

Evoluční výpočty

(úvod)

Jan Žižka

Institut Biostatistiky a Analýz (IBA)
Masarykova universita
Kamenice 126/3
62500 Brno

`zizka@iba.muni.cz`

Obsah

- Vyhledávání v mnoharozměrném prostoru
- Optimalizace, evoluce
- Evoluční výpočty, Darwin, simulace darwinovského vývoje
- Genetické algoritmy (GA), základní principy a parametry
- Výběr členů populace pro reprodukci (přizpůsobenost)
- Reprodukce a základní metody (křížení, mutace)
- Diversita populace a její využití
- Genetické programování
- Ukázka aplikace GA na TSP
- Shrnutí
- Základní literatura

Vyhledávání v mnoharozměrném prostoru

Existuje mnoho metod umožňujících *hledat optimum*. Patří k nim *metody analytické* (derivační a numerické výpočty), a pokud je nelze použít, pak *metody využívající „hrubé síly“* jako je systematické prohledávání sekvenční (nebo náhodné s vysokým počtem pokusů).

Na hledání optima se používají také různé *metody aplikované v umělé inteligenci* (hledání do šířky, do hloubky, mini-max, α - β , A^* a řada dalších).

Metody umělé inteligence se snaží *redukovat vysokou výpočetní náročnost* systematického prohledávání a zároveň vysokou *nejistotu náhodného hledání*.

Optimalizace

Jedním z ideálních cílů indukčního strojového učení je stanovení *optimálních parametrů algoritmů*, které slouží např. jako klasifikátory (rozhodovací stromy, naivní Bayes, aj.) nebo aproximátory neznámých mnoharozměrných funkcí (umělé neuronové sítě, nelineární regrese, aj.). Často však nelze optima dosáhnout—např. nulové klasifikační chyby na datech přicházejících v budoucnu, po ukončení tréninku.

Není-li možné z libovolných důvodů stanovit potřebné parametry *přímým výpočtem* (např. kvůli neznalosti skutečného rozložení dat—podmínkou často bývá normální rozložení, nebo kvůli neznámé nelineárnosti či kombinatorické a výpočetní složitosti, apod.), lze aplikovat alternativní metody, které se pokusí hledané parametry indukovat z trénovacích příkladů.

Optimalizace (pokračování)

Jinou alternativou strojového učení je metoda založená na představě, že z *omezeného množství příkladů* možných řešení se vybere to nejlepší, které je k dispozici a zároveň poskytuje přijatelné výsledky.

Otázka je, jak a kolik příkladů vytvořit, aby z nich bylo možné vybrat kvalitní řešení se spolehlivým odhadem maximální chyby na budoucích datech, neznámých v době tréninku. Jednou z možností je *metoda náhodných pokusů a omylů*, která ale pro reálné komplexní problémy je z časových a paměťových důvodů prakticky nepoužitelná: důvodem je aplikace pouhé náhodnosti ovlivněné velmi malou pravděpodobností dosažení optima.

EVOLUTIONARY COMPUTATION

Optimalizace (pokračování)

Inspirací slibného, i když časově náročného řešení je teorie *evolučního vývoje*, která vysvětluje vznik biologických druhů v určitém prostředí za předpokladu, že je k dispozici dostatek času na postupné přizpůsobování se existujícím podmínkám.

Pokusy o simulaci vývoje a mutace se začaly používat v praktických inženýrských odvětvích k řešení vysoce složitých problémů v oblastech statistického řízení procesů, strojového učení a optimalizace funkcí. Příkladem může být návrh optimálního tvaru vnitřku stříkací pistole pro dosažení co nejlepšího promíchání plynu a tekutiny; matematicko-fyzikální popis turbulence k dokonalému návrhu nepostačoval. Jinou úlohou byl návrh složitého elektronického filtru—jaké součástky, v jakém propojení, a jak minimalizovat cenu?

EVOLUTIONARY COMPUTATION

Optimalizace (pokračování)

Abstraktní představa chápe každý parametr řešení jako jednu z dimenzí a možná řešení jako body v mnoharozměrném prostoru, který může trpět mnoha funkčními potížemi známými z matematiky: vysoce zvlněná mnoharozměrná krajina s řadou lokálních minim a maxim, sedlovými body, nespojitostmi, body bez možnosti derivace, apod. Najít globální extrém je analyticky nemožné.

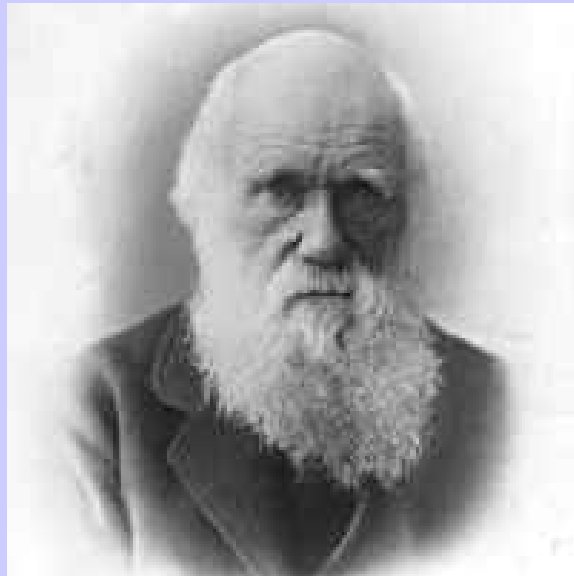
První pokusy již v r. 1959 navrhovaly hledání (sub)optima metodou „šplhání do kopce“ (*hill-climbing*) z různých míst prostoru s nadějí, že odněkud bude nalezen přijatelný extrém (nemusí být nutně globální, protože nemusíme vůbec vědět, jak by měl globální extrém vypadat—akceptujeme pak třeba řešení poskytující přijatelně malou chybu a např. řešení s nulovou chybou nemusí nikde existovat).

Optimalizace a evoluce

Experimenty s dynamickým evolučním vývojem umělého života (např. hierarchické ekosystémy) založené na vývoji populace přinesly z praktického (ale i z teoretického) hlediska řadu slibných výsledků pro oblasti, které neumíme dostatečně dobře popsat vhodným modelem, ale v kterých přesto zcela zjevně lze dosáhnout vyhovujících řešení—na Zemi přežívá mnoho biologických druhů vlivem vývoje; je to ovšem důkaz „pouze“ empirický.

Rozvoj teoretických podkladů spolu s nárůstem výpočetní výkonnosti strojů (včetně možností využívat paralelismus) dnes umožňuje aplikovat evoluční hledání co nejlepšího výsledku celkem běžně v nejrůznějších disciplínách.

Darwinovská evoluce



Charles Robert Darwin (1809–1882) je autorem evoluční teorie, která vznikla jeho pozorováním různých druhů v přírodě. Svá pozorování prováděl jako neplacený výzkumník na pětileté vědecké expedici, která začala 27. prosince 1831 vyplutím plachetnice *HMS Beagle*.



Darwin mohl zkoumat živočichy i fosilie na různých kontinentech (převážně v Jižní Americe, ale i jinde) a svá pozorování navzájem porovnávat.

Darwinovská evoluce (pokračování)

Darwina vedly jeho výzkumy k založení teorie, popsané v knize *O původu druhů prostřednictvím přirozeného výběru* (1859).

Jeho klasická teorie vývoje v kombinaci s Weismannovým selekcionismem a Mendelovou genetikou je akceptována jako základ současných neodarwinistických přístupů ke zkoumání vzniku a vývoje živočišných druhů.

Darwinova teorie měla a má řadu zastánců i odpůrců, ale v den svého skonu 19. dubna 1882 patřil bezesporu k již obecně vysoce uznávaným vědcům. Pohřben je na hřbitově v Westminster Abbey.

Darwinovská evoluce a její simulace

Darwinova evoluce je v principu jednoduchá, vychází z existence *různorodé populace*, v níž přežívají jedinci s nejlepšími vlastnostmi pro dané prostředí.

Tento pohled na Darwinovu teorii vedl k myšlence modelovat výpočetně složité úlohy tak, že se vytvoří počáteční rozsáhlá populace, jejímiž členy jsou různě kvalitní počáteční řešení.

Cílem je dostatečně pokrýt prostor možných řešení, přičemž *není nutno generovat všechna*, a *vývojem* tvořit nová lepší řešení z přibližně (či zcela náhodně) navržených počátečních.

Darwinovská evoluce ... (pokračování)

Mnoharozměrný prostor (dimenze jsou dány parametry řešené úlohy) obecně obsahuje mnoho lokálních extrémů a cílem je najít extrém globální. Proto je zejména na počátku nutná co největší různorodost členů populace, protože hodnoty jejich vlastností jsou souřadnice jednotlivých řešení v prostoru.

Čím hustěji a rovnoměrněji je prostor obydlen, tím větší je šance, že jeden z obyvatel je přinejmenším někde poblíž hledaného globálního extrému, tj. optimálního řešení. Na druhé straně, rozsáhlá populace zvyšuje výpočetní složitost, takže je zapotřebí najít vhodný kompromis.

Darwinovská evoluce a GA

Simulovaná evoluce vychází z předpokladu, že v dané populaci existují jedinci kvalitnější a méně kvalitní. Kvalitnější jedinci jsou ti, kteří jsou blíže hledanému globálnímu extrému.

Na určení kvality jedince se používá funkce ohodnocující jeho *přizpůsobenost prostředí* (*fitness function*). Tato funkce je stanovena konkrétně vzhledem k hledanému cíli úlohy. Sleduje se, zda kvalita řešení úlohy se zvyšuje, zda klesá chyba, apod.

Lepší jedinci mají větší pravděpodobnost přežití, protože vývoj probíhá tak, že se vybírají kvalitnější členové pro vytvoření následující generace, a méně kvalitní vyhynou (nemají potomky v další generaci).

Darwinovská evoluce a GA (pokračování)

Kvalita jedince je dána kombinací hodnot jeho parametrů, které jako souřadnice udávají jeho blízkost optimu (polohu *optima* však neznáme).

Každý parametr se považuje za *gen*, tj. za jednu z vlastností.

Soubor genů jedince pak tvoří *chromosom*, což je popis jedince z hlediska *relevantních* parametrů řešené úlohy (je důležité definovat správně relevantní parametry).

Noví jedinci, *potomci*, vznikají kombinací chromosomů vybraných rodičů s možností zlepšit (i zhoršit) své vlastnosti.

Genetické algoritmy

Princip vývoje zlepšeného řešení úlohy je založen na naději, že kombinací vlastností kvalitních rodičů vzniknou ještě kvalitnější potomci, kteří od každého z rodičů zdědí souřadnice, které je posunou blíže k optimu. Protože se neví, kde se optimum nachází, ale ví se, jak měřit funkcí přizpůsobenosti přibližování (nebo vzdalování) se jedince k optimu, lze úspěšnost vývoje sledovat pomocí růstu kvality nejlepšího jedince.

Nelze jednoznačně stanovit, kdy se má evoluce zastavit. Je možné např. ukončit vývoj tehdy, kdy již přestalo zlepšování, nebo kdy byl vyčerpán předem stanovený maximální počet generací. Žádná metoda nezaručuje konec vývoje dosažením optima, protože v reálných úlohách běžně hrozí uváznutí v lokálním extrému a obecně nelze určit, že dosažený extrém je nebo není globální.

Genetické algoritmy (pokračování)

Nalezené řešení, představované nejkvalitnějším jedincem v populaci, může nebo nemusí být přijatelné, pokud se zlepšování populace zastavilo. To je zcela na interpretaci uživatelů genetických algoritmů.

Nalezené řešení, představované nejkvalitnějším jedincem v populaci, může nebo nemusí být přijatelné, pokud se zlepšování populace zastavilo. To je zcela na interpretaci uživatelů genetických algoritmů.

Genetické algoritmy (pokračování)

Genetické algoritmy (GA) mají některé **základní parametry** a řadu dalších parametrů, kterými lze podle aplikační potřeby metodu modifikovat. K základním parametrům patří:

- definice jednotlivých vlastností jedince (geny)
- stanovení metody kódování vlastností (genů) do/z chromosomu
- počet členů populace (velikost populace): čím větší, tím lepší
- funkce měření přizpůsobenosti
- způsob seřazování jedinců dle kvality
- metoda vytvoření počáteční „nulté“ generace
- metoda generování potomků (tvorba nové generace z předchozí)
- kritérium ukončení vývoje

Genetické algoritmy (pokračování)

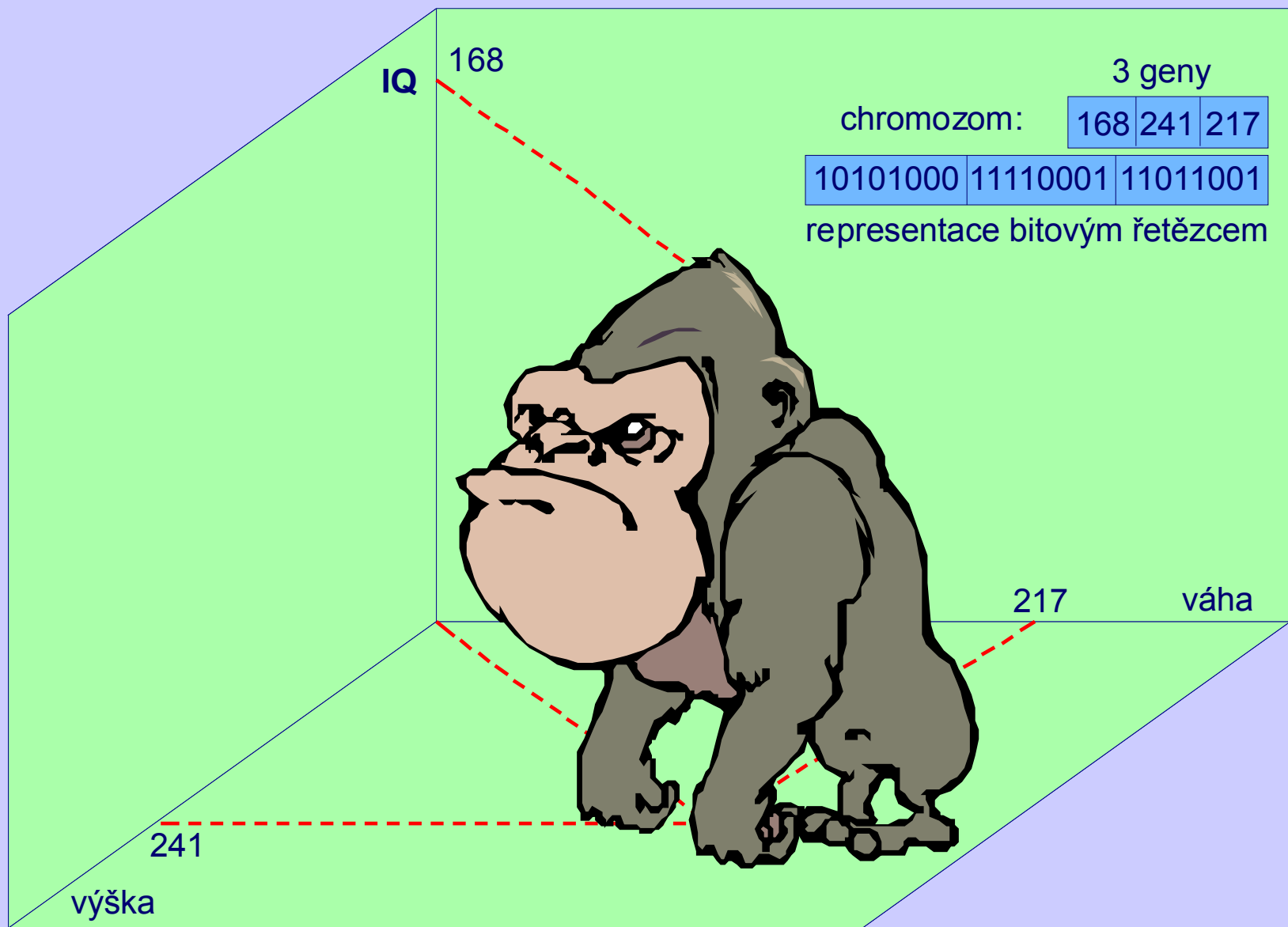
Evoluce probíhá cyklicky, vhodné je sledovat kvalitu nejlepšího jedince v každé generaci a průměrnou kvalitu populace.

Pokud se nejlepší jedinec již málo liší od ostatních, došlo k usazení se populace v nějakém místě, které může být optimem (či přijatelným suboptimem) nebo populace zdegenerovala a skončila v lokálním extrému, z něhož již nemůže uniknout.

Příklad popisu jedince pomocí tří vlastností:

IQ, výška, váha.

EVOLUTIONARY COMPUTATION

Genetické algoritmy (pokračování)

Obecný genetický algoritmus

1. **Definice** problému a funkce přizpůsobenosti každého řešení, stanovení genetických operátorů pro reprodukci.
2. **Inicializace** počáteční generace vzhledem k omezením (stanovení velikosti populace, zakódování náhodně vygenerovaných hodnot **genů** do vektoru – **chromosomu**). Nultá generace je vytvořena z mnoha co nejrozličnějších jedinců (zabydlení prostoru).
3. **Dekódování** všech chromosomů zpět do původního prostoru (může představovat značnou režii), stanovení jejich individuálních kvalit **funkcí přizpůsobenosti** a přiřazení příslušných skóre vzhledem k cíli řešení.
4. Přiřazení **pravděpodobnosti reprodukce** pro každý chromosom úměrně jeho kvalitě vzhledem k ostatním chromosomům v populaci.
5. **Pravděpodobnostním výběrem** jsou zvoleny chromosomy, na něž se použijí specifické genetické operátory (např. **křížení a mutace**) pro vytvoření potomků další generace. Aplikací **elitismu** se do další generace mohou dostat přímo i někteří nejlepší jedinci beze změny.
6. **Test**, zda bylo dosaženo kritérium **ukončení vývoje** nebo zda byl vyčerpán disponibilní čas pro vývoj. Pokud **ne**, návrat do bodu 3. Pokud **ano**, funkce přizpůsobení předá nejlepšího dekódovaného jedince



EVOLUTIONARY COMPUTATION

Standardní metoda výběru

Funkce přizpůsobení přiřadí každému chromosomu tzv. *skóre*. Nejvyšší skóre znamená nejvyšší *pravděpodobnost výběru* pro vytvoření nové generace. *Standardní metoda* počítá *přizpůsobenost* f_i i -tého chromosomu ($0.0 \leq f_i \leq 1.0$):

$$f_i = \frac{q_i}{\sum_{j=1}^n q_j}$$

Standardní metoda neumožňuje ovlivňovat výběr; jedinec s nulovou kvalitou má nulovou pravděpodobnost se zúčastnit reprodukce. Vyřazování jedinců vede k rychlejší *degeneraci* populace snižováním různorodosti (je dobré být i *odlišný*, tj. nejen *přizpůsobený*). I *nekvalitní jedinec* může mít nějaké geny, které by mohly v kombinaci s geny jiného chromosomu vytvořit potomka kvalitnějšího než jsou rodiče.

Seřazovací metoda výběru


Kvalita se použije pouze pro *seřazení chromosomů*. Nejlepší kandidát dostane pravděpodobnost $0.5 < p < 1.0$, tj. vyšší než náhodnou a nižší než jistotu. Pro p je možno zvolit např. hodnotu $2/3 \doteq 0.667$, a hodnoty zbývající do 1.0 jsou rozděleny mezi ostatní jedince podle vztahu:

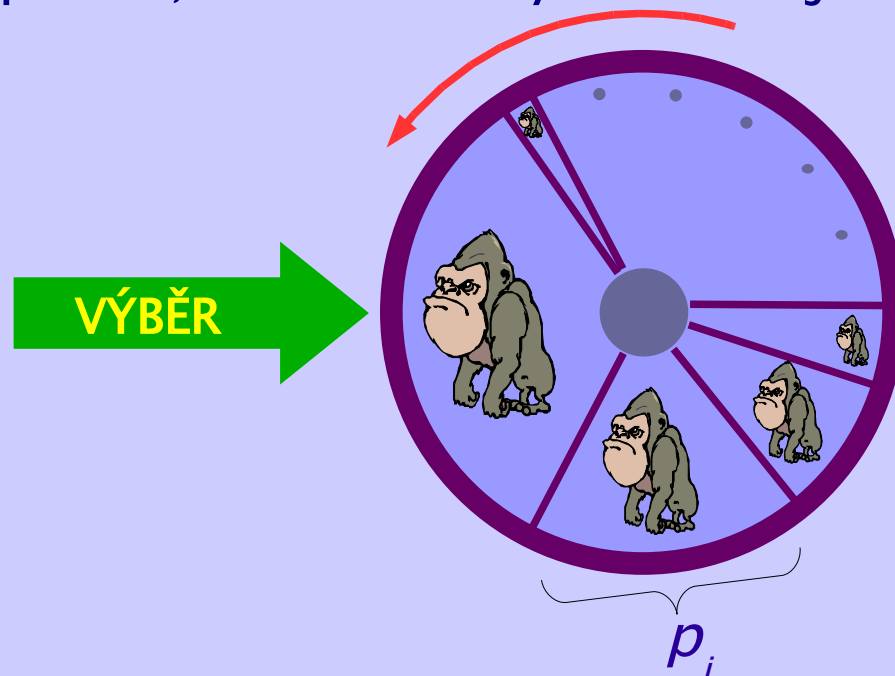
$$p_i = 1 - p \left(\sum_{j=1}^{i-1} p_j \right), \quad i=1, \dots, n$$

Seřazovací metoda přidělí pravděpodobnost nenulovou i jedincům s nulovou kvalitou. Křížení umožňuje efektivní prohledávání prostorů s množstvím lokálních extrémů. Popsané křížení je „jednopohlavní“; existují i modifikace na „dvoj- a více-pohlavní“, kdy existují specifické geny (aplikačně závislé).

EVOLUTIONARY COMPUTATION

Ruletové kolo pro výběr

Často používanou metodou výběru jedince pro reprodukci na základě *pravděpodobnosti* vyplývající z přizpůsobenosti je tzv. *ruletové kolo*. Lze si představit roztočení kola, jehož obvod má délku $= p = 1$ a je rozdělen oblouky, jejichž počet odpovídá počtu členů populace a jejichž délka je dána pravděpodobností výběru příslušných jedinců. Kolo se náhodně zastaví v nějaké pozici, která určí vybraného jedince .



EVOLUTIONARY COMPUTATION

Mutace

Mutace je napodobenina *náhodných změn*. Míra mutace odpovídá míře náhodného prohledávání prostoru; u GA se používá střídavě. Lze ovšem hledat řešení i pouze mutací, pokud prostor nemá lokální extrémy. Výhoda použití mutace je v tom, že umožňuje vnášet zcela *nové hodnoty* genů, jinak by hodnoty mohly být omezeny jen na inicializační z nulté generace. Mutace *snižuje degeneraci zvyšováním různorodosti* a zabraňuje potenciální ztrátě pohybu podél některých dimenzí prostoru (všechny parametry úlohy jsou relevantní).

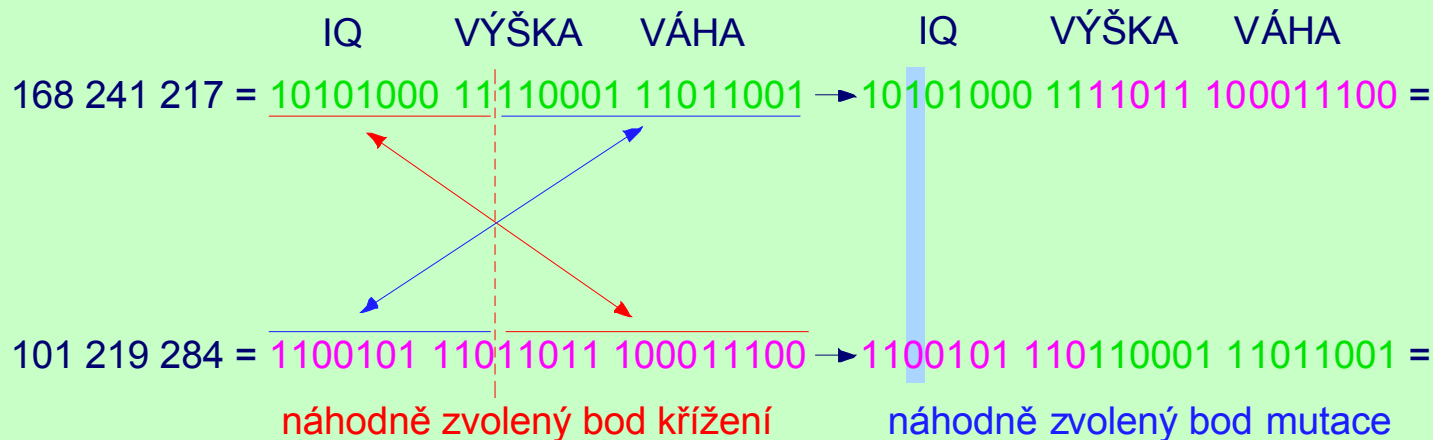
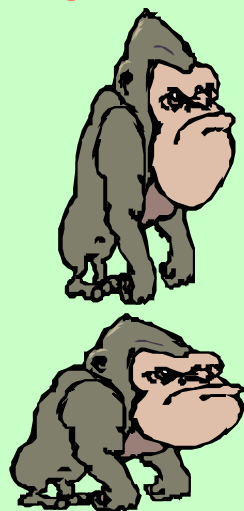
Mutace znamená náhodný výběr určité pozice (hodnoty) v chromosomu, např. jednoho bitu a změnu hodnoty z ,0‘ na ,1‘ (nebo naopak):

```
...0100000000000000100011010...  
...0100000001000000100011010...
```

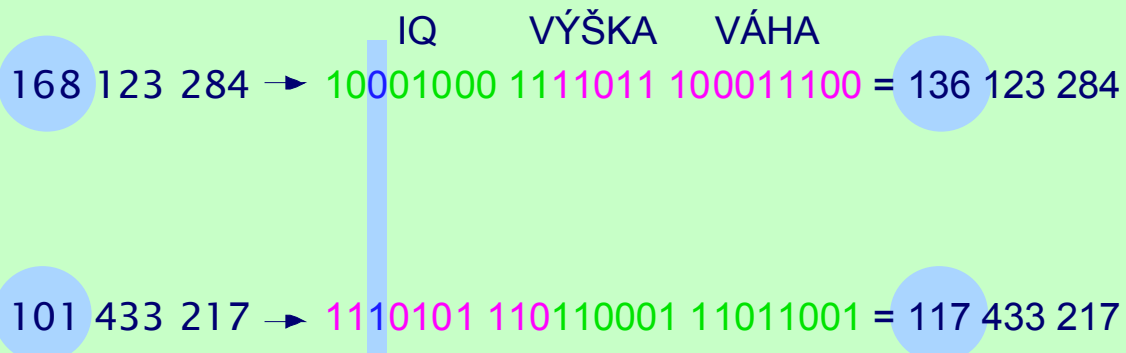

EVOLUTIONARY COMPUTATION

Reprodukce

generace k



generace $k+1$



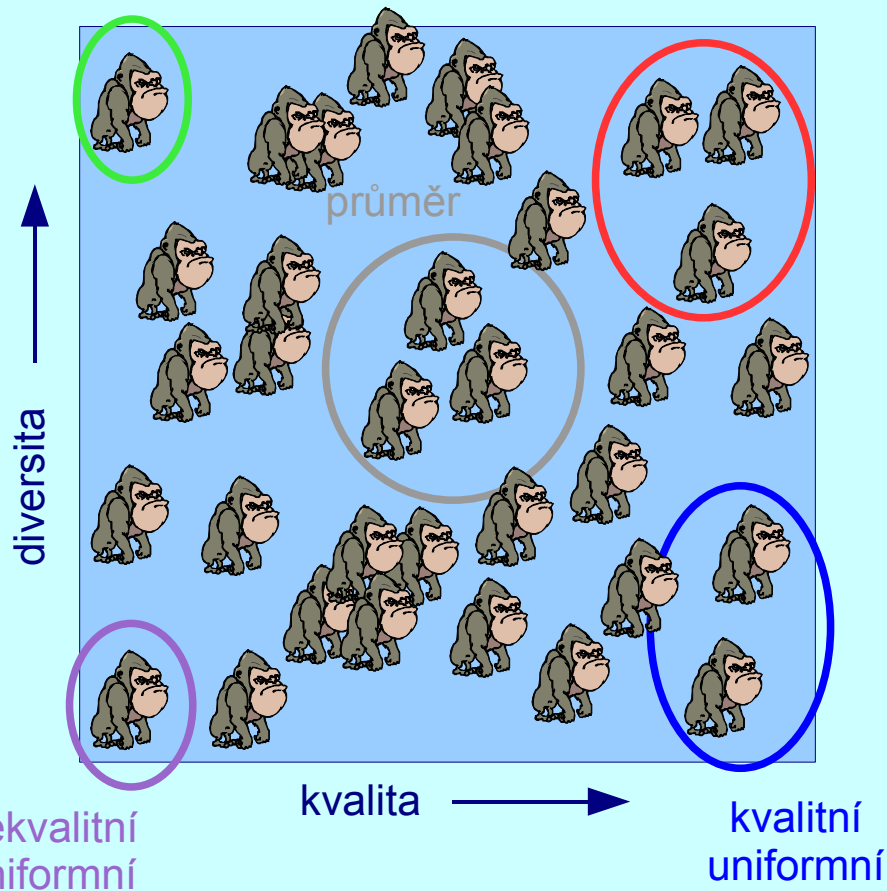
Diversita a přežití nejrůznorodějších

Chromosomy mají tendenci vymizet i tehdy, pokud je jejich skóre jen o málo nižší než u chromosomů blízkých nejlepšímu. Tato *uniformita* (či již zmíněná *degenerace*) nutí pak populaci se vyvíjet jenom určitým směrem, který nemusí vést ke kýženému globálnímu extrému. V přírodě však bylo ukázáno, že i *nepřizpůsobeně* vypadající (odlišní) jedinci a druhy přežívají docela dobře v ekologických prostředích, která leží mimo ostatní (přizpůsobeně vyhlížející) členy populace.

Prostorově-seřazovací metoda využívá *princip diversity* integrací kvality a diversity do přizpůsobivosti tak, že jedince ohodnocuje pro výběr k reprodukci i z hlediska jeho přínosu k různorodosti populace. *Míra odlišnosti* se spočítá jednoduše:

$$\text{diversita} = \sum_{i=1}^n \frac{1}{d_i^2} \quad (d \text{ je odlišnost, „vzdálenost“})$$

EVOLUTIONARY COMPUTATION

Diversita a výběr jedincůnekvalitní
různorodíkvalitní
různorodí

Jedince v populaci lze na jedné ose seřadit podle kvality a na druhé ose podle diversity, což umožní vytvořit výsledné *dvourozměrné pořadí*.

Na takto seřazenou populaci lze nakonec aplikovat výše zmíněnou seřazovací metodu (nejlepší kandidát na reprodukci dostane přiděleno $0.5 < p < 1.0$, ostatní jedinci si dle svého pořadí rozdělí zbytek pravděpodobností pro účast na reprodukci).

EVOLUTIONARY COMPUTATION

Diversita a překonávání lokálních extrémů

Pro mnoho algoritmů (např. umělé neuronové sítě) jsou lokální extrémů „pastí“, a aby v nich trvale neuvázly, vyžadují zabudování mechanismů, které umožní se z pastí dostat (např. *backtracking*, návrat po vlastní stopě). Další možnost je paralelní hledání (velký počet různých počátečních startovních bodů), kde je naděje, že jedna z větví bude chycena do globálního extrému.

U genetických algoritmů umožňuje *diversita* jako složka přizpůsobivosti vyhnout se již ***obsazenému*** lokálnímu extrému a „strhnout“ za sebou další putující jedince. Tím se lze vyhnout tomu, aby ***celá*** populace uvízla v jednom extrému a nebyla pak schopna najít i další, možná ještě lepší extrémů.

Diversita a překonávání... (pokračování)

Je-li *dostatečný počet jedinců* na obsazení lokálních extrémů, pak je dobrá šance, že někdo doputuje až do optima.

Genetické algoritmy *nemají* jako cíl se *vyhýbat lokálním extrémům*. Naopak: obsadí je a zbytek populace, odlišný od již „zabydlených“ členů, putuje dál a může (i nemusí) najít ještě lepší řešení.

Tím dochází ke specializaci („degeneraci“) populace v pozitivním smyslu. Po obsazení extrému se zkoumá, zda je v populaci někdo, kdo může „ještě výš“, a pokud ano, lze ho jako elitu použít pro další generaci přímo nebo i prostřednictvím křížení, protože má větší pravděpodobnost být vybrán. Nová generace pak směřuje od obsazeného lokálního extrému.

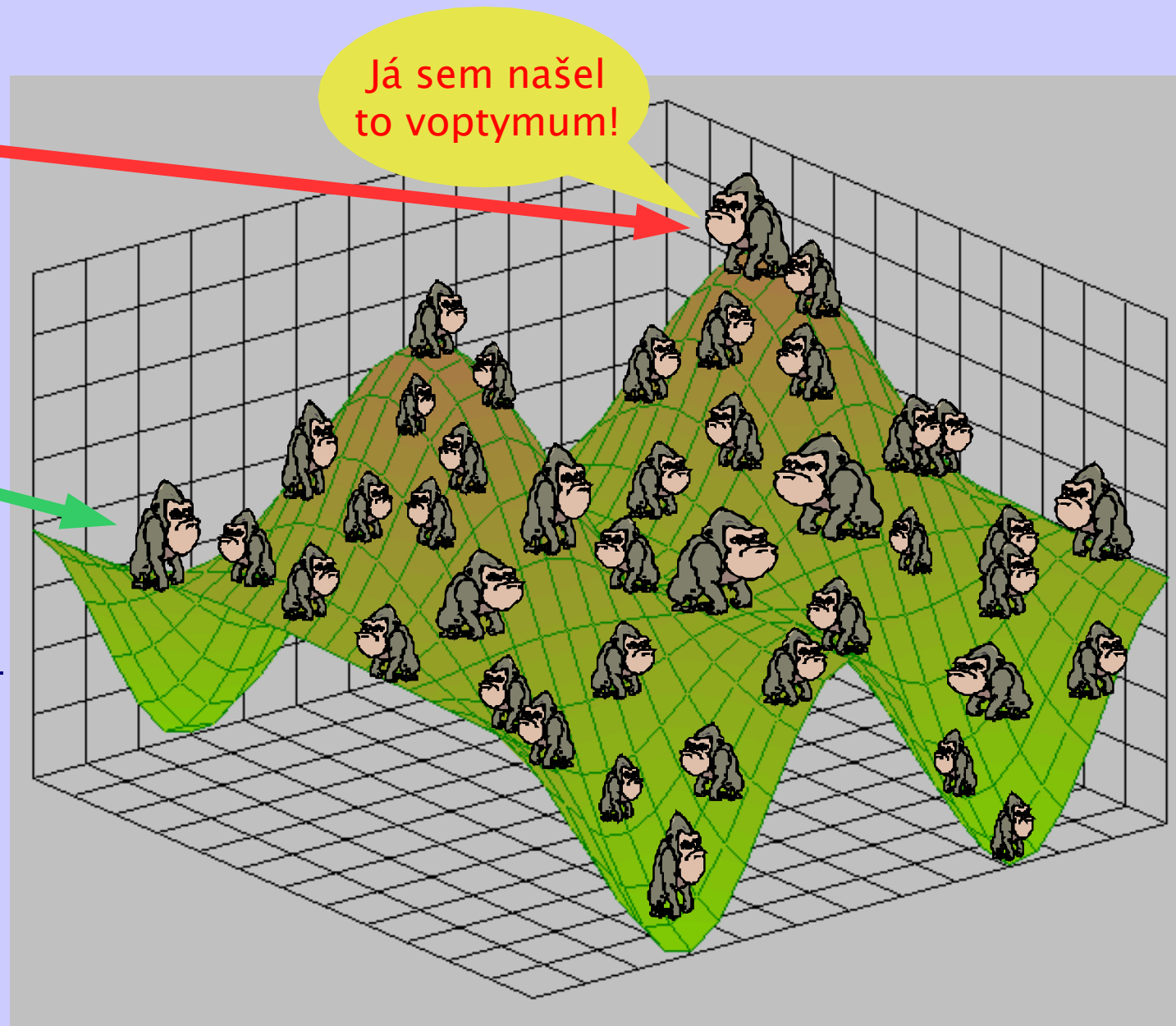
EVOLUTIONARY COMPUTATION

V m -té generaci se vyvinul jedinec, který se vyšplhal až na globální extrém.

Ostatní jedinci obsadili své niky.

Zde jsou jedinci, jejichž předkové přežili vlivem evoluce. Nepřizpůsobiví jedinci „vyhynuli“.

Jedinci v okolí optima již mají velmi podobné vlastnosti.



Genetické programování

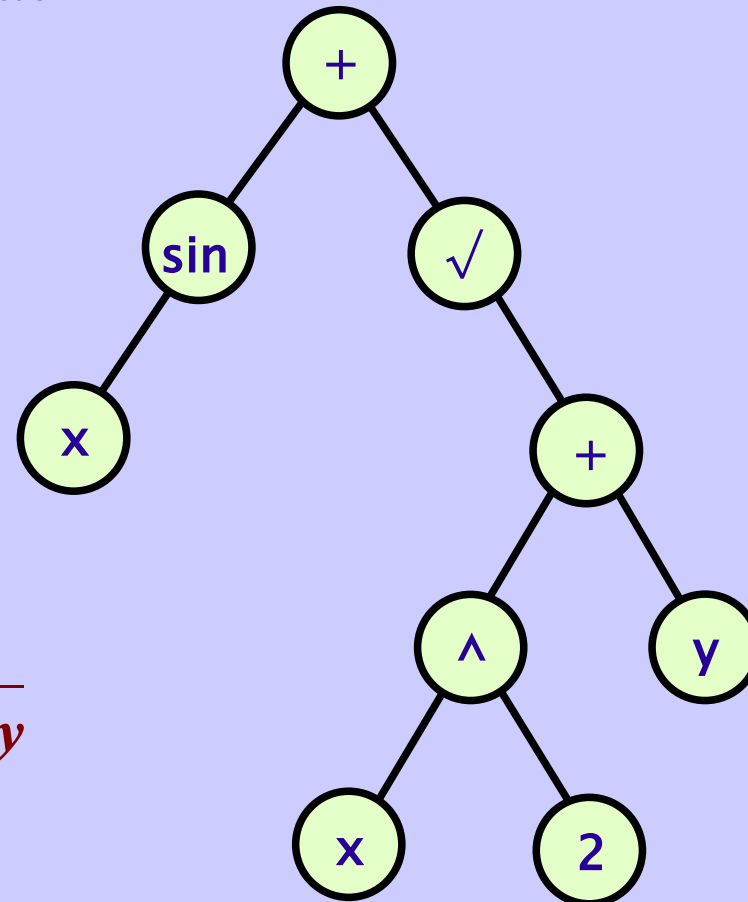
Zajímavá je idea aplikace principů evolučních výpočtů a GA na vývoj (částí) počítačových programů. Princip spočívá v *hledání neznámé funkce*, která má co nejlépe vyhovovat zadaným kritériím. Jedná se o určitou specializaci GA, která přinesla řadu úspěchů v praktické oblasti, např. řízení procesů.

Pro hledání neznámé funkce musí uživatel definovat *primitivní funkce* (např. \sin , \cos , \tan , \cap , \cup , $+$, $-$, Π , Σ , $\sqrt{\quad}$, $\sqrt[3]{\quad}$, $\sqrt[4]{\quad}$, *apod.*), z nichž chce sestavit výslednou hledanou složitější funkci. K tomu definuje *terminály* (např. proměnné x , y , z , i , j , n , m , konstanty 1 , 2 , 3 , π , *apod.*). Algoritmus GA pak prohledává velmi rozsáhlý prostor možných programů, které mohou být popsány primitivami. Zpočátku samozřejmě existují i zcela nesmyslné, náhodně sestavené funkce, které však brzy vymizí.

EVOLUTIONARY COMPUTATION

Genetické programování (pokračování)

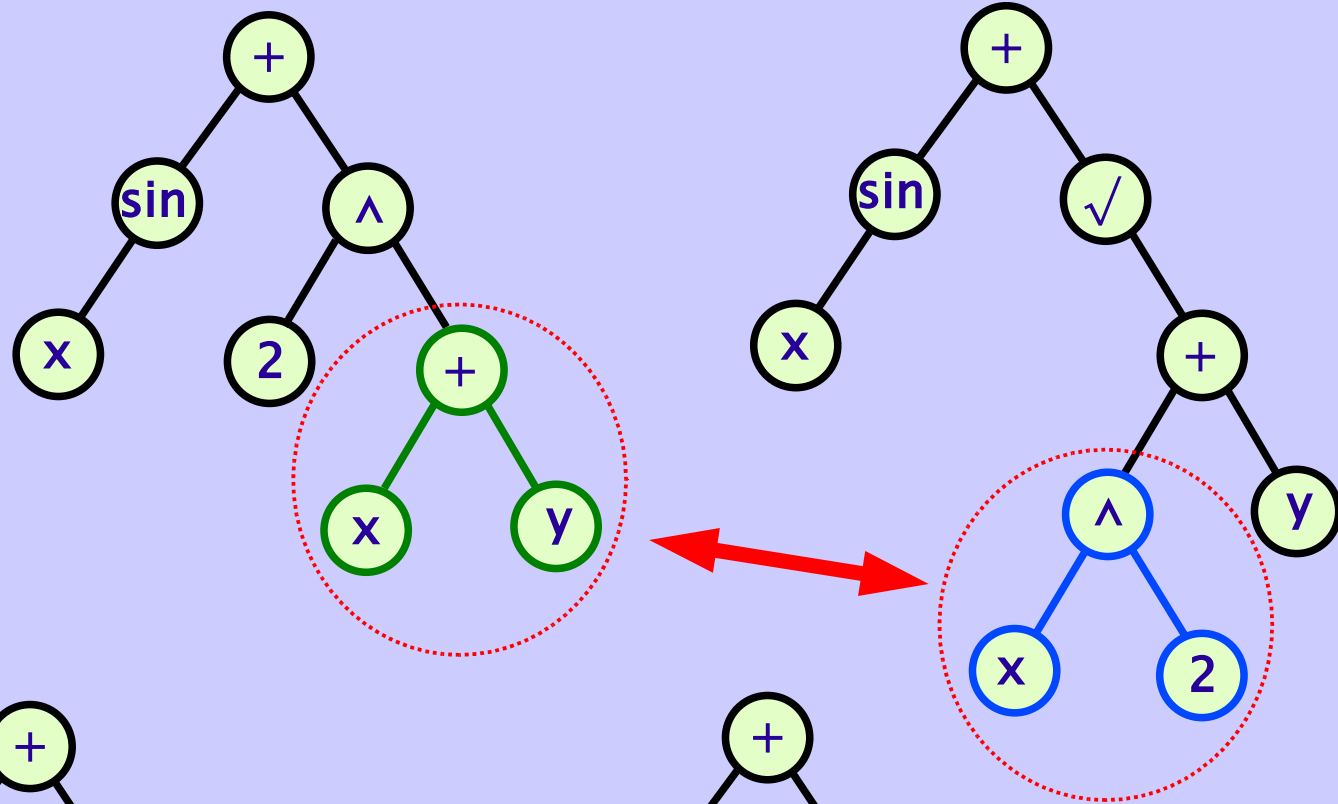
Chromosomy jako jedince v populaci lze popsat prostřednictvím stromové reprezentace funkcí, které lze sestavit z definovaných primitiv a terminálů, například:



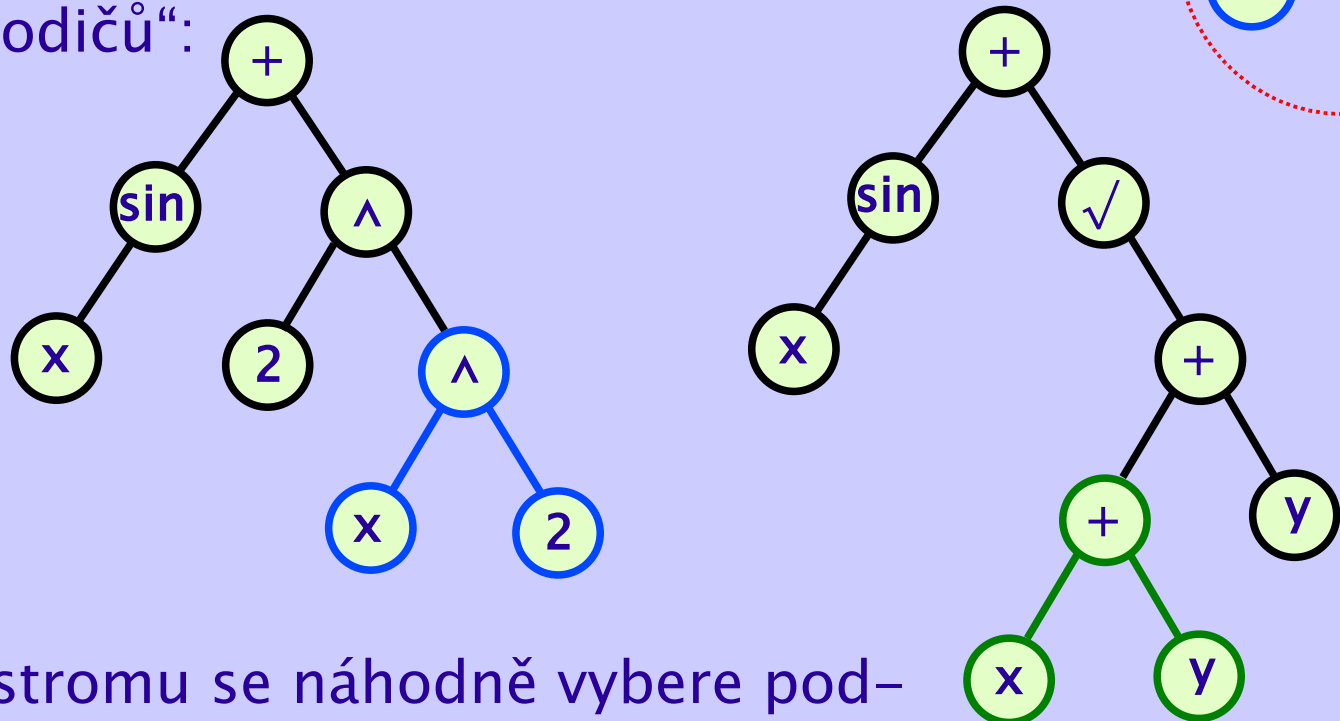
$$f(x, y) = \sin(x) + \sqrt{x^2 + y}$$

Křížení

Dvě „rodičovské“
původní funkce:



Nové funkce jako
„potomci“ vzniklí
křížením „rodičů“:



V každém stromu se náhodně vybere pod-
strom (kořen). Podstromy se pak vymění.

Genetické programování (pokračování)

Přizpůsobenost (*fitness*) všech funkcí v populaci se stanoví testem nad trénovacími daty—některé funkce například mohou mít lepší regresní koeficient než jiné, apod.

Po ohodnocení chromosomů (funkcí) jsou členové populace seřazeni a je na ně aplikován pravděpodobnostní výběr následovaný křížením.

Lze použít i mutaci, např. náhodně změnit terminální symbol (např. konstantu $2.11 \rightarrow 2.05$, apod., podobně primitivní funkci $\sin(x) \rightarrow \log(x)$ apod.). Mutace by neměly být příliš velké, jinak bude vývoj probíhat s větší náhodností—to platí pro metody GA obecně, protože je snaha vyhnout se náhodným výběrům a zároveň se vyhnout úplnému sekvenčnímu prohledávání všech možností.

Aplikace GA na problém obchodního cestujícího

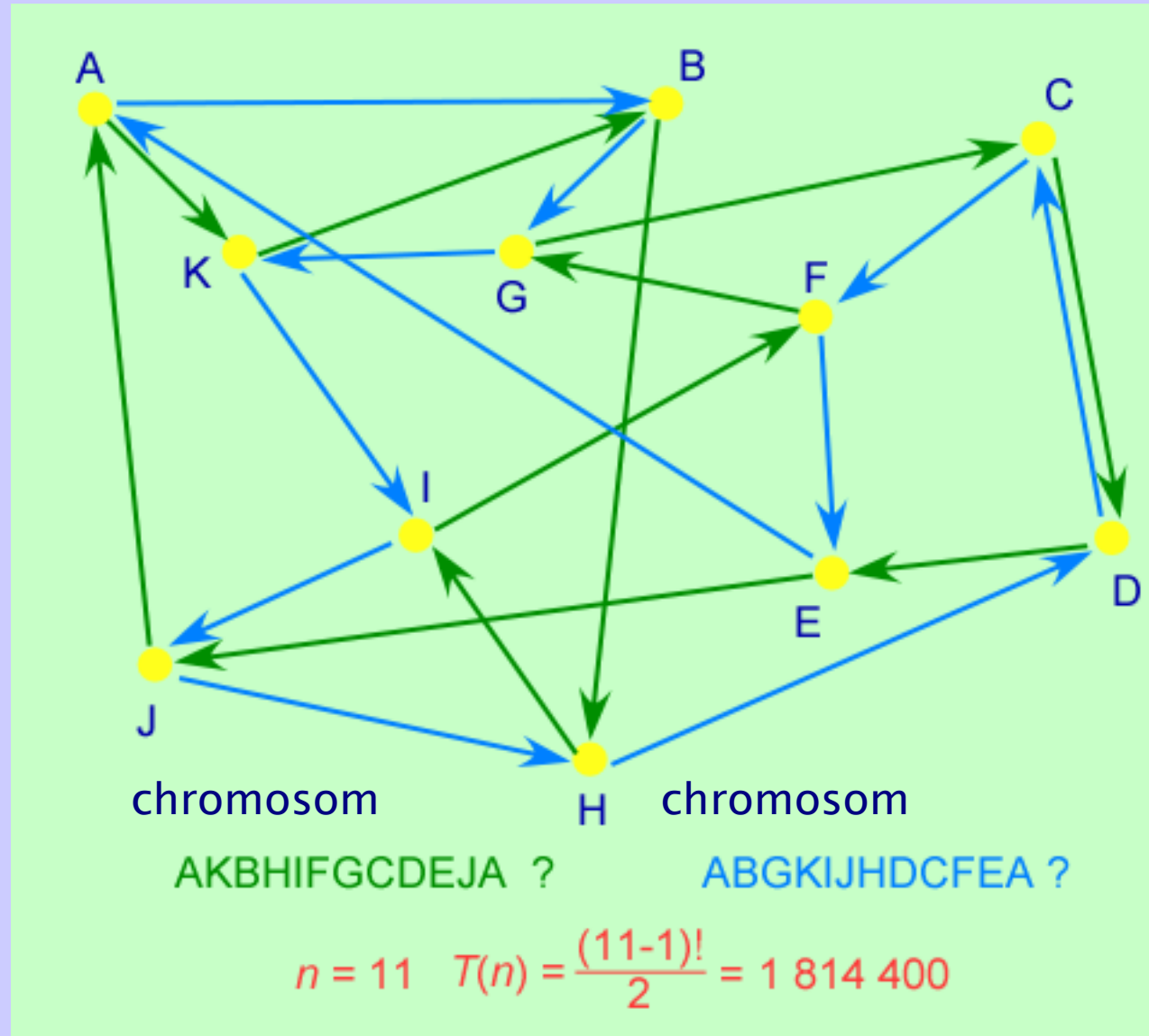
Jednou z úloh je *problém obchodního cestujícího* (*traveling salesman problem, TSP*). Je dána rozsáhlá množina bodů (měst) a úkolem je najít optimální cestu mezi nimi tak, aby obchodník přijel do každého města jen jednou a aby náklady na cestu byly minimální. Na tuto úlohu lze převést velké množství nejrůznějších problémů reálného světa (např. optimalizaci pohybu automatické vrtačky vytvářející otvory pro elektronické součástky na plošných spojích, propojování lokalit pro přenos signálu nebo energie, minimalizaci času a nákladů na včasné stanovení správné diagnózy a léčby pacienta, aj.). Není známo, jak pro desítky, stovky a více bodů najít optimální řešení. Genetické algoritmy jsou použitelné i pro tuto velmi obtížnou úlohu, i když také nemohou zaručit nalezení optima:

Pro n měst existuje celkem $T = (n-1)!/2$ různých řešení.

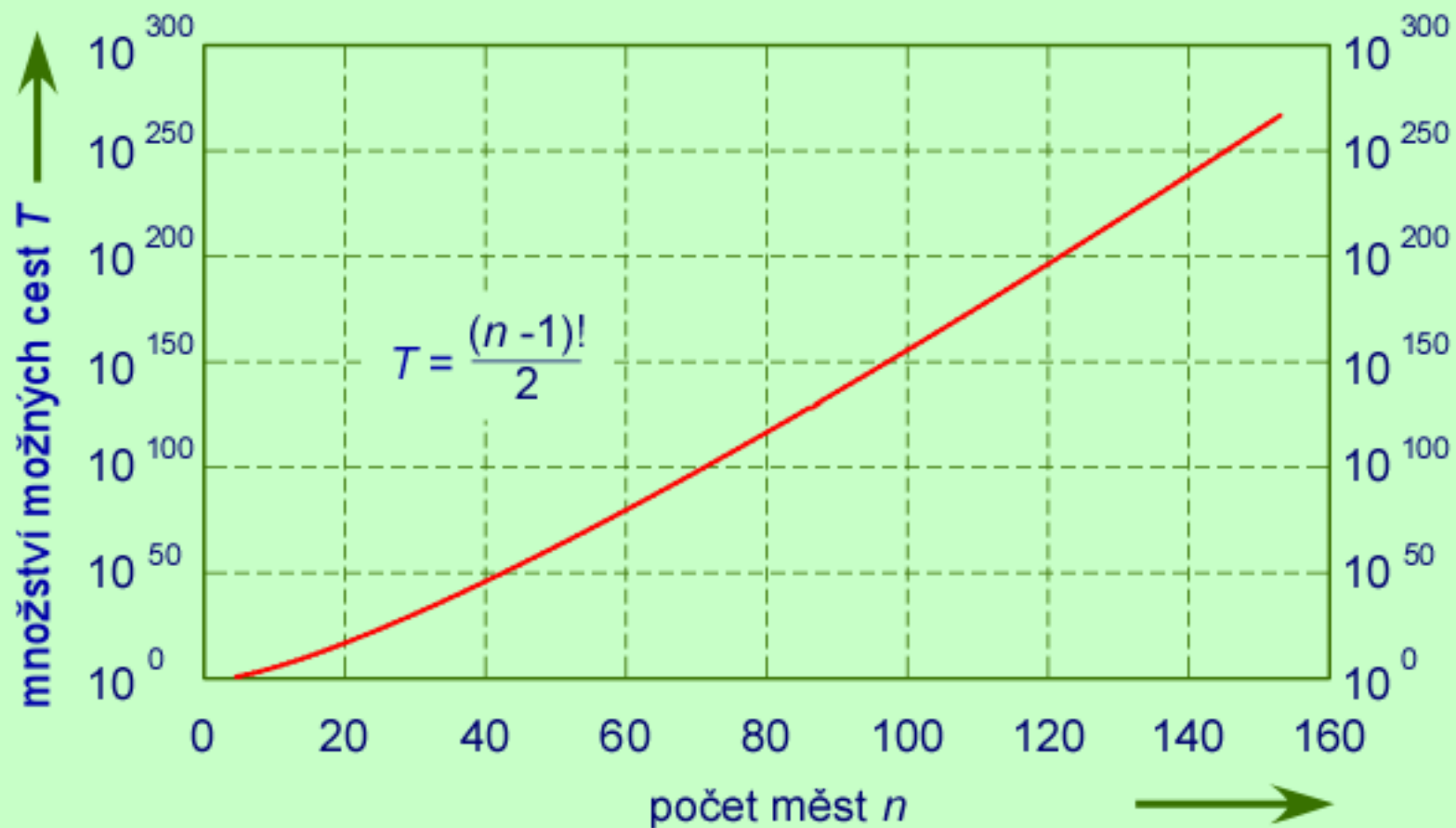
EVOLUTIONARY COMPUTATION

Snadno lze z existujících návrhů spojnic měst zjistit, která cesta je nejkratší, i pro velký počet měst. Není ale známo, jak najít optimální cestu. Množství různých řešení roste vlivem kombinatorické složitosti velmi rychle, silně nelineárně.

Výpočetní složitost (čas a paměť) neumožňuje prakticky pro n vyšší než pár desítek měst najít zaručeně optimální výsledek.



EVOLUTIONARY COMPUTATION

Problém obchodního cestujícího
(nelineární nárůst výpočetní složitosti)

Ukázka činnosti GA při řešení TSP

Demonstrační program **GATSP.exe**, vytvořený jako bakalářská práce studenta FI MU Tomáše Černého v r. 2004, umožňuje sledovat na grafickém výstupu činnost GA při řešení několika modelových úloh, na něž lze aplikovat TSP.

Program GATSP.exe umožňuje nastavit velké množství nejrůznějších parametrů genetických algoritmů a sledovat jejich vliv na řešení obtížné úlohy TSP.

GATSP.exe

EVOLUTIONARY COMPUTATION

Shrnutí

Genetické algoritmy jsou jednou z universálně použitelných metod na hledání vhodného nebo optimálního řešení mnoha reálných problémů, pokud nelze spolehlivě použít tradiční metody.

GA jsou výpočetně náročné a zahrnují nezanedbatelnou režii na nutnost oboustranné transformace parametrů řešeného problému pro stanovení přizpůsobenosti jednotlivých členů rozsáhlé populace.

GA obecně nezaručují nalezení optima (globálního extrému), přesto obvykle vedou k vylepšení výsledku. Jejich praktická aplikace na velmi mnoho rozličných problémů byla většinou úspěšná a stojí téměř vždy za pokus.

Základní literatura

Fogel, D. B. (2006) *Evolutionary Computation: Toward a New Philosophy of Machine Intelligence*. IEEE Press, NJ. Wiley-Interscience, John Wiley & Sons, Inc.

Goldberg, D. E. (1989) *Genetic Algorithms in Search, Optimization, and Machine Learning*. Reading, MA. Addison-Wesley.

Holland, J. H. (1992) *Adaptation in Natural and Artificial Systems*. Second Edition, Cambridge, MA. MIT Press.

Koza, J. R. (1992) *Genetic Programming*. Cambridge, MA. MIT Press.