

# Simulovaná evoluce I.

- princip a analogie
- základní typy simulované evoluce
- kódování instancí
- evoluční operátory
- selekce a selekční tlak
- řízení generací
- podmínky ukončení
- práce s omezujícími podmínkami

- schémata
- teorie stavebních bloků
- zavádějící (klamné) funkce
- kompetentní GA, fast messy GA
- bayesovská optimalizace
- paralelizace

# Principy

- více stavů najednou
  - snížení možnosti uváznutí v jediném minimu
  - zpravidla konstantní počet stavů po celou dobu výpočtu

- operace
  - $S \rightarrow S$  (unární operátory)
  - $S \times S \rightarrow S, S \times S \rightarrow S \times S$  (binární operátory)

- definice operací
  - nad reprezentacemi (zakódovanými stavy)
  - problémově nezávislé

# Analogie

konfigurace  $\rightarrow$  jedinec (fenotyp)  
kódování konfigurace  $\rightarrow$  genetická reprezentace jedince (genotyp, chromozóm)  
proměnná kódování  $\rightarrow$  gen  
hodnota proměnné  $\rightarrow$  alela  
aktuální množina konfigurací  $\rightarrow$  generace, populace

# Analogie

unární operátor  $\rightarrow$  mutace  
binární operátor  $\rightarrow$  křížení  
optimalizační kritérium  $\rightarrow$  zdatnost (fitness)  
rozšíření konfigurace uváznulé v lokálním minimu  $\rightarrow$  degenerace  
rozšíření kvalitní konfigurace  $\rightarrow$  konvergence

# Kostra simulované evoluce



# Základní typy simulované evoluce

## Genetické algoritmy

v klasické podobě

kódování	binární řetěz
operátory	binární: křížení <ul style="list-style-type: none"><li>• 1-bodové</li><li>• 2-bodové</li><li>• uniformní ...</li></ul> unární: <ul style="list-style-type: none"><li>• mutace</li><li>• inverze</li></ul>
řízení populace	nová generace nahradí původní

## Genetické programování

kódování	rozkladový strom výrazu
operátory	<ul style="list-style-type: none"><li>• křížení</li><li>• mutace</li><li>• definice stavebního bloku</li><li>• editace</li></ul>
řízení populace	nová populace nahradí starou

Objevování a využití regularit, symetrií, podobností, vzorců

## Evoluční programování

kódování	automat
operátory (pouze unární)	<ul style="list-style-type: none"><li>• změna výstupního symbolu</li><li>• změna přechodu</li><li>• přidání/vypuštění stavu</li><li>• změna počátečního stavu</li></ul>
řízení populace	z $\mu$ rodičů a $\lambda$ potomků se vybere $\mu$ členů nové generace

Operátory ohodnoceny mírou změny

## Evoluční strategie

kódování	vektor (reálných) čísel a jejich středních odchylek
operátory	<ul style="list-style-type: none"><li>• mutace (dominuje): přičtení Gaussova rozložení</li><li>• křížení<ul style="list-style-type: none"><li>- diskriminující</li><li>- průměrující</li></ul></li></ul>
řízení populace	z $\mu$ rodičů a $\lambda$ potomků se vybere $\mu$ členů nové generace, nebo náhrada vcelku

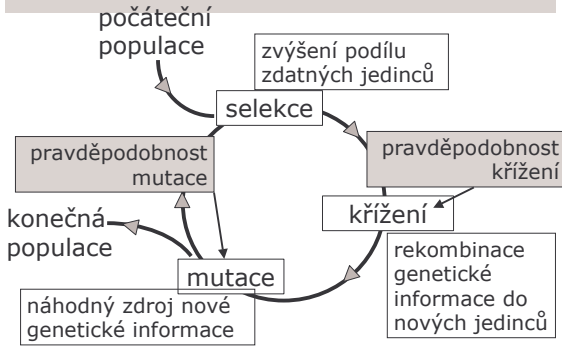
## Genetické algoritmy

- kódování (reprezentace)
- operace (genetické operátory)
- výběr pro operace (selekce)
- selekční tlak
- řízení populace

## Kódování

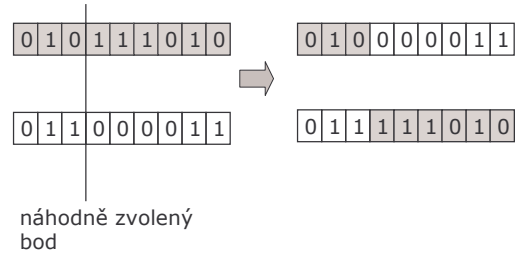
- klasická formulace: binární řetěz
  - přirozený pro řadu problémů (batoh, SAT...)
- vektor proměnných obecně různých domén
- permutace řetězce z dané abecedy
  - permutační problémy: např. TSP

# Genetické operátory



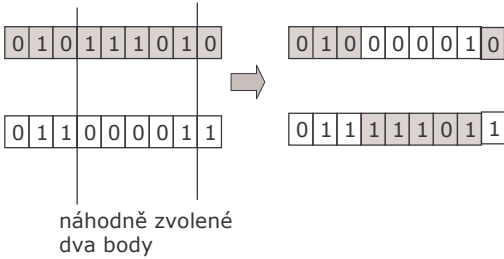
# Křížení (rekombinace)

jednobodové křížení



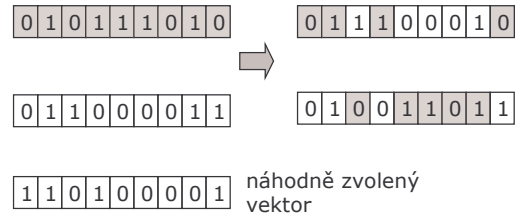
# Křížení (rekombinace)

dvoubodové křížení

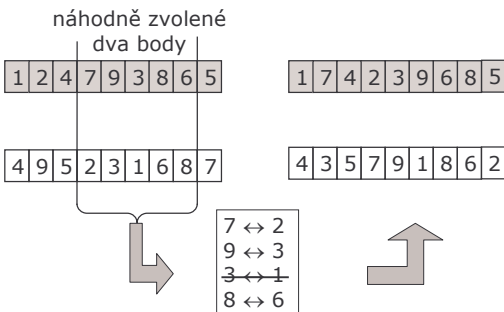


# Křížení (rekombinace)

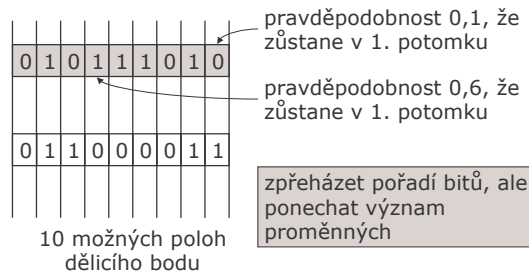
uniformní křížení



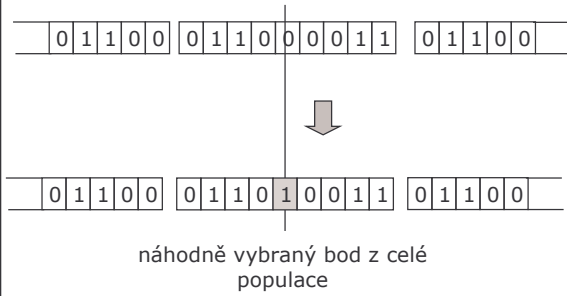
# Křížení pro permutační problémy (PMX)



# Inverze

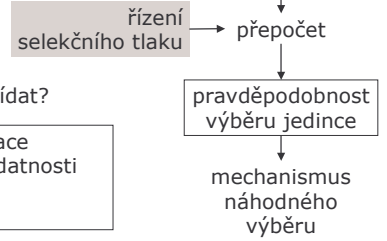


## Mutace



## Selekce

Účel: způsobit, aby početní zastoupení jedince v populaci odpovídalo jeho zdatnosti



Co je to odpovídat?

Převod informace obsažené ve zdatnosti na informaci v početnosti

## Selekce a mutace

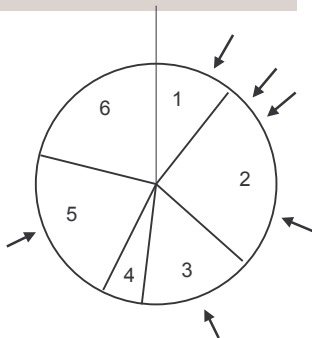


## Selekční tlak

- Pravděpodobnost výběru nejlepšího jedince
- Velký selekční tlak → nebezpečí degenerace populace
- Malý selekční tlak → pomalá konvergence
- Šum vnesený mutací může převážit nad pomalou konvergencí → divergence
- Regulace selekčního tlaku se liší podle výběrového mechanismu

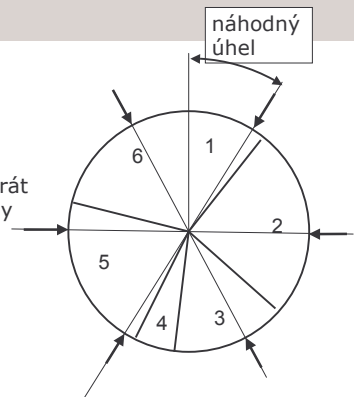
## Ruletový výběr

- Rozevření políčka rulety odpovídá žádané pravděpodobnosti výběru
- Provedeme  $m$  náhodných voleb úhlu, odtud  $m$  prvků
- Jeden prvek může být vybrán vícekrát
- 122235



## Univerzální stochastické vzorkování

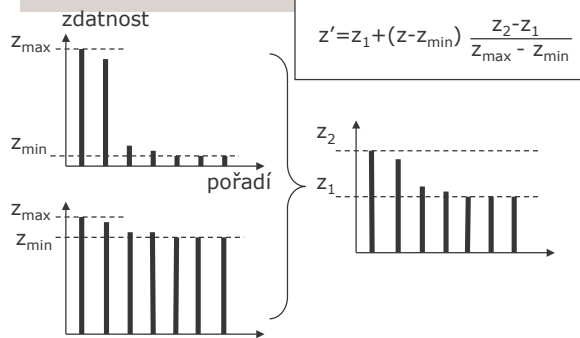
- Odměříme náhodný úhel a vybereme prvek
- Od tohoto bodu odměříme  $m-1$  krát úhel  $2\pi/m$  a vždy vybereme prvek
- 123556



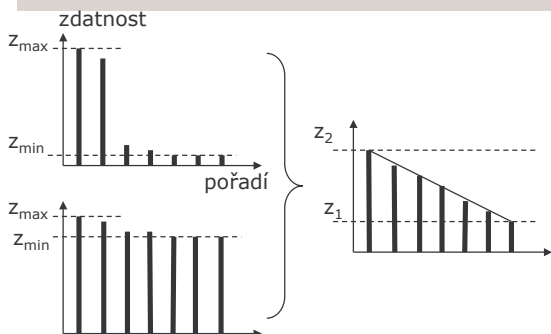
## Řízení selekčního tlaku pro ruletový výběr

- Přímá úměra mezi zdatností a poměrnou pravděpodobností výskytu často zdegeneruje populaci – je třeba nadržovat slabším
- Přepočítání zdatnosti lineární funkcí (scaling)
- Použití pořadí ve zdatnosti místo zdatnosti (ranking)

## Lineární škálování



## Ranking



## Turnajový výběr

- Náhodně vybrat  $r$  jedinců (turnaj) a z něj nejlepšího
- Opakovat až do naplnění populace
- Selekční tlak:
  - řídí se velikostí turnaje
  - „turnaj“ s jedním jedincem – žádný selekční tlak
  - turnaj přes celou populaci – jistota výběru nejlepšího jedince

## Zkrácený výběr

### Truncation Selection

- Dán práh  $0 < p < 1$
- Z populace  $M$  jedinců se vybere  $pM$  nejzdatnějších
- Každý vstupuje do rekombinace  $1/p$  krát
- Informace obsažená ve zdatnosti je redukována na jediný bit!
- Variace: několik (typicky 4) stejně velkých skupin podle zdatnosti, pro každou skupinu četnost výběru; redukce informace na typicky 2 bity

## Řízení populace

- Náhrada en bloc – nová generace vzniklá křížením nahradí starou
- Elitismus – několik málo nejlepších jedinců přejde ze staré populace do nové (pozor na degeneraci)
- Ustálená populace – po každém křížení potomek (potomci) nahradí nejslabšího (dva nejslabší) jedince
- Přechodové formy mezi en bloc a ustálenou populací

## Velikost populace

- Teorie: nutná velikost populace roste exponenciálně s velikostí problému:  $1,65 \cdot 2^{0,21 \cdot \text{délka}}$  (Goldberg) - nepraktické
- velmi malé populace ( $\sim 10$  jedinců): nebezpečí ztráty diverzity
- méně obtížné problémy: od 30 jedinců, lineární růst ( $n - 2n$ )
- obtížné problémy:  $\sim 100$  jedinců
- velké instance: sublineární růst s velikostí populace, technické důvody

## Podmínky ukončení

- Pevný počet generací
- Příznaky konvergence
  - změna průměrné zdatnosti mezi generacemi
  - rozložení zdatnosti v generaci (střední odchylka apod.)

## Omezující podmínky

Co udělat, jestliže výsledek genetického operátoru není řešením?

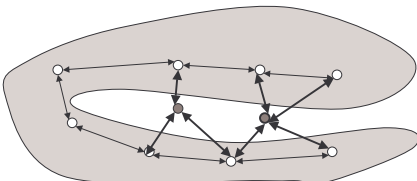
- relaxační techniky
- „trest smrti“
- speciální reprezentace a dekodéry
- oprava individuí

## Relaxace

- Princip: převést omezující podmínky na penalizaci
- Velikost penalizace: závislé na instanci
  - každé řešení lepší než ne-řešení
  - minimální penalizace, která zabrání, aby bylo ne-řešení vybráno jako optimální
- Způsob penalizace
  - vzdálenost od řešení
  - odhad ceny opravy
  - počet porušených podmínek

## „Trest smrti“

- zahodit výsledek
- zmarněná práce: výpočet optimalizačního kritéria, genetický operátor
- snížená dostupnost stavového prostoru



## Doménové reprezentace a operátory

- viz permutační operátory
- jiný příklad:
  - genotyp je vektor reálných čísel
  - nechť  $x_1, x_2$  řešení  $\Rightarrow \lambda_1 x_1 + \lambda_2 x_2$  řešení pro  $0 \leq \lambda_1, \lambda_2 \leq 1$  (konvexní prostor)
  - použijeme  $\lambda_1 x_1 + \lambda_2 x_2$  jako základ operátoru křížení s náhodným  $\lambda_1, \lambda_2$

# Dekodéry

- Princip: volit reprezentaci tak, aby každý genotyp odpovídal řešení
- Požadavky:
  - každé řešení musí být poskytnuto
  - každý výstup dekodéru je řešením
  - každé řešení reprezentováno tímž počtem genotypů (statistické vlastnosti)
  - lokality: malá změna genotypu působí malou změnu řešení
  - rychlý výpočet