

# Aproximácia funkcie ohodnotení v algoritmoch Q-learning neurónovou sieťou

Ing. Michal CHOVANEC (2013..2016)  
Fakulta riadenia a informatiky

*August 2016*

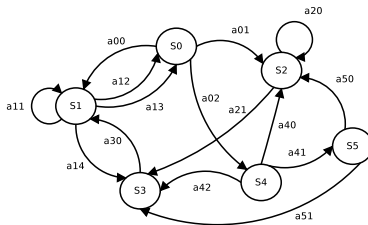
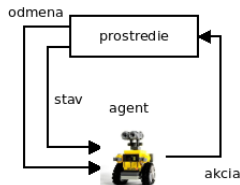
katedra technickej kybernetiky  
školiteľ : prof. Ing. Juraj Miček, PhD.

# Reinforcement learning

“A way of programming agents by reward and punishment without needing to specify how the task is to be achieved.”

[Kaelbling, Littman, Moore, 96]

- 1 Zistenie stavu
- 2 Výber akcie
- 3 Vykonanie akcie
- 4 Prechod do ďalšieho stavu
- 5 Získanie odmeny alebo trestu
- 6 Učenie sa zo získanej skúsenosti



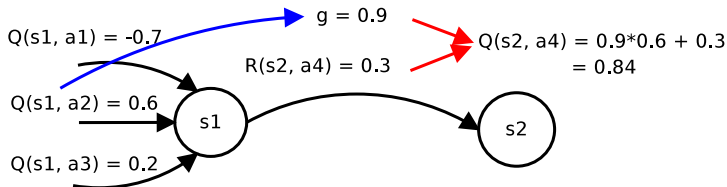
# Funkcia ohodnotení

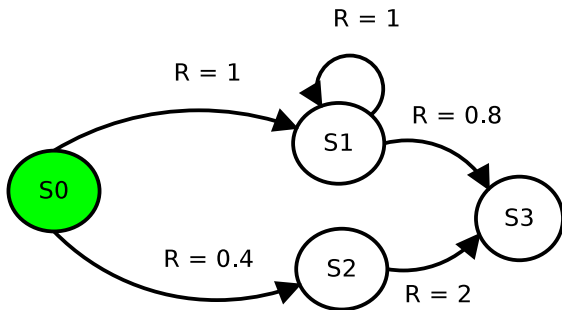
Daná je funkcia ohodnotení (Watkins 1989, konvergencia : Melo, Even-Dar, Kearns)

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

kde

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





Ohodnotenie ciest :

- $S(S0, S2, S3) = 2 + 0.9 * 0.4 = 2.36$
- $S(S0, S1, S3) = 1 + 0.9 * 1 = 1.9$
- $S(S0, S1, S1, S1) = 1 + 0.9 * (1 + 0.9 * 1) = 2.71$
- $S(S0, S1, S1, S1, S1) = 1 + 0.9 * (1 + 0.9 * (1 + 0.9 * 1)) = 3.439$
- $S(S0, S1, S1, S1, S1, S1, ...) = 10 < \text{—}$
- $S(S0, S1, S1, S1, S1, S1, ..., S3) = 10.8 < \text{—}$

# Implementačné problémy

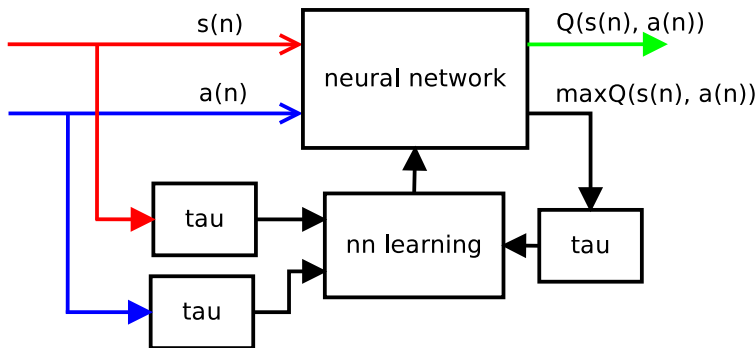
Problémy tabuľkovej interpretácie  $Q(s(n), a(n))$  : (Philip Sabes 1996, Melo 2007, Alborz Geramifard 2013)

- ① pre veľké počty stavov, mnohorozmerné stavové priestory alebo veľký počet akcií narastajú pamäťové nároky
  - robot s 20 senzormi kde každý má 256 hodnôt sa môže nachádzať v  $1.46 * 10^{48}$  stavoch
  - odhadovaný počet atómov v pozorovateľnom vesmíre je  $10^{80}$
- ② o nevyplnených  $Q(s(n), a(n))$  nevieme povedať nič,
- ③ pre rozsiahle stavové priestory ťažko vypočítateľné,
- ④ ako aproximovať  $Q(s(n), a(n))$ ?

cieľ : Aproximovať funkciu  $Q(s(n), a(n))$ , so zreteľom na rýchlosť učenia.

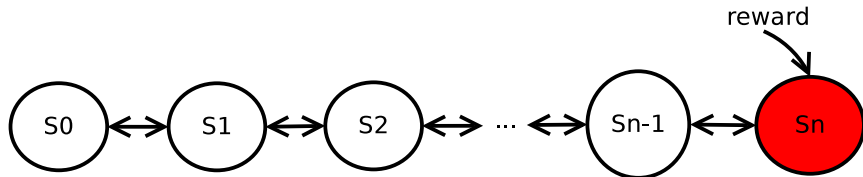
# Aproximácia doprednou perceptronovou neurónovou sieťou

Utopická predstava : (Sebastian Thrun 1993, Leemon Baird N/A, Silvia Ferrari 2005, Ma Yumei 2012)



Prečo nedáva správne výsledky?

Na základe experimentov - Snowball problém



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

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

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

...

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

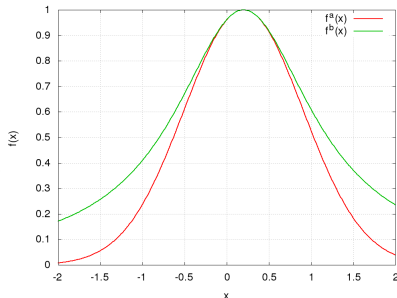
Vzhľadom na charakter učiaceho algoritmu

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

boli zvolené bázické funkcie

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

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



a ich lineárna kombinácia

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



# Nová základná funkcia

Tabuľka pre vybrané hodnoty (adaptívna tabuľka - AT)- umožní zachytiť skokovú zmenu

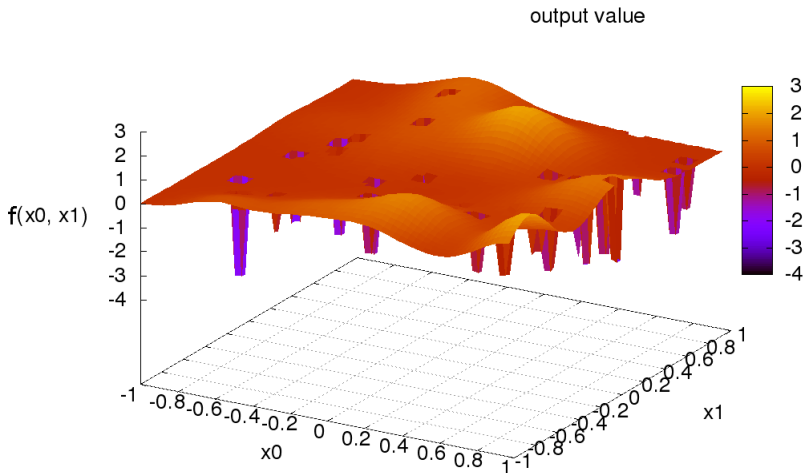
Gaussova krivka - dokáže pokryť nenulovými hodnotami celý definičný obor

$$P_i(s(n), a(n)) = \begin{cases} r_{ai} & \text{ak } s(n) = \alpha_i^1 \\ 0 & \text{inak} \end{cases} \quad (1)$$

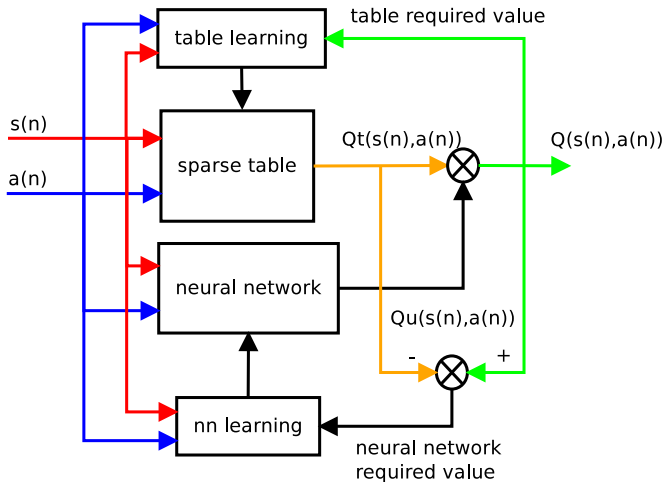
$$H_j(s(n), a(n)) = w_{aj} e^{-\beta_{aj} \sum_{i=1}^{n_s} (s_i(n) - \alpha_{aji}^2)^2} \quad (2)$$

$$Q(s(n), a(n)) = \sum_{i=1}^I P_i(s(n), a(n)) + \sum_{j=1}^J H_j(s(n), a(n)) \quad (3)$$

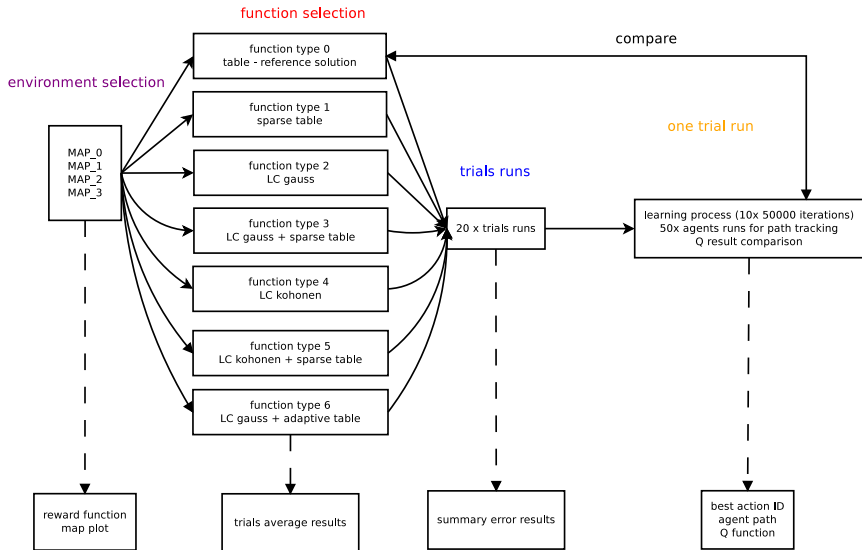
# Nová bázičná funkcia



# 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

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

- $R(s(n), a(n)) \in \langle -1, 1 \rangle$  náhodné prostredie (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 bazických funkcií  $l = 64$
- rozsah parametrov  $\alpha_{ja}(n)$ ,  $\beta_{ja}(n)$ ,  $w_{ja}(n)$

# Návrh experimentov - podmienky

$Q_{rt}(s(n), a(n))$  referenčná funkcia  $Q$  (funkcia  $r = 0$ ), 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 smerodajná odchylka

$$\bar{a}_j = \frac{1}{20} \sum_t e_{jt}$$

$$e_j^{\min} = \min_t e_{jt}$$

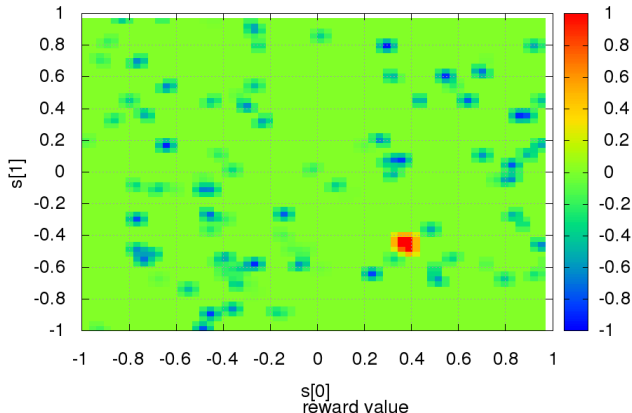
$$e_j^{\max} = \max_t e_{jt}$$

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

# Funkcia $R(s, a)$ , prostredie 2 - Výsledky experimentov

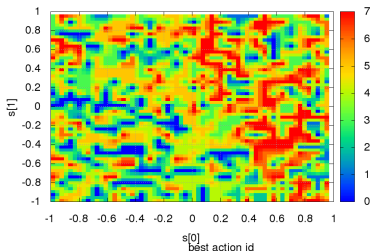
Pre každý stav je zvolená rovnaka množina akcií.

Ďalej platí  $s = (s[0], s[1])$ .

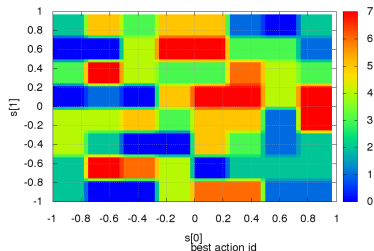


# Mapa najlepších akcií - Výsledky experimentov

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



Obr.: reference solution

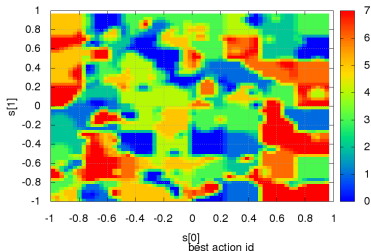


Obr.: sparse table

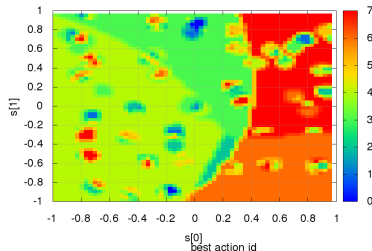


# Mapa najlepších akcií - Výsledky experimentov

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

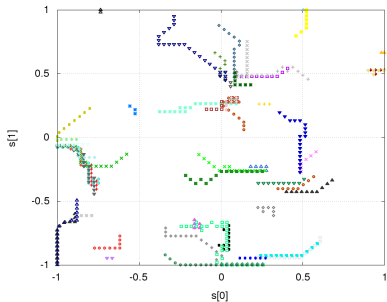


Obr.: sparse table + linear combination Gauss

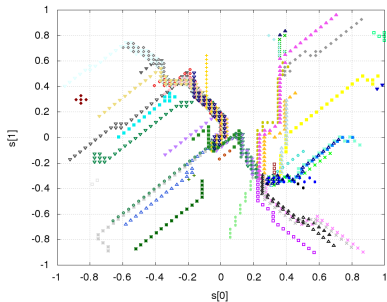


Obr.: adaptive table + linear combination Gauss

# Dráhy agentov pri voľbe najlepšej akcie

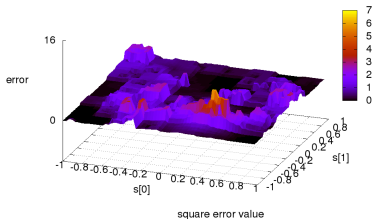


Obr.: sparse table + linear combination Gauss

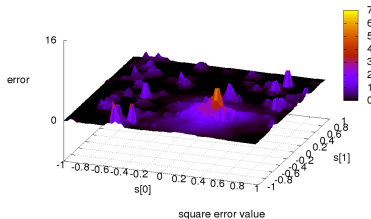


Obr.: adaptive table + linear combination Gauss

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

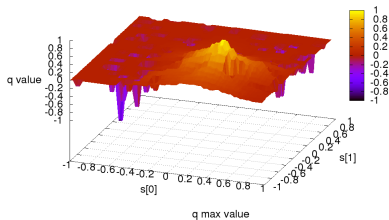


Obr.: sparse table + linear  
combination Gauss

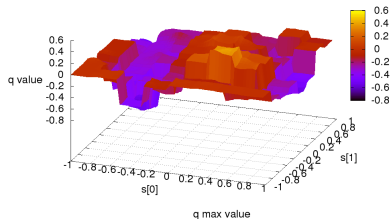


Obr.: adaptive table + linear  
combination Gauss

# max $Q(s, a)$ - Výsledky experimentov

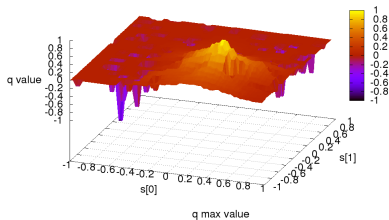


Obr.: reference table

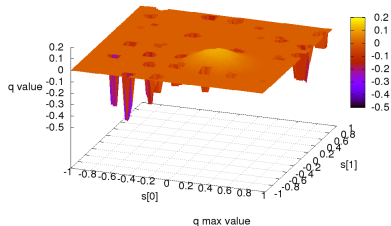


Obr.: sparse table + linear combination Gauss

# max $Q(s, a)$ - Výsledky experimentov

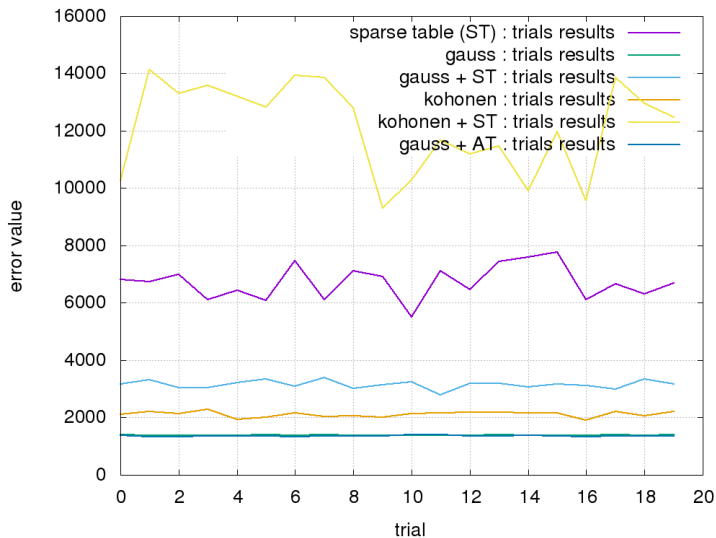


Obr.: reference table

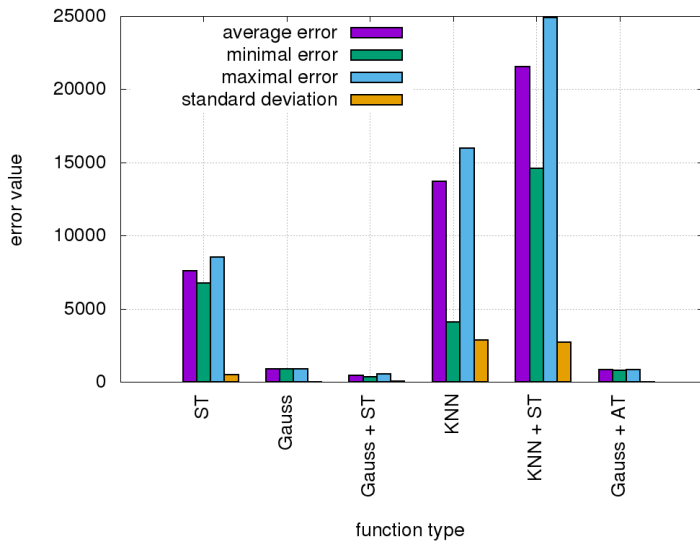


Obr.: adaptive table + linear combination Gauss

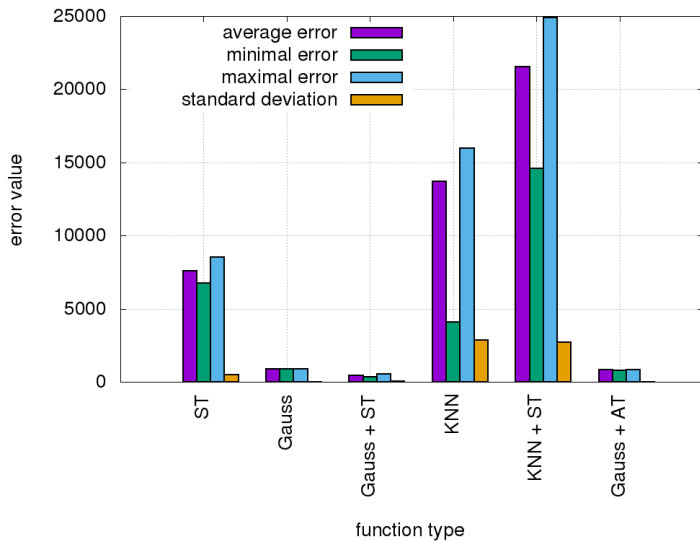
# Priebeh trialov - Výsledky experimentov



# Prostredie 0 - Výsledky experimentov

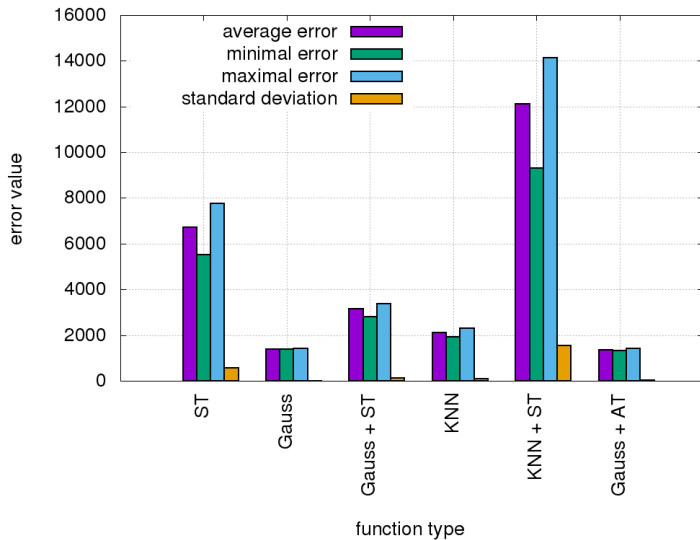


# Prostredie 1 - Výsledky experimentov

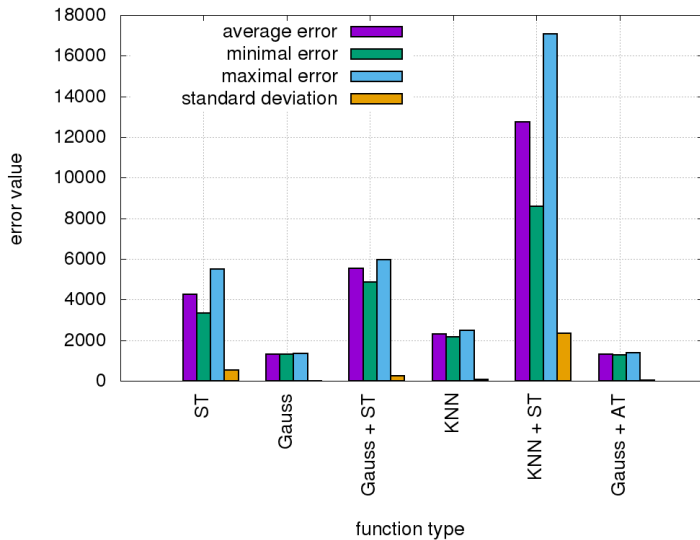




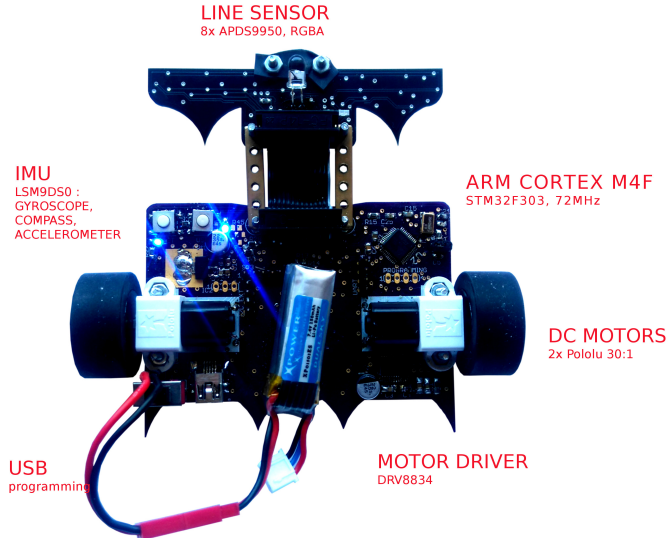
# Prostredie 2 - Výsledky experimentov



# Prostredie 3 - Výsledky experimentov

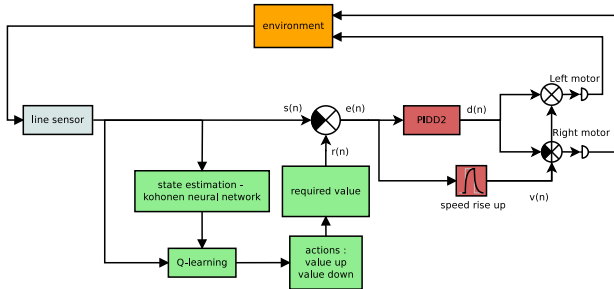


# Praktický experiment - robot sledující čiaru



# Praktický experiment - robot sledující čiaru

## Bloková schéma radiaceho bloku robota



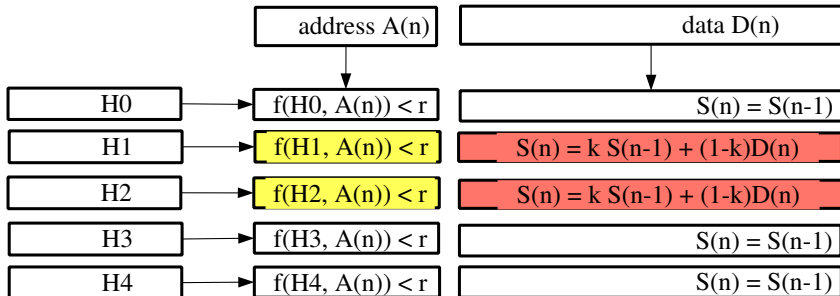
# Ďalšie smerovanie - sparse distributed memory (SDM)

sparse distributed memory - Pentti Kenerva 1988

hierarchical temporal memory - Jeff Hawkins 2012

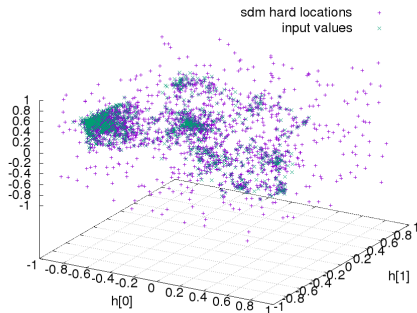
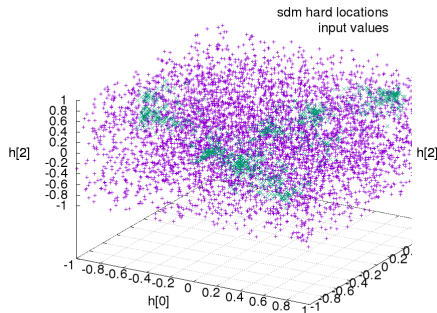
adaptive sparse distributed memory - Michal Chovanec 2016 (v tlači)

aproximácia  $D(n) = f(A(n))$



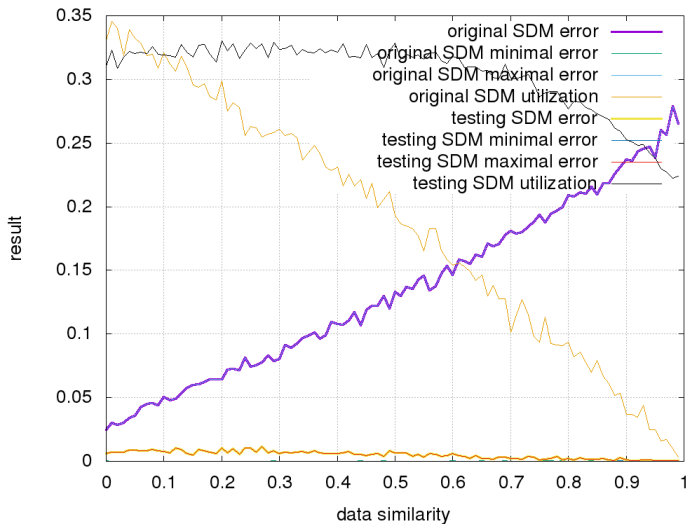
# Adaptive sparse distributed memory

Pokrytie priestoru s hard locations (3D rez 50D priestorom)



# Experimentálne výsledky

- aproximácia funkcie, predikcia časových radov, MNIST - rukou písané číslice



- Intelligent traffic-safety mirror [Inteligentné dopravné zabezpečovacie zrkadlo] / M. Hodoň, M. Chovanec, M. Hyben.
- Intelligent traffic-safety mirror by using wireless sensor network [Inteligentné dopravné-bezpečnostné zrkadlo s použitím bezdrôtovej siete] / Peter Danišovič, Michal Hodoň, Michal Chovanec.
- Tiny low-power WSN node for the vehicle detection [Jednoduchý energeticky-efektívny nód bezdrôtovej senzorovej siete určený na detekciu automobilov] / Michal Chovanec, Michal Hodoň and Lukas Cechovic.
- Maximizing performance of low-power WSN node on the basis of event-driven-programming approach : Minimization of operational energy costs of WSN node control unit / Michal Hodoň ... [et al.].
- Real-time schedule for mobile robotics and WSN applications / Michal Chovanec, Peter Šarafin.
- Universal synchronization algorithm for wireless sensor networks - FUSA algorithm / Michal Chovanec ... [et al.].
- Investigation of the gyro-sensor contribution to the straight movement of vehicle [Analýza vplyvu gyroskopického senzora pri priamom pohybe vozidla] / Michal Hodoň, Michal Chovanec.
- Udalosťami riadené programovanie v OS Suzuha = Event driven programming in Suzuha OS / Michal Chovanec.
- Required value classification using Kohonen neural network = Klasifikácia žiadanej hodnoty Kohonenovou neurónovou sieťou / Michal Chovanec.
- Akcelerometrické meranie výstrelu z luku = Accelerometrics measuring bow shot / Michal Chovanec a Jaroslav Múčka.
- Wireless sensor networks for intelligent transportation systems / Michal Hodoň, Juraj Miček, Michal Chovanec.
- Signal mixing using neural network / Michal Chovanec.



# Ďakujem za pozornosť

michal.chovanec@yandex.ru

[https://github.com/michalnand/q\\_learning](https://github.com/michalnand/q_learning)

