkcia voľby najlepšej z 8 akcií v stave $s = (s[0], s[1])$.

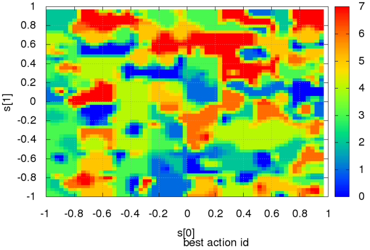


Obr. : sparse table

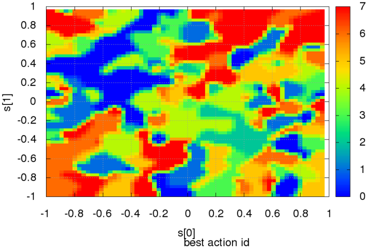


Obr. : linear combination Gauss

Funkcia voľby najlepšej z 8 akcií v stave $s = (s[0], s[1])$.



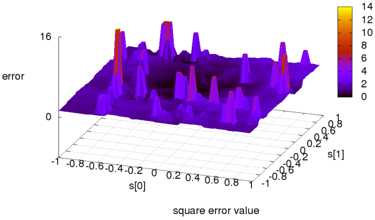
Obr. : sparse table + linear combination Gauss



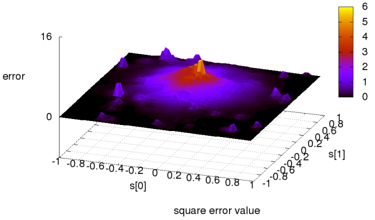
Obr. : linear combination Kohonen function

Chybové funkcie - Výsledky experimentov

$$e_{jt}(s) = (Q_{rt}(s, a - Q_{jt}(s, a))^2$$



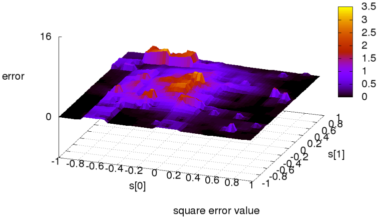
Obr. : sparse table



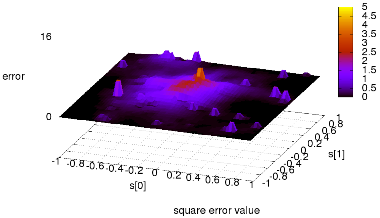
Obr. : linear combination Gauss

Chybové funkcie - Výsledky experimentov

$$e_{jt}(s) = (Q_{rt}(s, a - Q_{jt}(s, a))^2$$

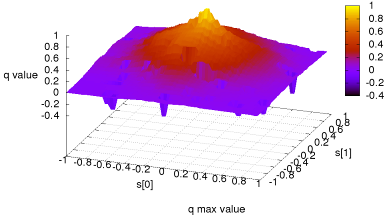


Obr. : sparse table + linear combination Gauss

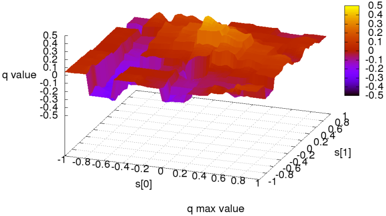


Obr. : linear combination Kohonen function

max Q(s, a) - Výsledky experimentov

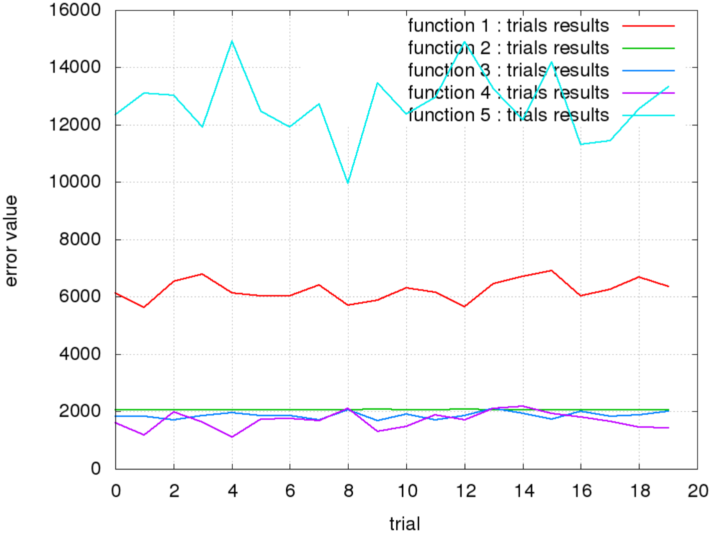


Obr. : reference table

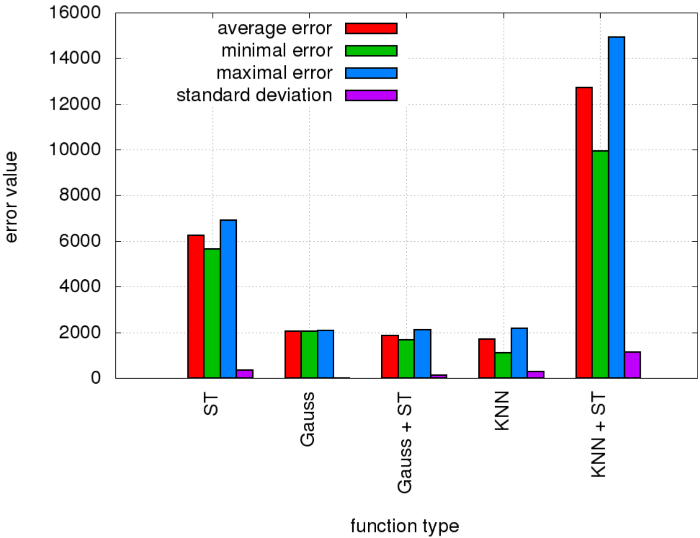


Obr. : sparse table + linear combination Gauss

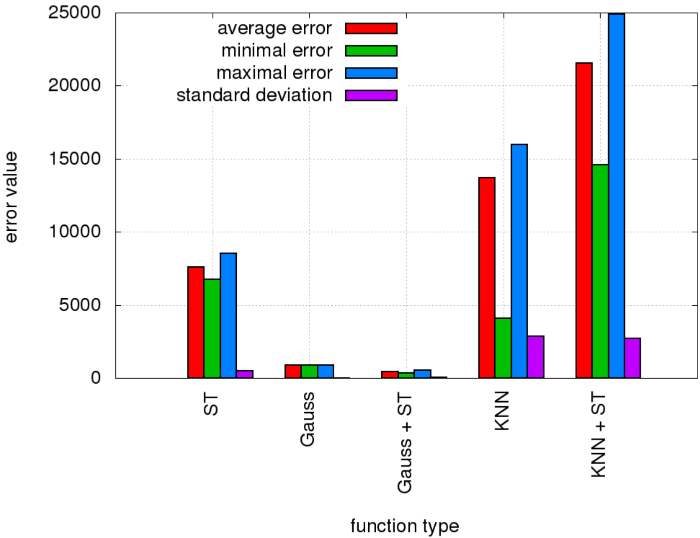
Priebeh trialov - Výsledky experimentov



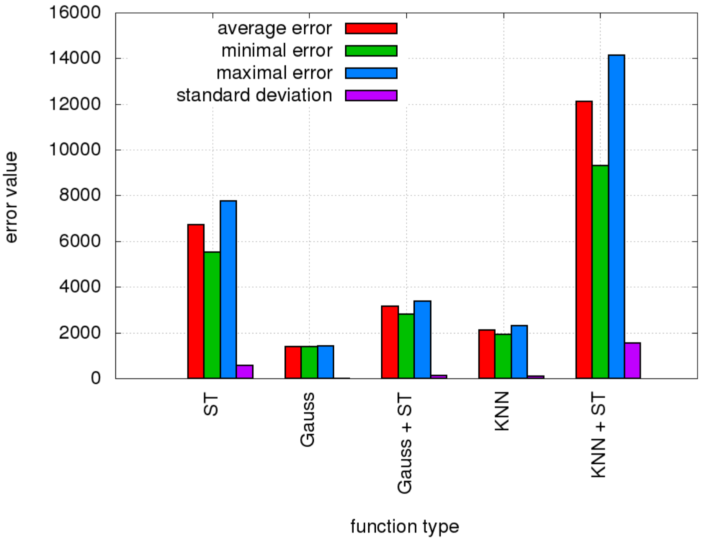
Mapa 1 - Výsledky experimentov



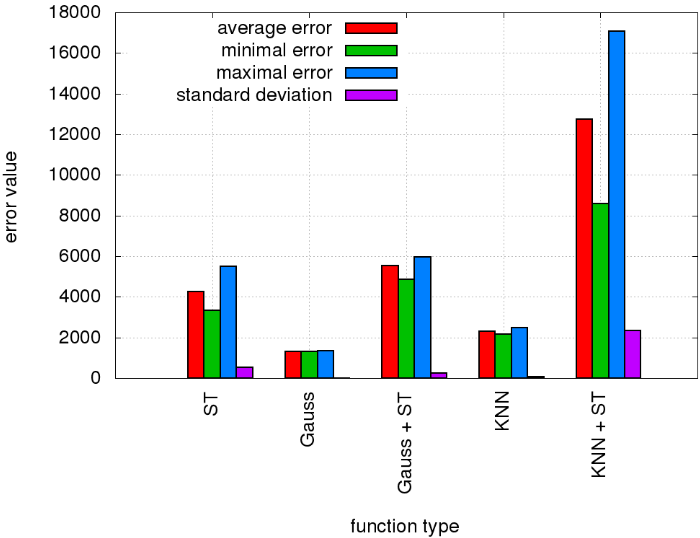
Mapa 0 - Výsledky experimentov



Mapa 2 - Výsledky experimentov



Mapa 3 - Výsledky experimentov



Ďakujem za pozornosť

michal.chovanec@yandex.ru

https://github.com/michalnand/q_learning