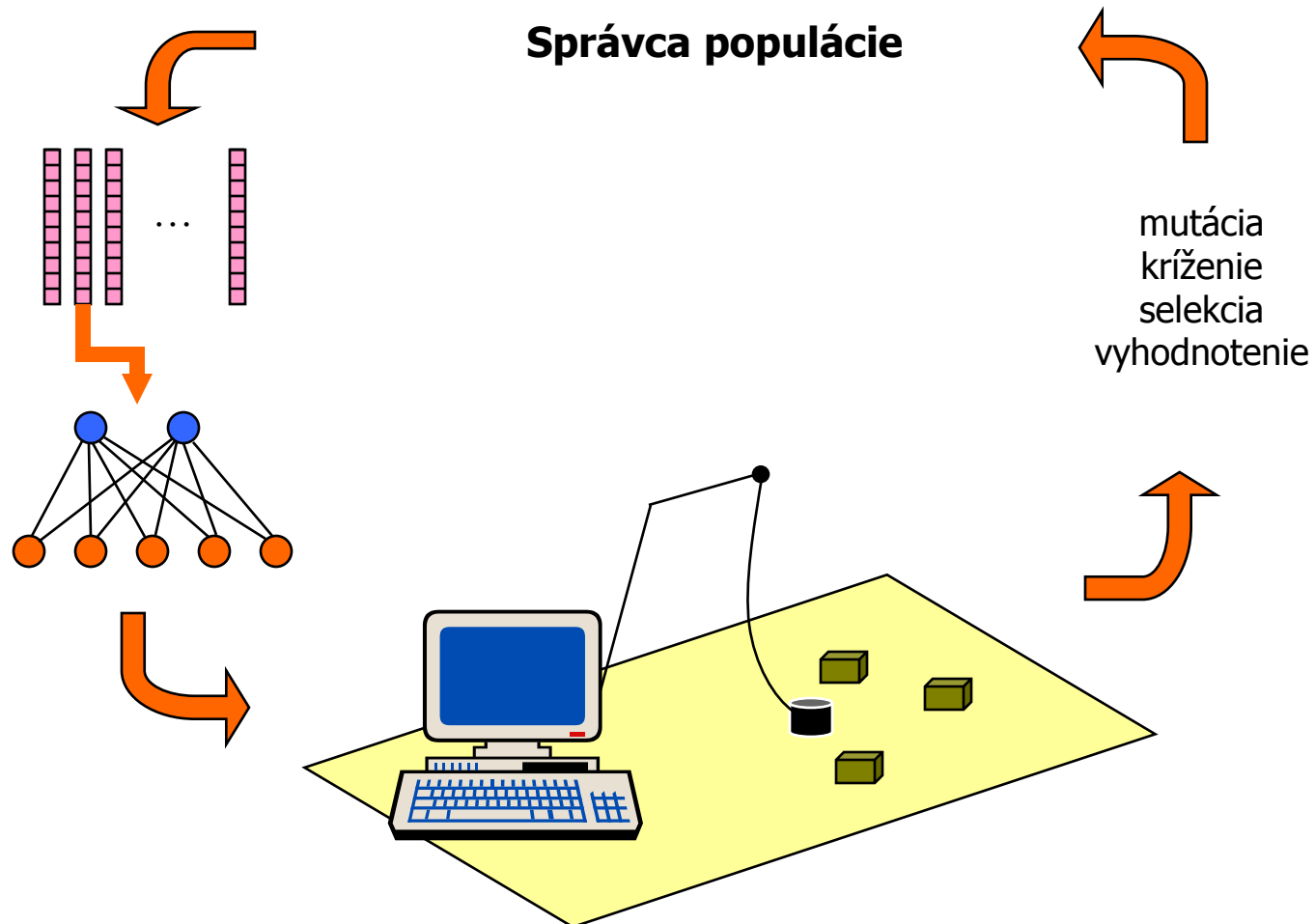


# Schéma evolučného experimentu



# Schéma evolučného experimentu



## ■ Premiestňovanie:

- ◆ pri prepínaní medzi dvomi jedincami v populácii sa robot náhodne premiestni
  - ◆ ak by sa testovaný jedinec dostal do zložitej pozície a túto pozíciu zachováme pre nasledujúceho jedinca, tak bude v nevýhode
- ◆ nie je dobré začínať vždy z toho istého miesta – robot by sa mohol naučiť pevnú postupnosť krokov („minimalistické riešenie!“)

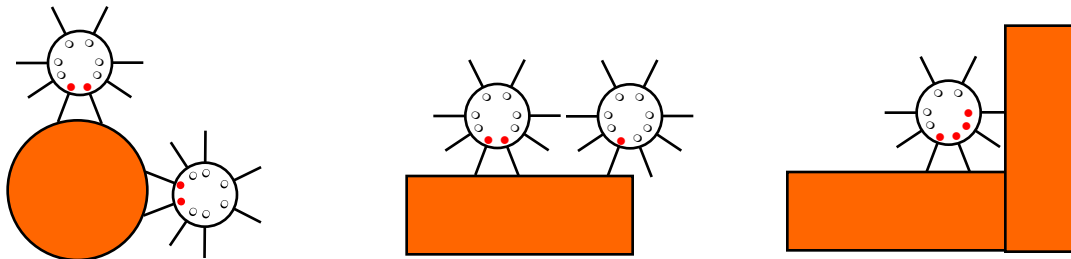
# Simulovaná evolúcia



- Fyzická evolúcia trvá dlho (dni)
- paralelná fyzická evolúcia nemusí pomáhať
  - ◆ roboty, ktoré sú mechanicky i softwarovo zhodní, nemusia fungovať totožne
- Najprv simulovaná evolúcia a potom najlepších jedincov vyskúšať na reálnych robotoch
  - ◆ nezrýchľuje vždy – napr. simulácia videnia
  - ◆ bez overovania na fyzickom robote vôbec nemusí fungovať
  - ◆ za určitých podmienok to ide, ale v zložitejších prostrediach sa to zatiaľ nevie
  - ◆ simulácia zjednodušuje – dodávku energie, „resetovanie“ stavu prostredia
    - ◆ pr. „Khepera čistič“
- 2 metódy simulácií
  - ◆ čo najvyššia vernosť
  - ◆ modelovanie iba podstatných aspektov

# Presné modelovanie interakcie robota s prostredím I

- *Senzory a motory i keď sú identické môžu fungovať rozdielne*
  - ◆ A) vzorkovanie rôznych objektov z rôznych vzdialeností a uhlov na reálnom robote
    - ◆ napr. aréna so stenami a rovnako veľkými cylindrami – 180 orientácií x 20 vzdialeností – *pre každý senzor zvlášť*
    - ◆ problematické v zložitejšom prostredí
      - robot blízko dvoch objektov – napr. skladaním vektorov aktivít senzorov



# Presné modelovanie interakcie robota s prostredím II

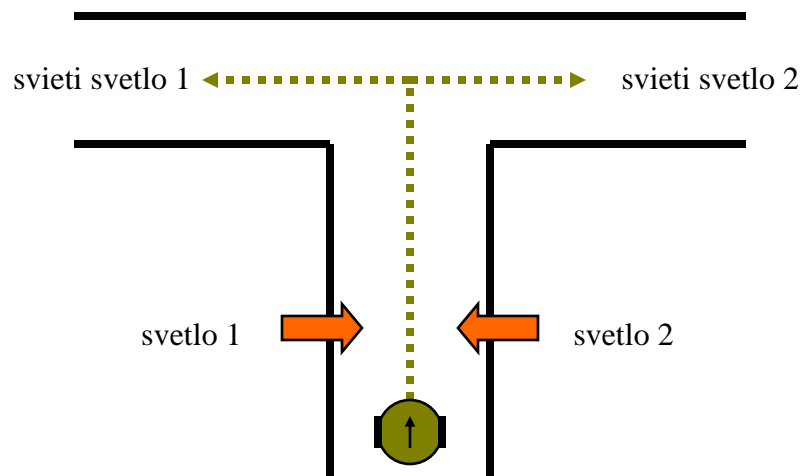
- ◆ B) matematický model, ale parametre funkcií sa odmerajú na reálnom robote
- *Fyzické senzory dávajú neisté hodnoty, výkonné prvky majú neistý efekt*
  - ◆ napr. teplota prostredia, doba práce
  - ◆ rieši sa pridaním šumu
    - A. náhodný šum – je treba odhadnúť interval
      - riadenie neurónovou sieťou občas **spolieha** na šum v určitom rozsahu
    - B. náhodný, ale systematicky menený (napr. rozsah)
- *Telo robota a charakteristiky prostredia musia byť presne reprodukované*
  - ◆ často nestačí mriežka ako model prostredia

# Minimálna simulácia

- dôležité charakteristiky interakcie robota s prostredím – tzv. *základná množina charakteristík* – je treba modelovať presne
- simulácia má okrem nich ďalšie aspekty, ktoré nekorešpondujú s realitou – tzv. *implementačné aspekty*
- implementačné aspekty sa musia meniť od pokusu k pokusu tak, aby riadenie, ktoré by na nich bolo závislé, nefungovalo (tj. záviselo iba na základnej množine charakteristík)
- základná množina charakteristík sa musí meniť od pokusu k pokusu tak, aby riadenie, ktoré je na nich závislé, bolo robustné (tj. len také variácie, s ktorými si riadenie poradí)
- príklad – Jacobi (1997) Khepera v bludisku tvaru T

# Khepera v bludisku tvaru T

- Na križovatke má zahnúť na tú stranu, kde svieti svetlo v hlavnej chodbe



## Základná množina charakteristík:

- pohyb kolies
- aktivácia detektorov vzdialenosti v chodbe
- aktivácia detektorov rozptýleného svetla v chodbe

## Implementačné aspekty:

aktivácia detektorov vzdialenosti v križovatke – **vôbec sa nesimulovala**

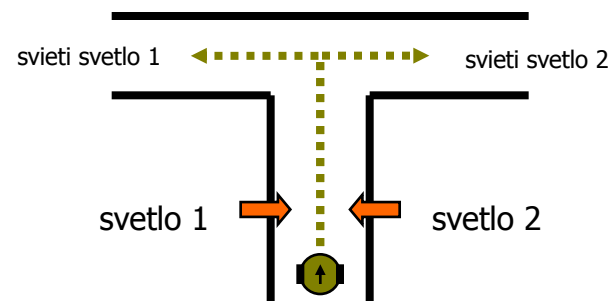
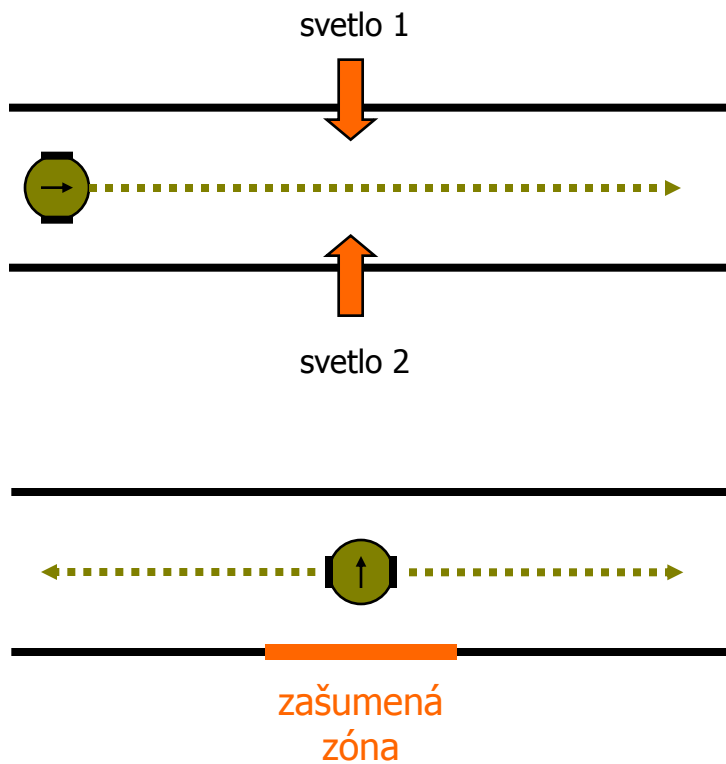
# Khepera v bludisku tvaru T



- Bludisko modelované dvomi segmentmi
  - ◆ hlavná chodba so svetlami – keď robot došiel na jej koniec, tak bol prenesený do druhého segmentu a otočený o  $90^\circ$
  - ◆ vodorovná chodba – po prenesení má robot za sebou komplikované napojenie chodieb; aby sa nemohol orientovať na základe senzorov smerujúcich do hlavnej chodby, tak ich hodnoty sa pri opakovaných pokusoch líšili
- aby sa zaručila robustnosť riadenia tak sa menilo:
  - ◆ strana, na ktorej svietilo svetlo
  - ◆ šírka chodby, dĺžka chodby, dĺžka osvetleného úseku chodby – v určitom obmedzenom rozsahu

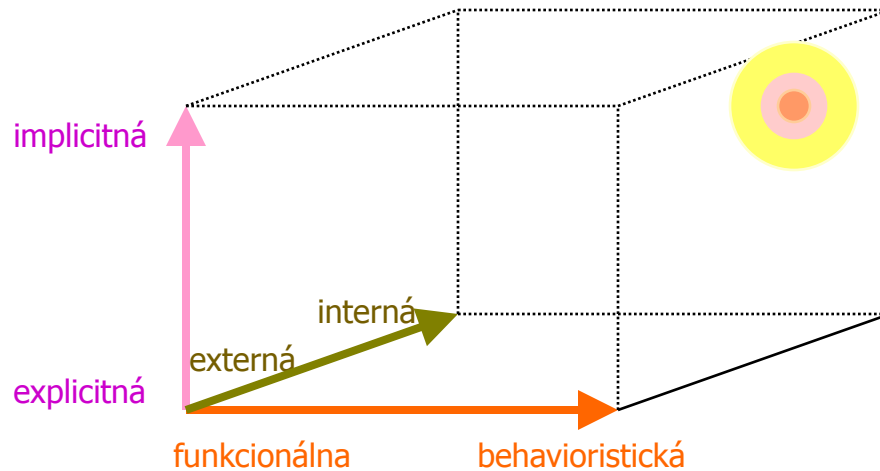


# Khepera v bludisku tvaru T



# Priestor funkcií fitness

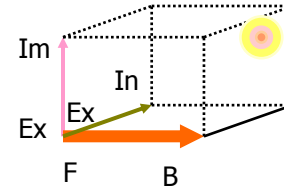
- Fitness funkcia – kľúčový význam
  - ◆ premenné dobre popisujúce funkčný systém môžu byť nanič na začiatku a naopak
  - ◆ malá zmena môže dať úplne iné výsledky
  - ◆ metóda pokusu a omylu – čas
- Priestor fitness funkcií



# Charakteristiky fitness funkcie

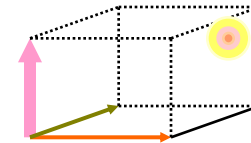
## ■ Funkcionálna x behavioristická

- ◆ Napr. kráčajúci robot
  - ◆ a) frekvencia opakovania
  - ◆ b) prejdená vzdialenosť



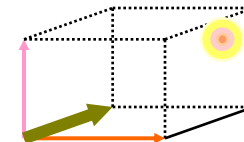
## ■ Explicitná x implicitná

- ◆ počet premenných, konštánt a obmedzení (~dĺžka zápisu funkcie)
  - ◆ a) dosiahnutá rýchlosť, aktivita vybraných neurónov, či dorazil k zdroju energie
  - ◆ b) ako dlho vydržal pracovať



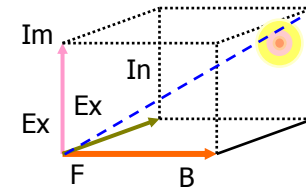
## ■ Externá x interná

- ◆ dá sa merať iba zvonku, alebo je dostupná robotovi z jeho senzorov?
  - ◆ a) absolútna vzdialenosť robota od cieľa
  - ◆ b) množstvo svetla zachytené senzorom, napätie batérií
- ◆ Externá – pri simuláciách „skoro zadarmo“, inak drahá (čas, vybavenie)



# Ako volit' fitness

- Optimalizácia súboru parametrov systému, ktorý je zložitý, ale dobre definovaný **FEE** – **F**unkcionálna, **E**xplicitná, **E**xterná – tradičný „inžiniersky prístup“
- autonómny robot schopný pracovať v neznámom a nepredvídateľnom prostredí **BII** – **B**ehavioristická, **I**mplicitná, **I**nterná – vhodná pre inkrementálny vývoj – „syntéza umelého života“
  - ◆ diagonála FEE - BII
- Subjektívna fitness – vyvíjajúci jedinci sú hodnotení človekom – **BEE**
  - ◆ pr. Biomorphy – človek vyberá jedincov, ktorí budú reprodukovani a mutovaní
  - ◆ môže sa meniť človek od človeka alebo i pre jedného človeka v čase



# Evolúcia jednoduchovej navigácie

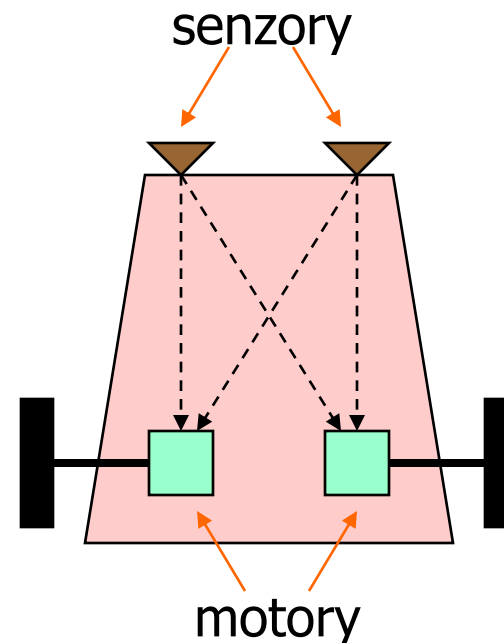
- Prehľad riešených úloh:
  1. Jednoduché správanie
  2. Správanie, ktoré zrejme vyžaduje pamäť alebo vnútorné stavy, ale ukáže sa, že to ide jednoduchšie
  3. Zložitejšia navigácia – modulárna architektúra a vnútorné stavy
- Navigácia vyžaduje zobrazenie zo sensorických údajov na motorické akcie



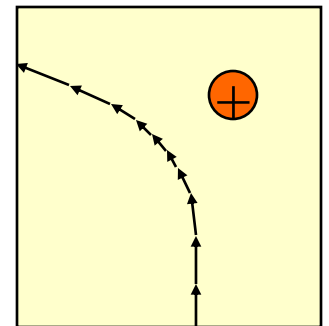
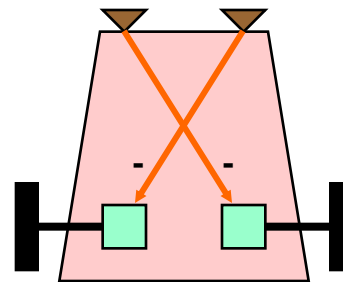
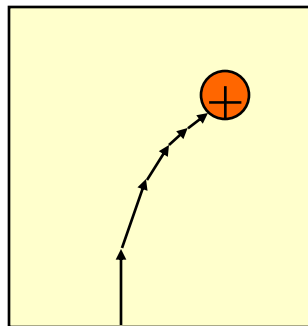
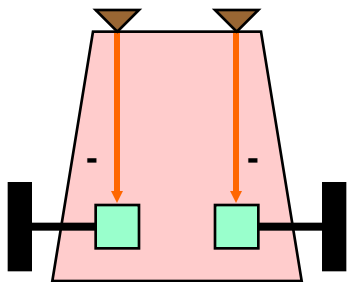
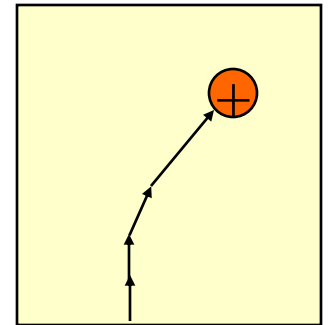
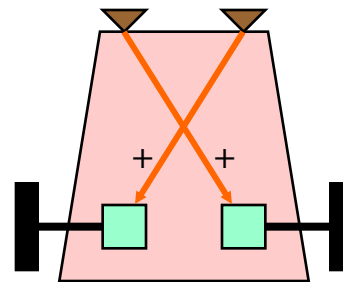
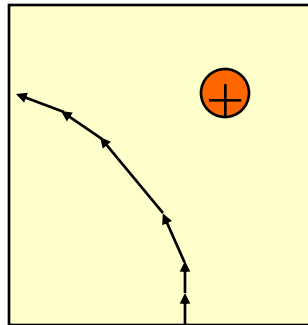
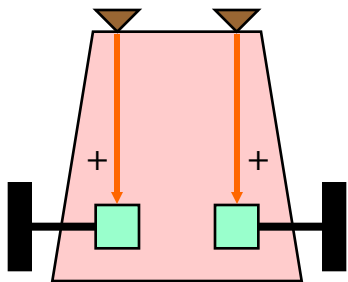
- ◆ vzájomná závislosť
- ◆ často nestačí urobiť tabuľku (stav senzorov → motory) pre všetky možné stavy senzorov
- „umelá evolúcia dokáže vytvoriť veľmi efektívne riadenie s využitím interakcií medzi robotom a prostredím“

# Priamy pohyb s obchádzaním prekážok

- Ak je morfológia robota symetrická – dá sa skúsiť **Braitenbergovo autíčko** (1984)
  - ◆ motory priamo spojené váženými spojmi so senzorami
  - ◆ spoj
    - ◆ pozitívny (+) – excitácia
    - ◆ negatívny (-) – inhibícia

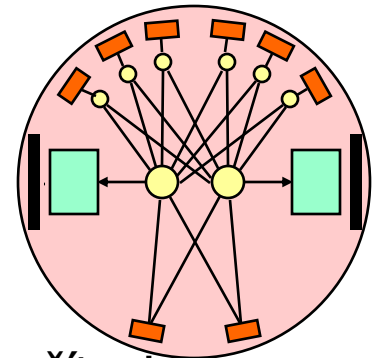


# Braitenbergovo autíčko



# Riadenie Breitenbergovho typu pre Kheperu

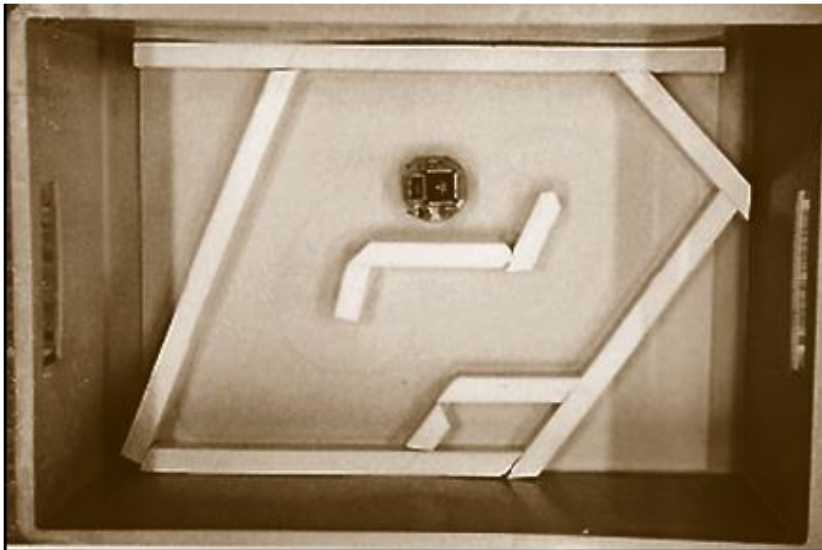
- Každý senzor má excitačný spoj s motorom na svojej strane a inhibičný spoj s motorom na druhej strane; váhy sú symetrické
- vstupy do motorov sú posunuté o kladnú konštantu – vynútený pohyb dopredu
- netriviálne nastavenie
  - ◆ vlastnosti motorov a senzorov (sila spoja)
  - ◆ pomery váh
    - ◆ ak najľavejší a ľavý čelný senzor majú rovnakú váhu
      - ◆ → robot zatača príliš mnoho pri prekážke vľavo
      - ◆ → alebo robot zatača málo pri prekážke čelne
    - ◆ hodnoty váh závisia od rýchlosti pohybu a frekvencie čítania senzorov, odrazivosti prekážok, ...
- rôzne roboty v rôznych prostrediach vyžadujú starostlivé nastavenie





# Evolučný prístup

- Bludisko s okruhom (80 x 50 cm)
- **cieľ**: maximálna rýchlosť vpred, vyhýbanie sa prekážkam



# Evolučný prístup

- Bludisko s okruhom (80 x 50 cm)
- **cieľ'**: maximálna rýchlosť vpred, vyhýbanie sa prekážkam
- **fitness** ( $k$  je číslo kroku,  $n$  je celkový počet krokov)

$$\Phi = \frac{1}{n} \sum_{k=1}^n \Phi_k \quad \Phi_k = V_k \left(1 - \sqrt{\Delta V_k}\right) (1 - i_k)$$

$$0 \leq V, \Delta V, i \leq 1$$

$$V_k = |l_k| + |r_k| \quad \Delta V_k = |l - r|$$

$l_k$  ... rýchlosť ľavého motora,  $l_k \in \langle -0,5 ; 0,5 \rangle$

$r_k$  ... rýchlosť pravého motora,  $r_k \in \langle -0,5 ; 0,5 \rangle$

$i_k$  ... normalizovaná hodnota najaktívnejšieho infračerveného čidla

# Fitness

$$\Phi = \frac{1}{n} \sum_{k=1}^n \Phi_k \quad \Phi_k = V_k \left(1 - \sqrt{\Delta V_k}\right) (1 - i_k)$$

$$0 \leq V, \Delta V, i \leq 1$$

$$V_k = |l_k| + |r_k| \quad \Delta V_k = |l - r|$$

$l_k$  ... rýchlosť ľavého motora,  $l_k \in \langle -0,5 ; 0,5 \rangle$

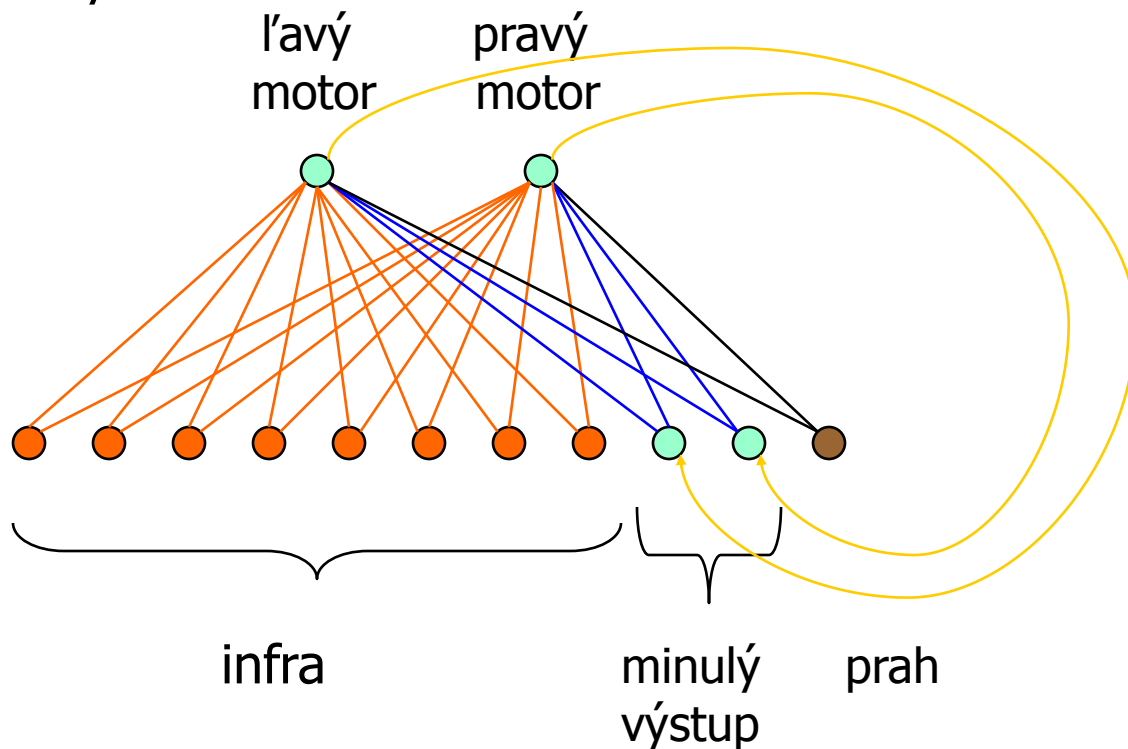
$r_k$  ... rýchlosť pravého motora,  $r_k \in \langle -0,5 ; 0,5 \rangle$

$i_k$  ... normalizovaná hodnota najaktívnejšieho  
infračerveného čidla

- $V$  maximalizuje rotáciu kolies (4 maximá)
- $(1 - \sqrt{\Delta V})$  maximalizuje rotáciu rovnakým smerom
- $(1 - i)$  posilňuje vyhýbanie sa prekážkam
  - ◆ nepreferuje pohyb dopredu!

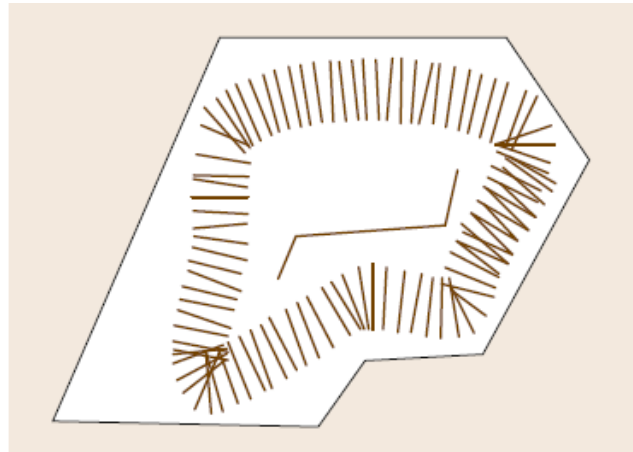
# Riadenie

bez skrytých neurónov, 8 vstupných (infra,  $\langle 0,1 \rangle$ ),  
2 výstupné (motory, sigmoida posunutá do  $\langle -0.5,0.5 \rangle$ ),  
2 neuróny na uchovanie stavu motorov do ďalšieho cyklu (Elman)  
výstupné neuróny majú prahy



# Evolúcia

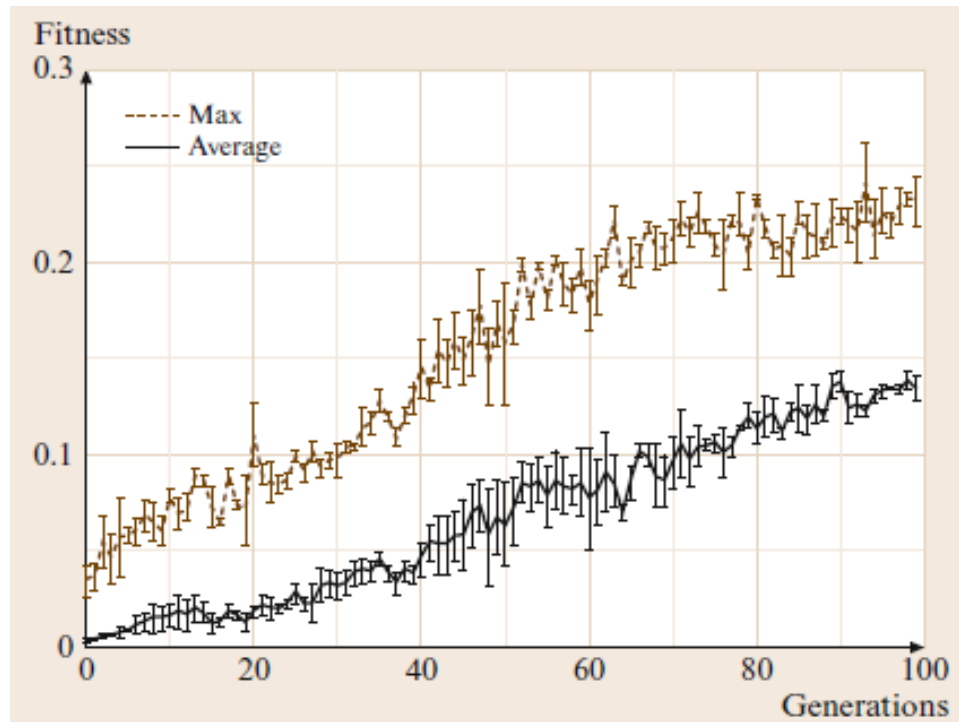
- 80 jedincov v populácii
- Váhy a prah inicializované na malé čísla okolo 0
- Infrsenzory snímané každých 300 ms
- 1 experiment 80 senzoro-motorických cyklov ( $\approx 24$  s)



Najlepší jedinec na konci evolúcie

# Sledovanie vývoja

- Priemerná a maximálna fitness v populácii



Fitness najlepšieho jedinca v populácii a priemerná fitness populácie v jednotlivých generáciách. Zvislé úsečky ukazujú smerodajnú odchýlku z troch behov s rôznymi počiatočnými populáciami.

# Sledovanie vývoja



- Oddelene jednotlivé zložky fitness (str. 76)
- pre Breitenbergovo autíčko (so symetrickými váhami) existujú rovnovážne body, keď sa vstupy zo symetrických senzorov vzájomne anulujú – autíčko sa zastaví
- tento problém riešia rekurentné spoje (zle sa nastavujú ručne)

# Ďalšie vyvinuté vlastnosti

- Teoreticky by robot mohol cúvať, ale vždy preferoval smer, kde má viacej senzorov
  - ◆ nerozvinulo sa v pravouhlom bludisku bez prekážok
- maximálna rýchlosť vyvinutých jedincov nikdy nedosiahla maximum 80 mm/s. ale len asi 48 mm/s – pravdepodobne preto, že senzory sú snímané každých 300 ms a pri vyššej rýchlosti by robot narážal do prekážok, ktoré nestačil zaregistrovať. Maximálna rýchlosť nevzrástla ani pri dojnásobnom počte generácií.

