

# Genetické algoritmy

## Algoritmus pro matchování hvězd

Filip Hroch

ÚTFA, MU, Brno

29. říjen 2007

# Co to jsou GA?

## Genetické algoritmy (GA)

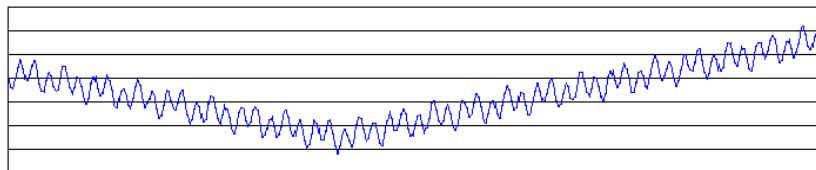
- ▶ řeší problémy na základě principů evoluce
- ▶ nejsou deterministické
- ▶ obvykle hledají extrémy vhodných funkcí

## Historie

- ▶ I. Rechenberg (1960) — uvedl na svět ideu GA,
- ▶ J. Holland (1975) — rozpracování GA
- ▶ J. Koza (1992) — nazval svou metodu GA

# Příklad použití GA

## Hledání globálního minima



- ▶ pravděpodobnost nalezení minima Newtonovou metodou, zlatým řezem apod. je malá ...

# Biologické pozadí

## Genetická informace v biologii

- ▶ v buňkách živých organismů nesou genetickou informaci chromosomy
- ▶ chromosomy jsou makromolekuly obsahující DNA
- ▶ DNA je dvojšroubovice z mnoha proteinu reprezentujících geny
- ▶ všechny chromozomy v buňce tvoří dohromady genom

## Rozmnožování při dělení probíhá

- ▶ rekombinace (crossover), dvě DNA se rozpletou a spojí
- ▶ mutace, při rekombinaci se občas některé geny změní

# Princip GA

## Algoritmus GA

- ▶ Vytvoření náhodné populace (množina chromosomů, každý obsahuje několik genů, reprezentace 2D polem).
- ▶ Necháme populaci žít:
  1. Náhodně vybíráme chromosomy podle schopnosti žít.
  2. Křížíme je.
  3. Občas je zmutujeme.
  4. Počítáme vhodnost populace pro naše účely (fitness funkce).
  5. Pokud nám vlastnosti populace vyhovují, tak skončíme. Jinak opakujeme od 1.

# Vlastnosti GA

## Jak fungují?

- ▶ křížení hledá cestu k minimu
- ▶ mutace vyvádí z lokálního minima
  
- ▶ jde o velmi obecný algoritmus
- ▶ volba fitness funkce a pravděpodobností křížení a mutací
- ▶ vyhledají globální extrém a neskončí v lokálním
- ▶ extrém není obvykle přesně lokalizován (v porovnání s Newtonem)
- ▶ nevdí měnící se podmínky během minimalizace
- ▶ není jasné, proč vlastně fungují (důkaz od J.Hollanda)

# Parametry pro GA

## Fitness funkce

- ▶ reprezentuje problém (hledáme extrém této funkce)

## Pravděpodobnost křížení

- ▶ volíme kolem 80%, ovlivňuje rychlost vývoje (konvergence)

## Pravděpodobnost mutací

- ▶ zabraňuje sklouznutí do lokálního minima
- ▶ obvykle volíme kolem pár procent

## Velikost populace

- ▶ pokud zvolíme malou tak prohledáváme jen část prostoru
- ▶ pokud větší než je hranice, pak zbytečně ztrácíme čas

# Křížení a mutace

## Křížení

- ▶ náhodně rozdělujeme jednotlivé křížené chromosomy a pak spojujeme do nových
- ▶ ruská ruleta, náhodný výběr s ohledem na vhodnost ke křížení
- ▶ podle hodnoty, selekce se bere podle vhodnosti k dalšímu vývoji
- ▶ vyhození looseru, nejhorší chromozomy se nahrazují novými kříženci
- ▶ elita, přednost mají elitní chromosomy

## Mutace

- ▶ náhodně zvolené geny v chromozomu prohodíme



# Užitečné použití

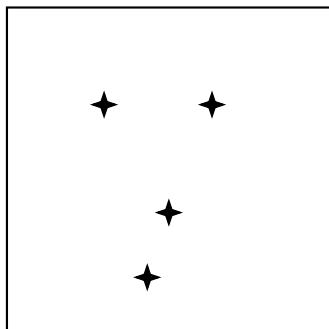
## Použití při minimalizaci funkcí

- ▶ volíme řešení v určitém intervalu  $a \dots b$
- ▶ ten rozdělíme na dílky a přiřadíme jim pořadová čísla
- ▶ jako chromosomy pak použijeme jejich 8-bit nebo 16-bit reprezentace
- ▶ po lokalizování extrému můžeme výsledek zpřesnit obvyklou minimalizační metodou

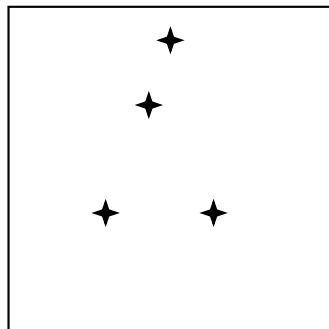
## Řešení astronomických problémů

- ▶ počáteční parametry pro světelné křivky: FOTEL, WD, ...
- ▶ hledání parametrů profilů galaxií
- ▶ testování modelů hvězdných atmosfér

# Hledání hvězd



no.	$x_1$	$y_1$
1	7	3
2	8	6
3	5	10
4	10	10



no.	$x_2$	$y_2$
4	5	5
3	10	5
2	7	10
1	8	13

## Match problém

- ▶ hledání takového uspořádání jedné množiny vůči druhé, aby byly ekvivalentní prvky na stejné pozici
- ▶ princip řešení je prostý: stačí najít všechny možné permutace jedné množiny a funkci, která pozná správnou permutaci
- ▶ problémem je počet všech možných permutací:

$n$     $n!$

5   120

10   3628800

15   1307674368000

20   2432902008176640000

25   15511210043330985984000000

30   265252859812191058636308480000000

- ▶ potřebný počet hvězd běžné počítače v reálném čase nezvládají
- ▶ GA umožňují podstatným způsobem redukovat počet permutací, protože zahazují předem ztracené možnosti

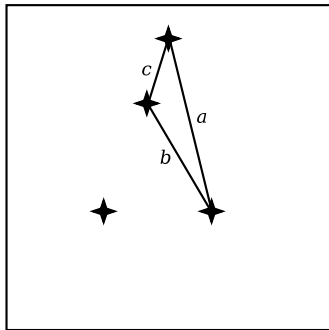
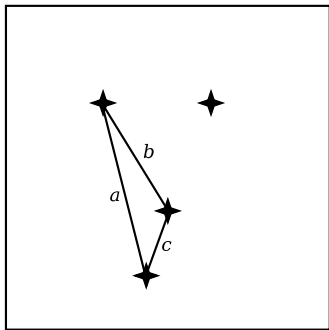
# Geometrické transformace

Uvažujeme následující:

- ▶ posunutí
- ▶ otočení
- ▶ změna měřítka
- ▶ zrcadlové převrácení

$$\mathbf{r}_2 = c(\mathbf{r}_0 + \mathbf{M}_{\pm} \mathbf{r}_1)$$

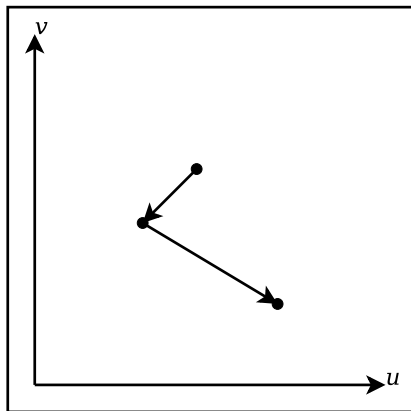
# Trojúhelníková podobnost



$$u = \frac{b}{a}, \quad v = \frac{c}{a}$$

$$a \geq b \geq c$$

# Fitness funkce



- ▶ součet rezidui mezi dvěma cestami v  $u - v$  prostoru

$$S = \sum [(u_1 - u_2)^2 + (v_1 - v_2)^2]$$

$$\text{fitness} \sim e^{-S/T}$$

# Zdrojový kód

```
!integer, dimension(...,size(iorder)) :: pop,newpop
...
call createpop(pop)
do j = 1, 50000
  ! statistics, compute fitness ss(i), sss
  do k = 1, npop, 2
    ! selection
    n1 = selection(ss,sss); n2 = selection(ss,sss)
    ! crossover
    ! call the Oracle to get harvest
    if( harvest < crossprob ) then
      ...
      call crossover(pop(n1,:),pop(n2,:),c1,c2,newpop(k,:),newpop(k+1,:))
    else
      newpop(k,:) = pop(k,:); newpop(k+1,:) = pop(k+1,:)
    end if
  end do
  ! mutation
  call mutation(newpop,mutprob)
  ! new generation
  pop = newpop
  ! write output or exit
end do
write(*,*) j,minpath,iorder,s
...
```

# Srovnání s dalšími matchovacími programy

## Utility

- ▶ FOCAS (IRAF), (<http://adsabs.harvard.edu/abs/1995PASP..107.1119V>, FOCAS Automatic Catalog Matching Algorithms)
- ▶ match: (<http://spiff.rit.edu/match/>)
- ▶ SExtractor (<http://sextractor.sourceforge.net/>)
- ▶ munimatch (<http://munipack.astronomy.cz/>)

Principy: trojúhelníková podobnost, srovnávání přes cykly,  $\sim n^6$



# Matchování versus registrace<sup>1</sup>

## Výhody

- ▶ nepotřebuje obrázek
- ▶ může srovnávat katalog a polohy hvězd z obrázku
- ▶ lépe funguje na málo hvězd
- ▶ je mnohem přesnější
- ▶ lze ošetřit složitější geometrické transformace

## Nevýhody

- ▶ je třeba detekovat “hvězdy”
- ▶ nefunguje na komplexní obrázky
- ▶ může být pomalejší

---

<sup>1</sup>Registrace odvozuje geometrické transformace na základě srovnání Fourierových obrazů

# Řídí se i reálné události GA?

- ▶ vědecké články se skládají jako geny z jednotlivých myšlenek?
- ▶ vývoj společnosti (lze “vybočit z řady”)?
- ▶ řídí se vývoj technologií (aut, počítačů, . . .) evolucí?
- ▶ jak a proč jsou nastavené fitness funkce?
- ▶ jak a proč jsou nastavené hodnoty mutací a křížení?
- ▶ genetické programování

# Reference

- ▶ Marek Obitko: Introduction to Genetic Algorithms, <http://cs.felk.cvut.cz/~xobitko/ga/>
- ▶ David Coley: An Introduction to Genetic Algorithms for Scientists and Engineers, World Scientific, 1999
- ▶ Press, Flannery, Teukolsky, Vetterling: Numerical recipes (Salesman problem, Simulated annealing)
- ▶ <http://www.scienceworld.cz/sw.nsf/0/AF36833D48EA0D2FC1256E9700489AF0?OpenDocument&cast=1>
- ▶ wikipedia rulez!