

StarLogo

Evolučné algoritmy

Martin Komara
Jozef Kriška

Obsah

- Agentové modelovanie
- História
- Entity v StarLogu
- Príklad: termity
- Používateľské rozhranie

Agentové modelovanie

- Agentové modelovanie je hlavná technika na študovanie **komplexných adaptívnych systémov** (CAS).
- „Rozdeľuj a panuj“ – redukcia komplexného problému na čiastkové problémy a skúmanie na úrovni týchto podproblémov.
- Interakcie medzi jednotlivými agentmi spôsobujú, že komplexný systém sa vyznačuje „emergentnými“ vlastnosťami.

História

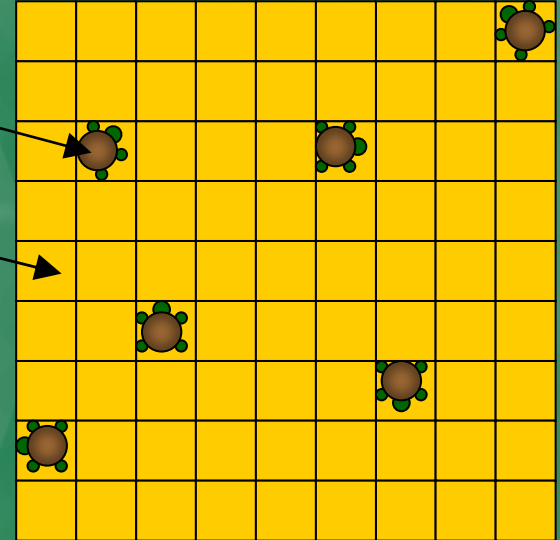
- 90-te roky - *Logo na Connection Machine (MIT, 16 384 procesorov po 8MHz)
- 1994 – MacStarLogo na Mac IIfx (25MHz)
- 1999 – StarLogo v Java

Programovací jazyk Logo

- Vyvinutý v 60-tych letech pány Feurzeigom a Papertom
- Založený na programovacím jazyku LISP
- Jednoduchá syntax
- Prvky přirozeného jazyka

Entity v StarLogu

- Korytnačky (turtles)
- Dlaždice (patches)
- Pozorovateľ (observer)



Korytnačky

- Korytnačky predstavujú agentov, žijúcich na mriežke tvorenej dlaždicami.
- Korytnačky môžu reprezentovať ľubovoľný objekt (napr. mravec v kolónii, auto v dopravnej zápche, molekulu v plyne atď.)
- Majú 4 základné vlastnosti: pozícia na mriežke, natočenie, farbu, pozícia pera (hore / dole).
- Možnosť definovať ďalšie vlastnosti.

Dlaždice

- Dlaždice tvoria svet, v ktorom sa pohybujú korytnačky.
- Sú usporiadané do mriežky.
- Môžu vykonávať príkazy StarLoga a pôsobiť na korytnačky a dlaždice v ich blízkosti.

Pozorovateľ

- Existuje len jediný pozorovateľ.
- Pozoruje a riadi korytnačky a dlaždice.
- Vytvára nové korytnačky.
- Zbiera štatistické údaje o svete.
- Maže mriežku.

Príklad: termity

- Termity zbierajú drevené triesky a premiestňujú ich. Riadia sa pritom jednoduchými pravidlami:
 - Termit náhodne putuje
 - Ak narazí na triesku, zdvihne ju.
 - Hľadá ďalšiu triesku, ku ktorej ju položí.
- Tento cyklus sa opakuje

Termity: inicializácia

```
to setup
  ca
  crt number
  ask-patches [if (random 100) < density [setpc yellow]]
  ask-turtles [setup]
end
```

Termity: inicializácia termitov

```
to setup
  setshape termite-shape
  seth random 360
  jump random 200
end
```

Termity: hlavní cyklus

```
to go
```

```
  search-for-chip
```

```
  find-new-pile
```

```
  find-empty-spot
```

```
end
```


Termity: hľadanie triesky

```
to search-for-chip
  if pc = yellow
    [stamp black
     setshape termite-wood-shape
     jump 20
     stop]
  wiggle
  search-for-chip
end
```

```
to wiggle
  fd 1
  rt random 50
  lt random 50
end
```

```
to go
  >search-for-chip
  find-new-pile
  find-empty-spot
end
```

Termity: hľadanie ďalšej triesky

```
to find-new-pile
  if pc = yellow [stop]
  wiggle
  find-new-pile
end
```

```
to go
  search-for-chip
  >find-new-pile
  find-empty-spot
end
```

Termity: hľadanie voľného miesta

```
to find-empty-spot
  if pc = black
    [stamp yellow
     setshape termite-shape
     get-away
     stop]
  seth random 360
  fd 1
  find-empty-spot
end
```

```
to get-away
  seth random 360
  jump 20
  if pc = black [stop]
  get-away
end
```

```
to go
  search-for-chip
  find-new-pile
  >find-empty-spot
end
```

Termity: priebeh

