

VIZUÁLNA NAVIGÁCIA



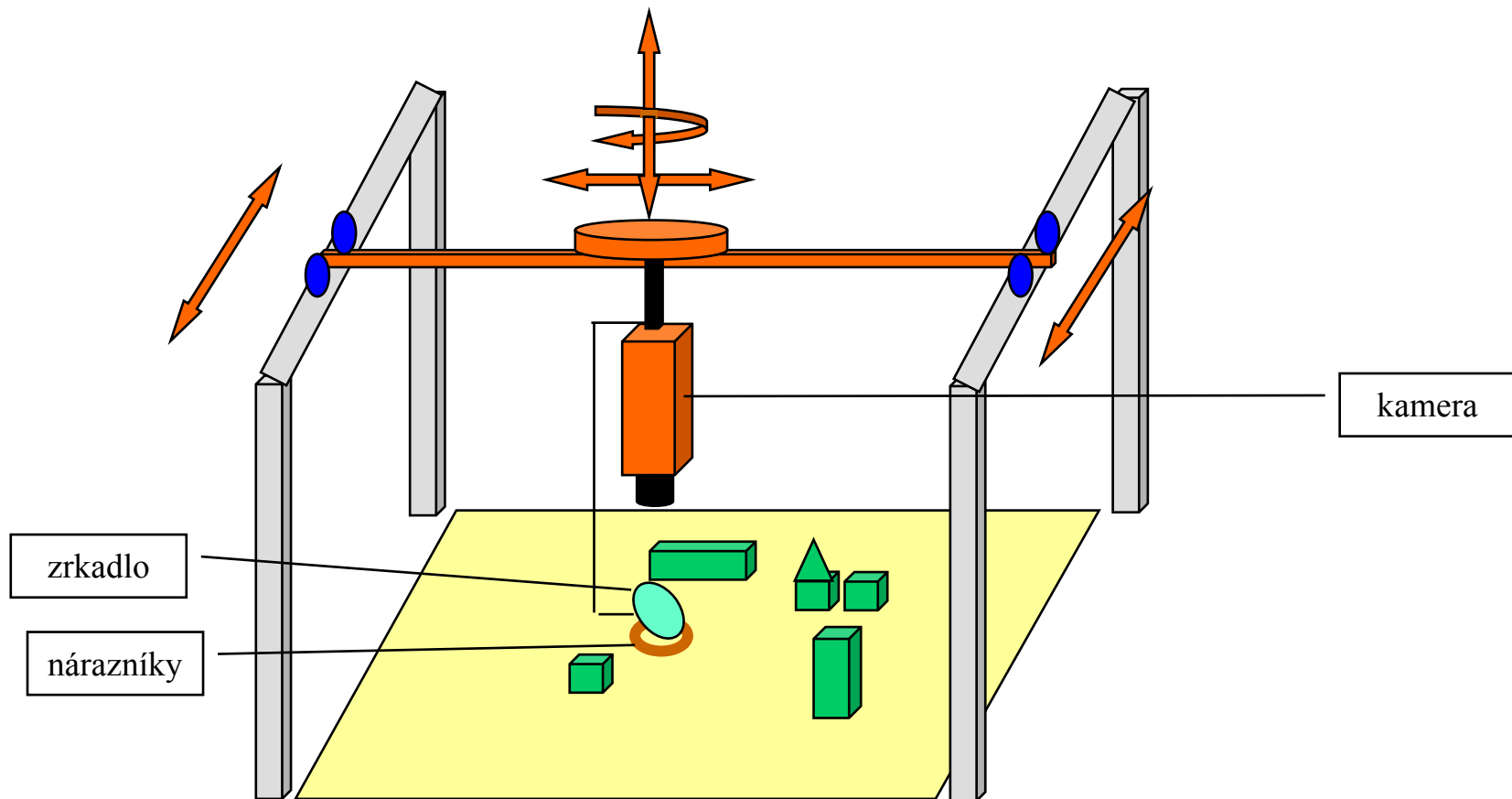
■ Ciele

- ◆ Problém Bootstrappingu
- ◆ Zrýchlenie evolúcie
 - ▼ Simulovaná
 - ▼ Doladenie na reálnom robotovi

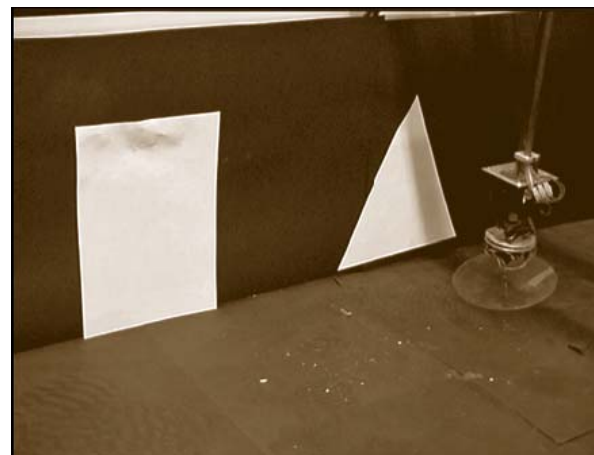
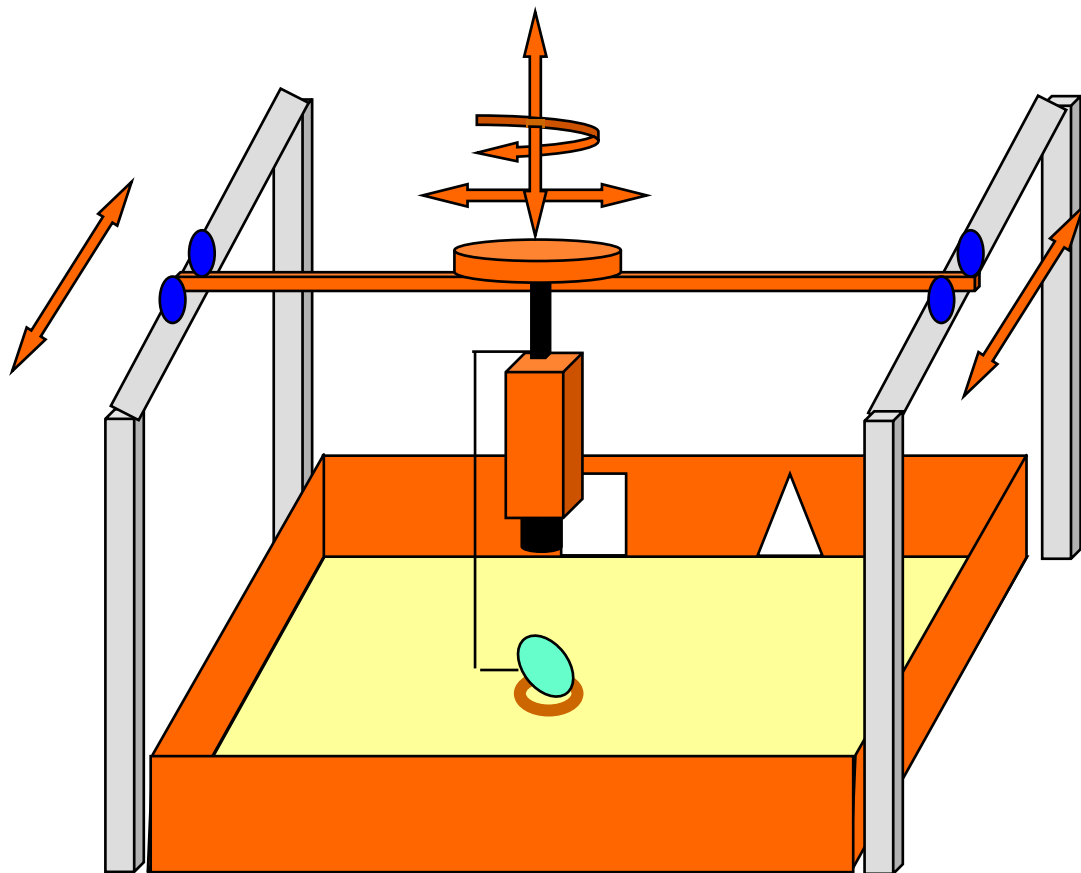
VIZUÁLNA NAVIGÁCIA

- Význam videnia pre živé organizmy
- zložitosť spracovania
 - ◆ Klasicky – predsprac., segmentácia, hľadanie vzorov, ...
 - ◆ **Evolučne – vývoj riadenia zviazaný s vývojom spracovania obrazu**
- hardware
- Harvey, 1994 – portálový robot (*gantry robot*)
 - ◆ kamera zavesená z portálového žeriavu riadeného počítačom
 - ◆ obraz cez naklonené zrkadlo + 4 dotykové čidlá
 - ◆ 64 x 64 bodový monochromatický obraz, 50 Hz
 - ◆ fyzická evolúcia
 - ◆ vývoj SAGA – Species Adaptation Genetic Algorithm – premenná dĺžka genotypu, do značnej miery geneticky skonvergovaná populácia
 - ◆ redukovaný vizuálny vstup

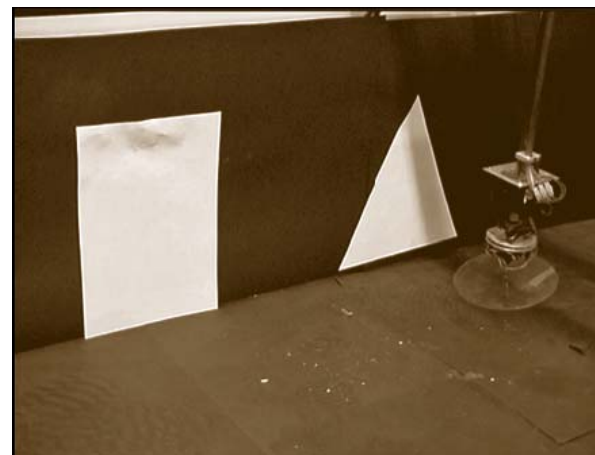
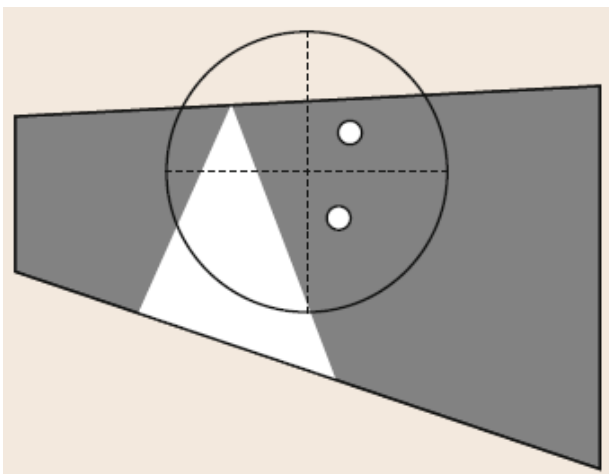
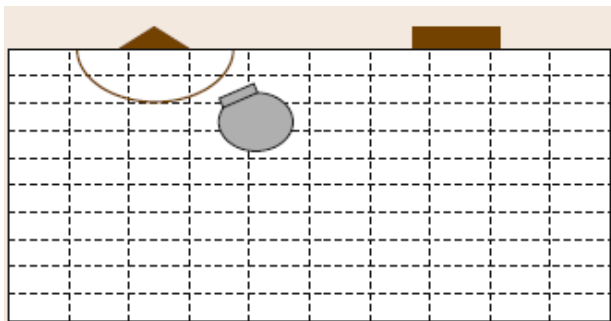
Portálový robot



Portálový robot

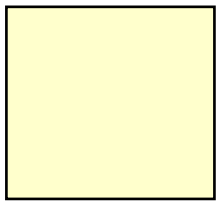


Portálový robot

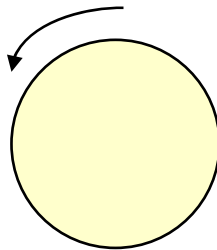


Vizuálna navigácia – senzory a riadenie

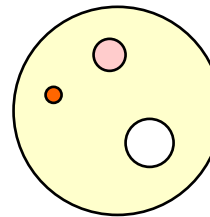
■ digitálny obraz



rotovaný obraz



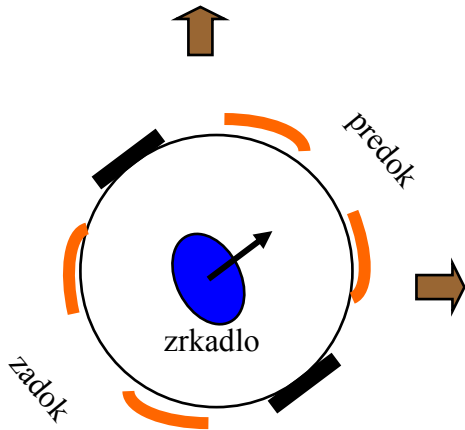
vzorkovanie



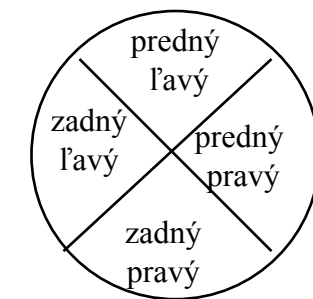
videnie



Recepcné polia - kruhové výseče



virtuálny robot



dotykové senzory

PP	□	0-1
ZP	□	0-1
PL	□	0-1
ZL	□	0-1

údaje z nárazníkov

Vizuálna navigácia – evolúcia riadenia



■ Súčasná evolúcia

◆ Receptčných polí

- ▼ polomer a poloha vzhľadom k stredu
- ▼ z celého poľa za rovnomerne vyberie 25 pixelov, ich jas sa zpriemeruje a prevedie na 4-bitové číslo

◆ Neurónovej siete

- ▼ 7 vstupov
- ▼ 4 výstupy – signál do 1 motora je rozdiel dvoch výstupov
- ▼ počet skrytých neurónov premenlivý
- ▼ neuróny majú oddelené excitačné a inhibičné kanály (viz. výstup)

■ Riešené úlohy

- ◆ Vyhýbanie sa prekážkam
- ◆ Sledovanie pohyblivých cieľov
- ◆ *Rozlišovanie objektov*

Vizuálna navigácia - bootstrapping



- **Úloha: Nájsť biely trojuholník a zostať pri ňom, vyhýbať sa bielému obdĺžniku**
- Premennivé svetelné podmienky
- **Bootstrapping –**
 - ◆ Priama evolúcia neúspešná
 - ◆ Evolúcia navigácie v 3 štádiách s rôznymi fitness funkciami i prostredím

Štádium 1



- Jedna dlhá stena celá biela (150 x 22 cm)
- fitness $\Phi_1 = \sum_{i=1}^{20} Y_i$
- Y_i – vzdialenosť od steny protiľahlej k cieľu meraná 20-krát v priebehu 1 experimentu (25 s)
- každý jedinec testovaný 4-krát z toho istého rohu, ale náhodne otočený
- inicializácia počiatočnej konfigurácie
 - ◆ náhodne sa generovali jedinci a vybraný bol jeden so „zaujímavým“ chovaním a ten sa naklonoval
- asi po 10 generáciách bol nájdený jedinec, ktorý sa najprv rozhliadol okolo seba a potom sa vydal k cieľovej stene

Štádium 2

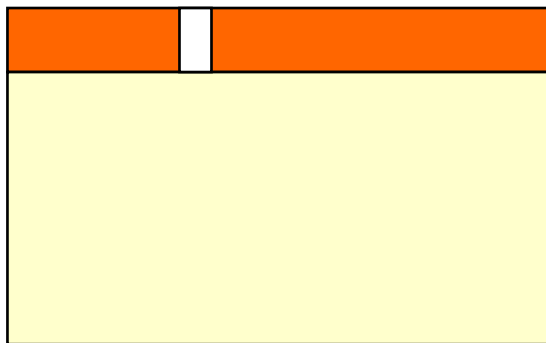
- Cieľ – 22 cm pruh v 2/3 tej istej steny

- fitness $\Phi_2 = \sum_{i=1}^{20} (-d_i)$

- d_i – vzdialenosť robota od stredu pruhu

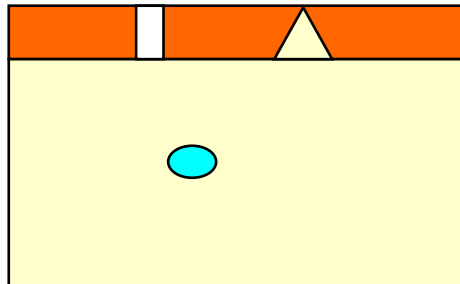
- pokračuje z poslednej generácie Štádia 1

- po 6 generáciách sa našiel jedinec, ktorý našiel cieľ



Štádium 3

- Ciele: rovnoramenný trojuholník a obdĺžnik, oba 21 cm široké, 29.5 cm vysoké
- robot začína v 4 rôznych polohách s náhodnou orientáciou pri protíľahlej stene
- fitness
$$\Phi_3 = \sum_{i=1}^{20} [\beta(D_{1i} - d_{1i}) - \sigma(D_{2i} - d_{2i})]$$
- D_{1i} (D_{2i}) – vzdialenosť trojuholníka (obdĺžnika) od rohu portálu; konštanty
- d_{1i} (d_{2i}) – vzdialenosť robota od trojuholníka (obdĺžnika)



Výsledok



- Extrémne minimalistický systém
 - ◆ 2-3 pixely stačia na spoľahlivé riešenie úlohy
- Hlavná nevýhoda:
 - ◆ Inkrementálna evolúcia – musí navrhnúť experimentátor
- **Re-adaptácia**
 - ◆ ako modifikovať riadiaci systém pri zmenách vonkajších podmienok
 - ◆ 2 úrovne
 - ▼ jedinec – robustnosť a generalizácia NS
 - ▼ populácia – rôznorodosť jedincov v populácii
 - pri zmene podmienok sa dokáže rýchlo adaptovať (pr. vizuálna navigácia – tam sa ešte využívali zásahy človeka, zmeny fitness).
 - Adaptácia pri zmene robota – s pevnou fitness
 - adaptácia na fyzického robota po simulácii evolúcie – s pevnou fitness

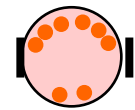
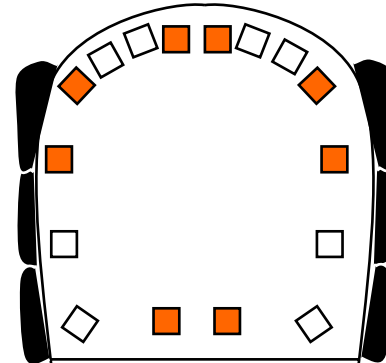
Adaptácia pri zmene robota

- Riadiaci systém pre jednoduchú navigáciu Khepery v bludisku
 - ◆ adaptovaný na prostredie i na morfológiu a senzory Khepery
- Ale Khepera sa nehodí do terénu a je príliš slabá $\Rightarrow$ riadiaci systém bol prenesený na Koalu; zmena senzorov a motorov je tiež zmena prostredia; Koala sa podobá Khepere:
 - ◆ 6 kôl, ale prostredné sú nižšie uložené
 - ◆ infra senzory majú väčší dosah (50 cm), ale má ich 16
 - ◆ software je totožné
- Koalu umiestnili do zväčšeného bludiska pre Kheperu; ako vstup vzali iba 8 zo 16 infra senzorov

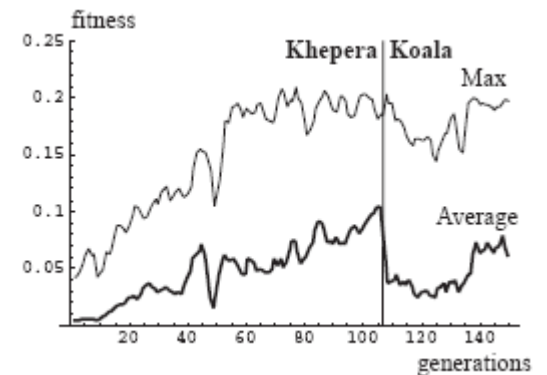


Zmena robota

- 106 generácií na Kheperu
- ďalej do 150. generácie na Koale



- po prenose na Koalu fitness poklesla, ale rýchle sa vrátila na úroveň poslednej generácie pre Kheperu



Simulácia → realita

■ Výhody simulácie

- ◆ rôzne experimentálne podmienky
- ◆ rôzne fitness funkcie
- ◆ 100 gen. 1 hod., na reálnom robote 66 hod.

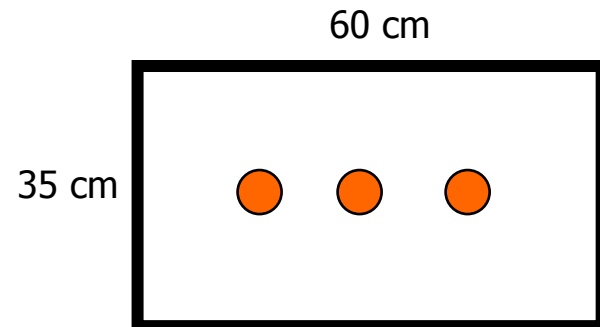
} dá sa toho vyskúšať
viac, než sa prejde na
reálneho robota

■ prechod zo simulácie do reality je tiež krok inkrementálnej evolúcie

■ Pr. Khepera v obdĺžnikovom bludisku s 3 valcami

- ◆ 100 jedincov v populácii
- ◆ 200 generácií simulovaných
- ◆ následne na fyzickom robote
- ◆ 3 experimenty

1. Bez šumu
2. Náhodný šum $< 0 ; 0.4 >$ na senzoroch
3. Konzervatívny šum – senzory dávajú hodnoty akoby bol robot posunutý náhodne $< -30, 30 >$ mm pre x i pre y



Simulácia → realita



■ Prechod na reálneho robota

1. Bez šumu – veľká zmena prostredia – mierny pokles fitness, ale po niekoľkých generáciách návrat na úroveň najlepšieho simulovaného jedinca
2. Náhodný šum – realistickejšia simulácia, ale výsledok podobný variantu bez šumu
3. Konzervatívny šum – žiaden pokles, iba vylepšenie, ešte bližšie realite, pretože
 - simuluje aj odchýlky senzorov spôsobené chybami v motorickom systéme
 - náhodný šum je na susedných senzoroach nezávislý
 - vonkajšie príčiny odchýlok ovplyvňujú pravdepodobne všetky senzory podobne

Záver



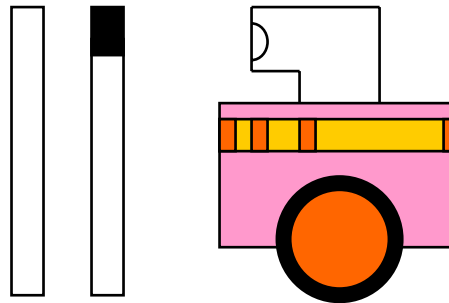
- Umelá evolúcia dokáže vytvoriť riadiaci systém porovnateľný s ručným kódom
- získaný systém je schopný ďalšieho vývinu
- pre zložitejšie úlohy je vhodné použiť **inkrementálny** postup riadený užívateľom, prípadne najprv simuláciu a **doladenie** na reálnom robote
- evolúcia robotov je kontinuálny proces s otvoreným koncom

REAKTÍVNA INTELIGENCIA

- Sensory sú priamo prepojené na motory a na rovnaký stav senzorov systém reaguje rovnako. Ak má systém stavy, tak iba pomocné na určenie motorických akcií – **vtelené a situačné systémy**
- Interná reprezentácia:
 - ◆ klasická umelá inteligencia – explicitná reprezentácia externého sveta
 - ◆ vtelené situačné systémy – čiastočný model sveta, iba tie aspekty, ktoré sú potrebné pre agentov, aby dosiahli svoje ciele
- vtelené a situačné systémy napriek intuícii dokážu riešiť aj zložité úlohy bez vnútorných stavov a bez vnútornej reprezentácie – využívajú koordináciu vnímania a akcií.
- **Senzorovo-motorická koordinácia** – agent môže čiastočne ovplyvniť senzorové signály tým, že vykoná nejakú akciu, ktorá zmení jeho polohu, alebo zmení okolité prostredie.
 - ◆ **ALE: existujú úlohy, na ktoré senzorovo-motorická koordinácia nestačí.**

Problém nejednoznačnosti vnemov

- *Dva (alebo viaceré) objekty generujú tie isté senzorové vnemy, ale vyžadujú rôzne odozvy.*
- Riešenie: vykonať akcie, aby sa našli jednoznačné vnemy
- Pr. Khepera v prostredí s dvomi typmi prekážok, ktoré sa líšia farbou špičky. Objektom s tmavou špičkou sa má vyhýbať, k ostatným sa má priblížiť. Khepera je vybavená 8 infra senzormi a 64-pixelovou lineárnou kamerou s uhlom záberu 36°
 - ◆ keď prekážku nevidí kamerou, tak je vnem nejednoznačný, ale stačí, aby sa otočila a dostala prekážku do zorného poľa – tzv. **aktívne vnímanie**



Aktívne vnímanie



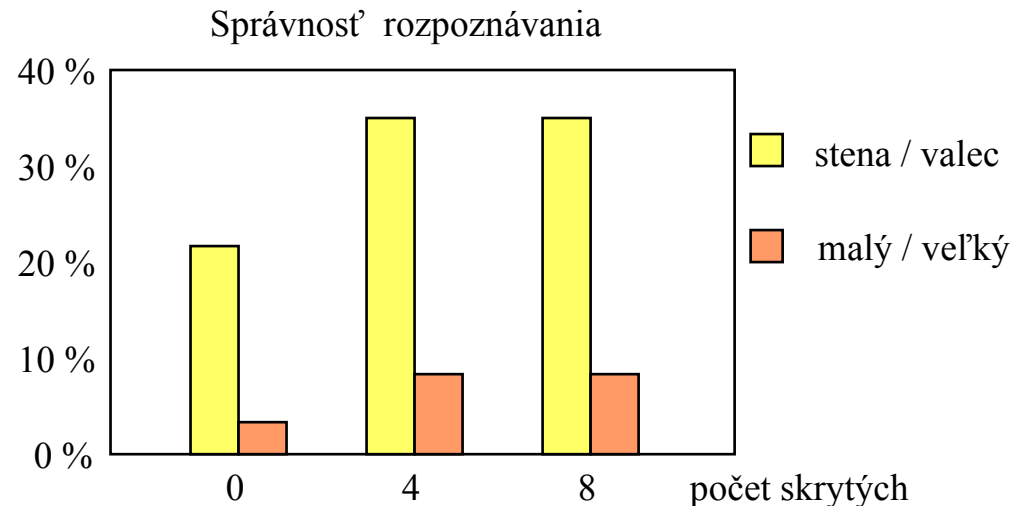
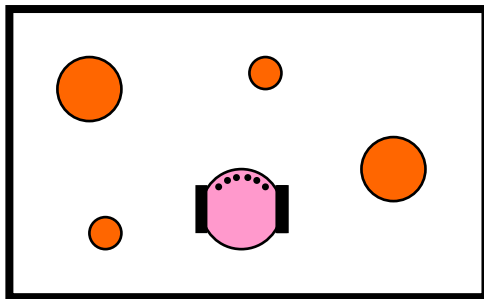
- Rôzne akcie môžu mať rôzne efekty:
 - ◆ niektoré vedú k jednoznačným vnemom, iné nie (ísť rovno, otáčať sa)
 - ◆ niektoré sú efektívnejšie (otáčanie smerom k objektu, alebo vždy tým istým smerom)
- Akcia
 - ◆ čiastočne určuje ako robot plní ciele
 - ◆ čiastočne určuje senzorové vnemy v nasledujúcom kroku
- je možné iba vtedy, keď robot dokáže nájsť aspoň jeden stav senzorov, ktorý je jednoznačný

Senzorovo-motorická koordinácia a zjednodušovanie ťažkých problémov

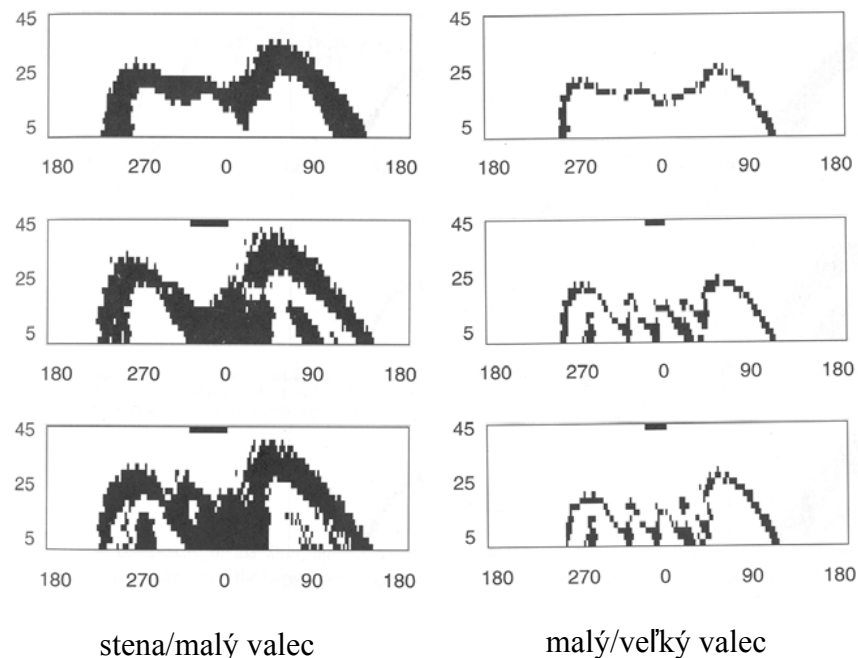
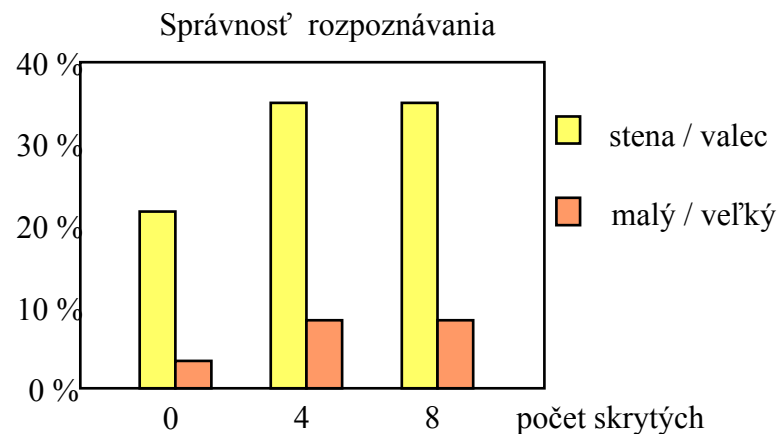
- Skupiny senzorových signálov, ktoré vyžadujú rôzne motorické reakcie sa môžu čiastočne prekryvať
 - ◆ problémy typu 1 (ľahké) – v signáloch zo senzorov je mnoho priamo pozorovateľných príznakov, ktoré umožňujú určiť správnu reakciu
 - ◆ problémy typu 2 (ťažké) – takéto príznaky sú iba slabé, skryté
- Čo s tým?
 1. problémy typu 2 sa niekedy dajú redukovať na problémy typu 1 **prekódovaním stavov** senzorov, aby sa zosilnili užitočné príznaky
 - ▼ napr. neurónová BP-sieť sa najprv učí na jednoduchšej podúlohe až potom na zložitej úlohe; prvá fáza ovplyvní kódovanie signálov vo vnútornej reprezentácii
 2. aktívna senzorovo-motorická koordinácia (je možná iba pre vtelené a situačné systémy)
 - ▼ napr. Khepera sa má približovať k veľkým (5,5 cm) a vyhýbať sa malým (2,3 cm) valcom; používa iba predných 6 infra senzorov

Pasívne rozpoznávanie malých a veľkých valcov

- 6 vstupných neurónov, 1 výstupný (malý/veľký)
 - ◆ a) bez skrytých neurónov
 - ◆ b) skrytá vrstva – 4 neuróny
 - ◆ c) skrytá vrstva – 8 neurónov
- tréningová množina 2 (objekty) x 20 (vzdialeností) x 180 uhlov = 7200 vzorov



Pasívne rozpoznávanie malých a veľkých valcov



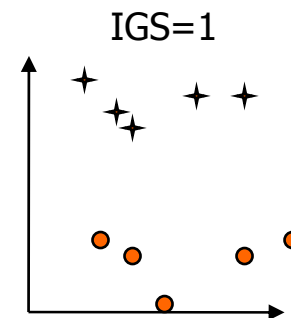
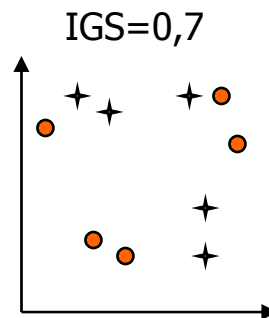
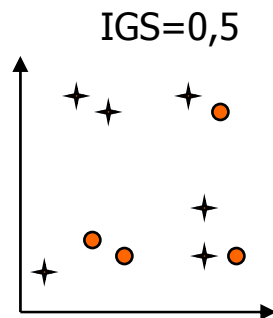
- Vzájomné polohy Khepery a objektu, pre ktoré NS správne rozlíšila malý valec od steny
 - ♦ obr. nesymetrické – senzory nie sú identické
 - ♦ VÝSLEDOK: **pasívna sieť** nie je schopná rozlíšiť objekty (problém typu 2)

Pasívne x aktívne vnímanie

- **Úloha:** priblížiť sa k veľkému valcu a vyhnúť sa malému valcu
- robot sa dokázal pomocou **aktívneho vnímania** naučiť rozlíšiť malé a veľké valce
 - ◆ v 86 % prípadov: robot sa pohybuje, dokiaľ nenarazí na objekt; začne sa okolo neho otáčať a ak je to malý valec, tak ho opustí
 - ◆ v 14% zastaví pred objektom, ale rozlíšenie je horšie
 - ◆ krúživé pohyby vyvolávajú rozdielne signály pre malé a veľké valce
 - ◆ *paradoxne: relatívnych polôh robota a objektu, pri ktorých robot správne rozlíši malý a veľký valec je menej ako u pasívnej siete*

Index geometrickej separability

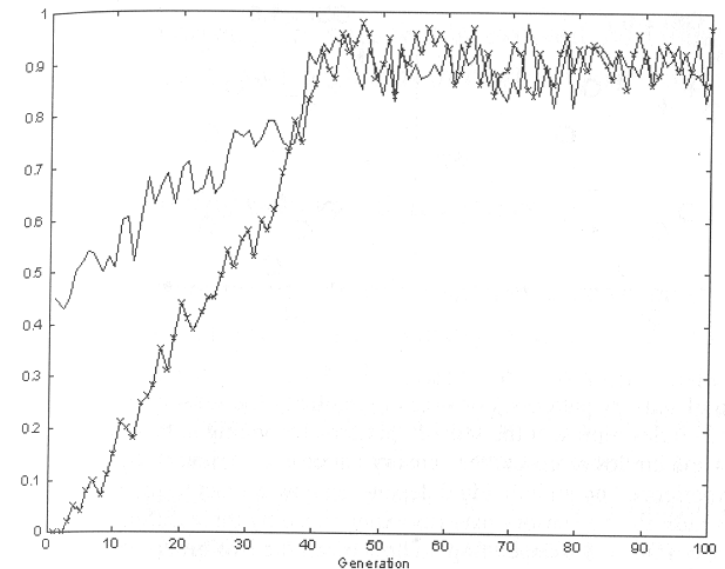
- Zaznamenáme stavy senzorov v N krokoch robota – $x_1, x_2, \dots, x_N$
- pre každý z nich určíme, či najbližší (Euklidovský) vzor z týchto vzorov je z rovnakej triedy, ak áno pripočítame 1, inak 0. Výslednú hodnotu vydáme N .



- Pre pasívnu sieť
 - ◆ IGS stena/malý valec – 0,56
 - ◆ IGS malý/veľký valec – 0,50
- Pre robota
 - ◆ IGS okolo 0,9

Index geometrickej separability

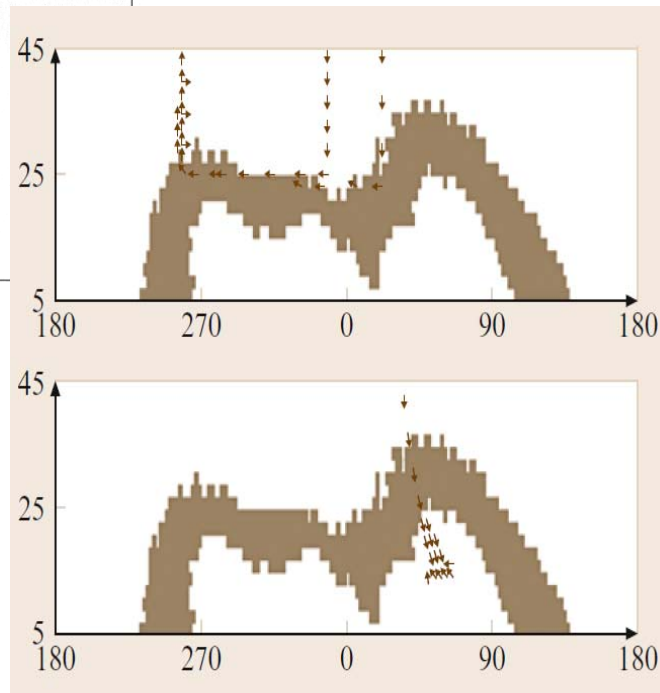
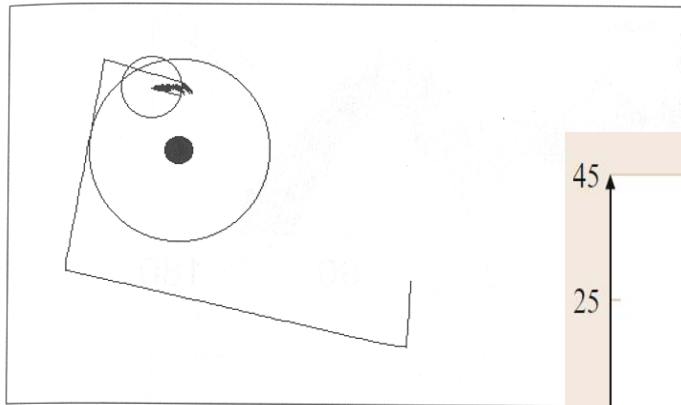
- Zaznamenáme stavy senzorov v N krokoch robota – $x_1, x_2, \dots, x_N$
- pre každý z nich určíme, či najbližší (Euklidovsky) vzor z týchto vzorov je z rovnakej triedy, ak áno pripočítame 1, inak 0. Výslednú hodnotu vydelíme N .
- Pre robota
 - ◆ IGS okolo 0,9



Riešenie pomocou atraktora správania

- Úloha: Khepera má rozlíšiť steny od malého valca a zostať pri valci
 - ◆ aréna 60 x 35 cm, náhodne umiestnený valec (2,3 cm)
 - ◆ NS: 6 vstupných neurónov (infra), 2 výstupné (motory)
- vyvinutí jedinci nekrúžia okolo, ale posúvajú sa vpred/vzad alebo doľava/doprava, pritom zostávajú pod určitým uhlom a v určitej vzdialenosti
- pozícia, na ktorej začne robot oscilovať, je atraktor a robot sa do nej dostane nezávisle na tom pod akým uhlom sa k cieľu priblížil
- IGS je stabilizované asi na 0,8 – tzn. že rozlíšenie cieľa závisí menej na schopnosti aktívne vyberať signály, ktoré sú ľahko rozlíšiteľné
- riešenie je založené na takých reakciách na signály zo senzorov, ktoré spôsobia vytvorenie atraktoru v blízkosti valca, ale nie v blízkosti steny

Atraktor správania



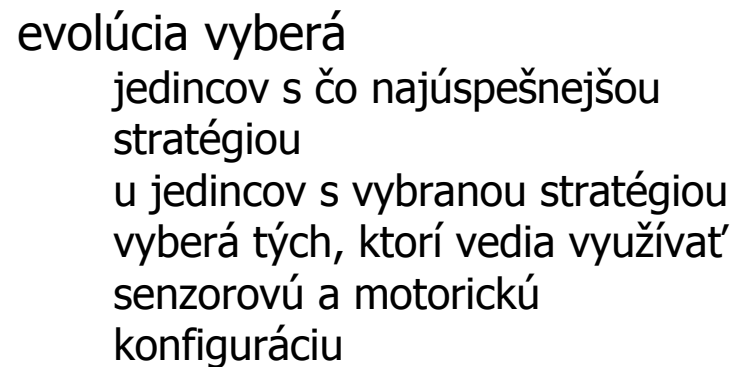
Pohyb khepery
blízko steny

Pohyb khepery
blízko valca

Ďalší atraktor správania

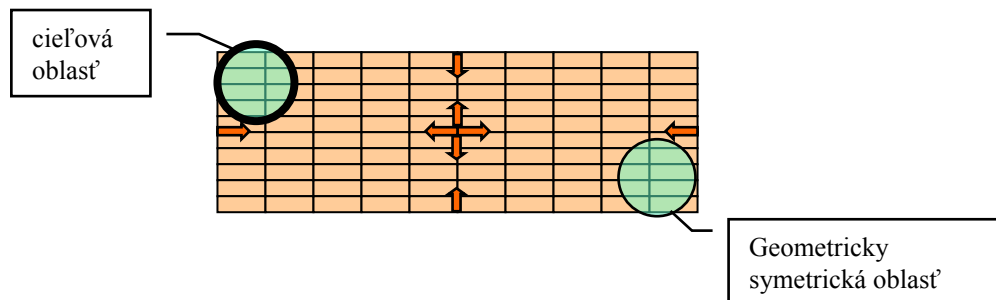
■ Simulovaný agent

- ◆ v kruhovom páse zo 40 buniek (20 vľavo + 20 vpravo);
- ◆ bunky na oboch stranách náhodne očíslované 0-19;
- ◆ agent je v jednej bunke a na vstupe má číslo bunky, v ktorej sa nachádza
- ◆ môže sa pohnúť o 1 bunku v smere/proti smeru hodin. ručičiek
- ◆ cieľ agenta je zostať v ľavej polovici
- ◆ NS – 20 vstupov – číslo bunky (práve jeden neurón má 1); 1 binárny výstup (v smere/proti smeru); váhy iba 0/1
- ◆ vnemy z ľavej a pravej polovice sa úplne prekrývajú – senzorovo-motorická koordinácia nestačí
- ◆ riešenie: v ľavej časti sa vyvinú atraktory



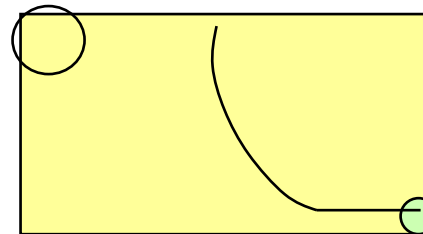
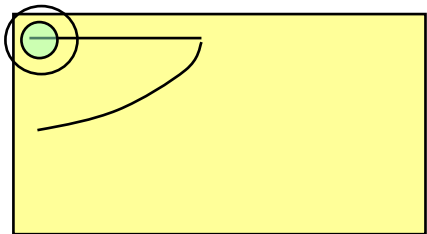
Využívanie obmedzení

- Klasický inžiniersky prístup: prostredie a štruktúra (tela) robota sú obmedzenia a riešenie je treba postaviť na základe modelu prostredia a tela robota podľa obmedzení
- umelá evolúcia vidí obmedzenia ako príležitosti
- Pr. krysy hľadajú potravu; najprv odkrytú, potom na tom istom mieste zahrabanú, bludisko náhodne otočené
 - ◆ 35% krýs hľadalo potravu správne, 31% sa pozeralo na symetricky opačný kút – tzv. **rotačná chyba**
 - ◆ niekoľko autorov ukazovalo, že krysy museli mať postavenú mapu prostredia a používať ju pre navigáciu



■ Khepera

- ◆ začínala iba z vyznačených 8 pozícií
- ◆ NS s 8 vstupnými neurónmi (infra) a 2 výstupnými neurónmi (motory)
– *čiste reaktívne riadenie*
- ◆ podobne ako krysy v 41% prípadoch Khepera našla správne miesto, v 41% urobila rotačnú chybu, v 18% oblasť nenašla
- ◆ princíp: rozlíšenie dlhej a krátkej steny; stačí, aby robot sledoval dlhú stenu a v 50% nájde správne miesto, v 50% urobí iba rotačnú chybu
- ◆ Khepera nemôže rozlíšiť dlhú a krátku stenu! Pomôže senzoro-motorická koordinácia – **pohybuje sa po zakrivenej dráhe a nutne narazí na dlhú stenu**
- ◆ *NEZNAMENÁ to, že krysy nepoužívajú mapu*



Reaktívne riešenie



- Keď sa aréna zväčšila 2x, tak v 2 prípadoch (z 8) Khepera iba jazdila v kruhu
- keď sa aréna 2x zmenšila, tak v 2 prípadoch (z 8) Khepera narazila na a sledovala kratšiu stenu
- Čo sa stane, keď prostredie nebude tak obmedzené?
 - ◆ Khepera štartuje z náhodného miesta v náhodnej orientácii
 - ◆ Veľkosť arény premenlivá (pomer zachovaný) – 25% chýb
 - ◆ Mení sa i pomer veľkostí strán arény – 33% chýb
- Záver: čím menej obmedzení, tým ťažšia úloha

Diskusia



- Väčšinou v robotickom výskume sa problémy riešia rozložením na jednoduchšie podúlohy, ktoré sa implementujú do samostatných modulov; napr. v metódach založených na správaniach sa robí dekompozícia na úrovni správaní
 - ◆ a) rozložiť správanie na elementárne správania
 - ◆ b) implementovať základné správania
 - ◆ c) koordinácia jednotlivých modulov (vrstiev) správaní
- b) a c) sa často dá naučiť (samoorganizácia)
- a) sa robí ručne – **v evolučnej robotike nie je treba**

Diskusia



- Pr. robot, ktorý sa má vyhýbať stenám a zostať pri valci
 - ◆ základné správania:
 - ▼ prehľadávanie prostredia – pohyb
 - ▼ vyhýbanie sa prekážkam
 - ▼ priblíženie a zotrvanie v blízkosti objektu
 - ▼ rozlíšenie objektov – **toto je ale ťažšie než zvyšok**
 - ◆ takáto dekompozícia problém nezjednodušila!
 - ◆ Vo vyššie uvedenom sa evolúciou vytvorilo správanie, ktoré rozlíšenie objektov umožnilo
- rozklad na základné správania je ťažký a umelá evolúcia často nájde riešenie, ktoré človek sotva navrhne
 - ◆ dáva viacej rôznych riešení naraz
 - ◆ často vyberá riešenia založené na vyvinutých správaniach