

# Intelligens Rendszerek

Genetikus algoritmus

Dr. Molnár András

# Biológiai inspiráció

## **A genetikus programozás alapját Darwin evolúciós elmélete adja.**

Fennáll egy életközösség, egy populáció valahány egyeddel. Az egyedek egy része szaporodik, létrehozzák utódaikat, valahány egyed tovább él, a többi egyed pedig elpusztul. A közösségben generációk sora váltja egymást, miközben az egyedek átalakulnak, jó esetben fejlődésen mennek át.

Ha jól szabályozott a rendszer, akkor az újonnan létrejött populáció hosszú távon átlagosan jobb "minőségű" egyedeket fog tartalmazni, és létrejöhet olyan egyed, ami valamilyen szempontból ideális.

# Mikor használhatók?

**A genetikus algoritmusokat akkor célszerű alkalmazni, ha:**

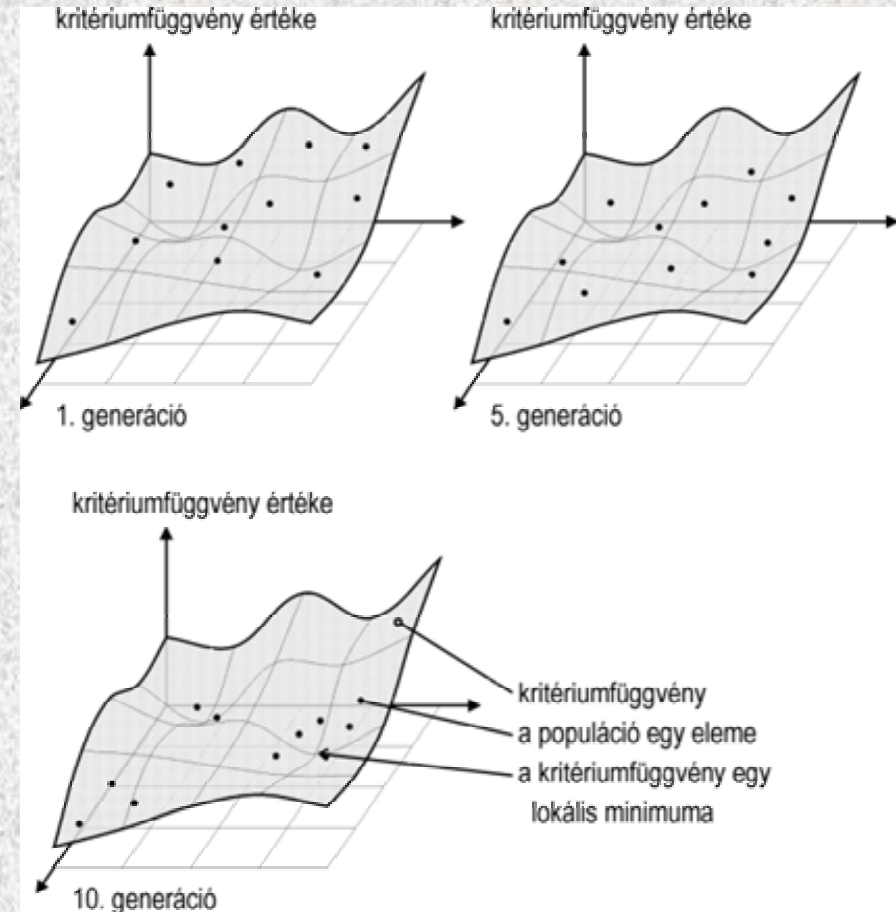
- **Ismert a probléma célértéke (kritériumfüggvény).**
- **Nem ismert a pontos célérték, de leírható (meghatározható) az egyes eredmények egymáshoz viszonyított jósága.**
- **Nem ismert az eredményhez vezető optimális út.**
- **Sok eljárás ismert, de nem ismert olyan módszer, amely optimális eredményt ad.**

- Amikor a kritériumfüggvény felületének lokális minimumai vannak, önmagában a gradiens módszerek nem alkalmazhatóak, mert „beragadnak” egy lokális minimumba!
- Zaj bevezetése lehetővé teszi a lokális minimumokból való kiszabadulást.
- *Az evolúciós algoritmusokat sztochasztikus szélsőérték-keresésnek nevezzük. Ezek biológiai analógiákra épülnek.*
- *Az evolúciós algoritmusok négy fő csoportja:*
  - *a genetikus algoritmusok,*
  - *a genetikus programozás,*
  - *az evolúciós stratégiák és*
  - *az evolúciós programozás.*

# Az algoritmus mechanizmusa

- A probléma grafikus reprezentációja:

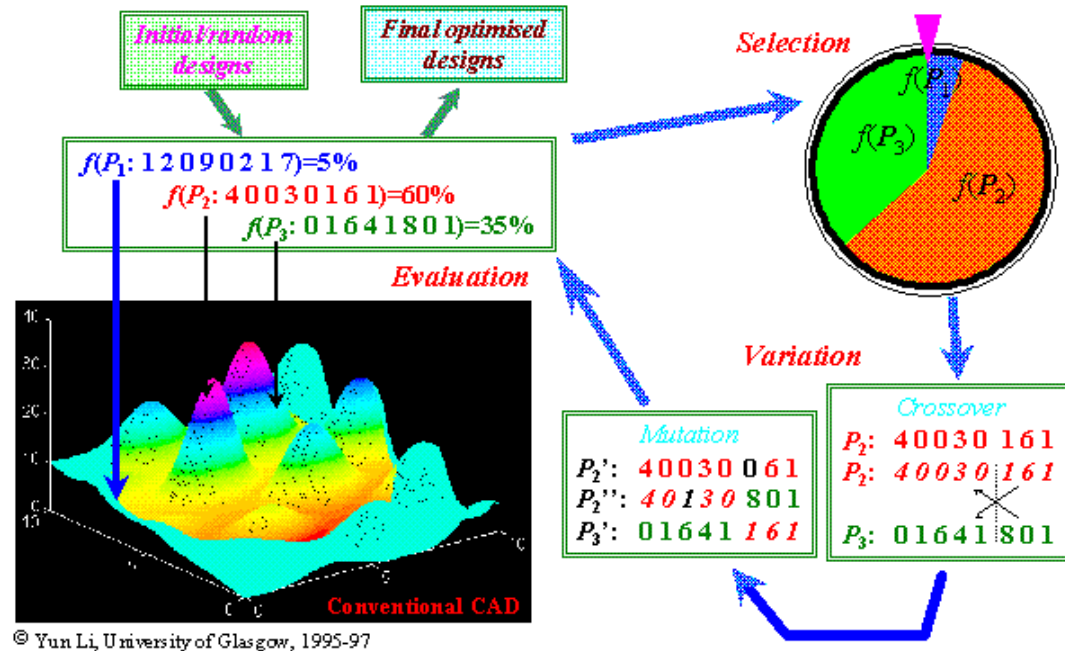
Adott génkészlettel véges sok lépésben a rendszer „leáll”, azaz nem javul tovább!



# Az algoritmus mechanizmusa

- A kezdeti populáció mértéke hatással van az eljárás sebességére:
  - Elvi megközelítéssel
    - Nagy populáció elvileg gyorsabb eredményt ad
    - Kis populáció elvileg lassabban ad eredményt
  - Számítás igény miatt
    - Nagy populáció lassú
    - Kis populáció gyors
  - Meg kell keresni az optimum közeli populációt!
  - A populáció mértéke nincs befolyással az eredmény jóságára!

## Computer-Automated Design by Artificial Evolution



- A mutáció biztosítja a rendszer folyamatos működését (jóságának növekedését). A mutáció mértéke szintén hatással van az eljárás sebességére:
  - Mutáció nélkül a rendszer gyorsan leáll.
  - Alacsony mutáció esetében lassan jutunk eredményre.
  - Magas mutáció esetében a rendszer „szétesik”, nem ad eredményt.

# Az algoritmus mechanizmusa

- Minden evolúciós algoritmus közös általános végrehajtási ciklusa:
  - stratégiai paraméterek választása (induló egyedek (individuumok) száma, szelekciós eljárás kiválasztása stb.)
  - a populáció kezdőértékeinek megadása
  - az individuumok értékelése (fitness számolás)
  - generációs ciklus:
    - » szülők választása, utódok generálása műveletekkel
    - » utódok értékelése (fitness számítás)
    - » új populáció kiválasztása (különböző stratégiákkal)
    - » megállási feltétel (általában adott számú ismétlés, azaz generáció szám után, vagy adott jóság elérése után)
  - eredmények közzlése

# Szelekciós eljárások

- **Fitness arányos szelekció rulettkerék mintavételezéssel:** Ennél az eljárásnál minden egyednek kijelölünk egy szeletet a rulettkerékből úgy, hogy a szelet mérete arányos legyen az egyed fitness értékével. A kereket megpörgetjük, és az az egyed lesz kiválasztva, amelyik felett megáll. A gond ezzel az eljárással, hogy ha az egyedek fitness értékei között nincs nagy különbség a populáción belül, akkor minden egyed közel azonos eséllyel lehet szülőnek választva.
- **Rang szerinti szelekció:** A sorrend szerinti szelekció, éppúgy, mint a fitness arányos szelekció, a rulettkerék alapján választja ki az egyedeket. Itt a rulettkerék szelete egy egyednek a sorba rendezett populációban betöltött helye szerint határozódik meg. Így a leggyengébb egyed kap egy egységnyit, míg a legjobb  $n$  egységnyit, ahol  $n$  a populáció nagysága.
- **Verseny szelekció:** A verseny szelekció hasonló a sorrend szerinti szelekcióhoz a szelekciós nyomást tekintve, de sokkal hatékonyabban számítható. Két egyedet kiválasztunk véletlenszerűen a populációból. Egy véletlen  $r$  számot generálunk 0 és 1 között. Ha  $r < k$  (ahol  $k$  egy paraméter, például 0,75), akkor a jobb egyedet választjuk. Egyébként a gyengébb egyed kerül kiválasztásra.
- **Legjobb szelekció:** Ennél az eljárásnál egyszerűen a populáció legjobb egyedét választjuk.
- **Véletlen szelekció:** Ennél az eljárásnál véletlenszerűen választunk ki egy egyedet.
- **Interaktív szelekció:** Ezt az eljárást az interaktív evolúció érdekében vezették be. Interaktív evolúcióval például grafikai elemeket lehet kifejleszteni, ahol az alkalmas fitness függvényt nem tudjuk megalkotni. Ebben az esetben a program felhasználója tud választani az egyedek közül. Egyéb területek mellett az interaktív szelekciót gyakran alkalmazzák tervezéshez és alakfelismeréshez.

# Szelekciós eljárások

- **Szigma scaling:** A szigma scaling (mint a fitness csökkentés) egy másik eljárás a fitness arányos szelekció problémájának megoldására. Itt a rulettkerék szelete az egyed fitness értékének, a populáció átlagának és szórásának függvényében rendelődik az egyedekhez. A módosított fitness értékeket az alábbiak szerint számítjuk ki:

$$f^*(i) = \begin{cases} 1 + \frac{f(i) - F}{2s} & \text{ha } s \neq 0 \\ 1 & \text{ha } s = 0 \end{cases}$$

Ahol  $f^*(i)$  az  $i$  egyed módosított fitness értéke,  $f(i)$  az  $i$  egyedhez tartozó eredeti fitness érték,  $F$  a populáció átlag fitness értéke és  $s$  a populáció fitness értékeinek szórása.

- **Boltzmann szelekció:** A szigma scaling a "szelekciós nyomást" a folyamat alatt egy konstans érték körül tartja. Néha szükségünk lehet rá, hogy a szelekció jobban kihangsúlyozza a magasabb fitness értékű egyedeket. Ezen a felfogáson alapszik a Boltzmann szelekció. Ebben az eljárásban a rulettkerék szelete a következő formulával rendelődik egy  $i$  egyedhez:

$$f^*(i) = \frac{\exp(f(i)/T)}{\langle \exp(f(i)/T) \rangle}$$

Ahol  $f^*(i)$  az  $i$  egyed módosított fitness értéke,  $f(i)$  az  $i$  egyedhez tartozó eredeti fitness érték, a  $\langle \dots \rangle$  az aktuális populáció középértékét jelöli és  $T$  egy hőmérséklet, mely egyenletesen csökken a folyamat során. Alacsonyabb  $T$  értékek mellett nő a különbség az  $f^*(i)$ -ben az alacsony és magas fitness értékek között.