

Genetické algoritmy a genetické programování

Josef Hynek

- Řešení úloh prostřednictvím soutěže v rámci populace vyvíjejících se potenciálních řešení
- Základní principy evolučních výpočetních technik
- Návrh efektivních algoritmů a možnosti využití
- Automatická tvorba programů na základě darwinovské evoluční teorie



Upozornění pro čtenáře a uživatele této knihy

Všechna práva vyhrazena. Žádná část této tištěné či elektronické knihy nesmí být reprodukována a šířena v papírové, elektronické či jiné podobě bez předchozího písemného souhlasu nakladatele. Neoprávněné užití této knihy bude **trestně stíháno**.

Používání elektronické verze knihy je umožněno jen osobě, která ji legálně nabyla a jen pro její osobní a vnitřní potřeby v rozsahu stanoveném autorským zákonem. Elektronická kniha je datový soubor, který lze užívat pouze v takové formě, v jaké jej lze stáhnout s portálu. Jakékoliv neoprávněné užití elektronické knihy nebo její části, spočívající např. v kopírování, úpravách, prodeji, pronajímání, půjčování, sdělování veřejnosti nebo jakémkoliv druhu obchodování nebo neobchodního šíření je zakázáno! Zejména je zakázána jakákoliv konverze datového souboru nebo extrakce části nebo celého textu, umísťování textu na servery, ze kterých je možno tento soubor dále stahovat, přitom není rozhodující, kdo takovéto sdílení umožnil. Je zakázáno sdělování údajů o uživatelském účtu jiným osobám, zasahování do technických prostředků, které chrání elektronickou knihu, případně omezují rozsah jejího užití. Uživatel také není oprávněn jakkoliv testovat, zkoušet či obcházet technické zabezpečení elektronické knihy.





Copyright © Grada Publishing, a.s.

Obsah

Předmluva	9
1. Úvod	13
Část I: Genetické algoritmy.....	17
2. Genetický algoritmus krok za krokem.....	19
3. Proč genetické algoritmy fungují?.....	27
4. Příklad – optimalizační úloha	33
5. Zvyšování účinnosti genetického algoritmu	41
5.1 Kódování	42
5.2 Ohodnocující funkce	43
5.3 Seleční mechanismus	45
5.4 Genetické operátory	50
5.5 Vytvoření nové populace.....	53
5.6 Paralelní genetické algoritmy	54
5.7 Závěr	57
6. Proč genetické algoritmy selhávají	59
6.1 Důvody pro možný neúspěch	59
6.2 Kdy genetický algoritmus použít?.....	65
7. Hybridní genetické algoritmy	67

8. Úlohy s omezujícími podmínkami	71
8.1 Klasifikace optimalizačních úloh	72
8.2 Ilustrativní příklad	74
8.3 Metody práce s omezujícími podmínkami	75
8.3.1 Penalizační metoda	75
8.3.2 Opravné algoritmy a speciální operátory	77
8.3.3 Dekodéry	79
8.4 Shrnutí	81
Část II: Evoluční algoritmy v úlohách s omezujícími podmínkami ...	83
9. Problém splňování logických formulí	85
9.1 Definice problému	86
9.2 Tradiční metody řešení	86
9.3 Genetické algoritmy	87
9.4 Hybridní genetický algoritmus	88
10. Problém N dam	91
10.1 Tradiční algoritmy pro řešení problému N dam	92
10.2 Genetické algoritmy	93
10.3 Hybridní genetické algoritmy	95
10.4 Návrat k heuristickému algoritmu	96
10.5 Shrnutí	98
11. Problém určení minimálního počtu kontejnerů	101
11.1 Definice problému	101
11.2 Tradiční algoritmy pro řešení problému	102
11.3 Genetické algoritmy	104
11.4 Hybridní evoluční algoritmus pro seskupovací problémy	105

12. Problém obchodního cestujícího	111
12.1 Definice problému	112
12.2 Tradiční algoritmy pro řešení problému	112
12.3 Genetické algoritmy pro problém obchodního cestujícího.....	114
12.4 Hybridní genetické algoritmy pro problém obchodního cestujícího....	118
Část III: Genetické programování	121
13. Genetické programování	123
13.1 Úvod do problematiky	123
13.2 Základy genetického programování	125
13.3 Vytvoření počáteční populace	127
13.4 Genetické operátory	129
14. Zvyšování účinnosti genetického programování.....	135
14.1 Nadbytečné části kódu	136
14.2 Automaticky definované funkce	137
14.3 Genetické programování s typováním	139
14.4 Využití cyklů v genetickém programování	142
15. Příklad – strojové učení.....	145
15.1 Zadání úlohy	146
15.2 Řešení pomocí genetického programování	146
15.3 Poznámky k průběhu hledání řešení	148
16. Příklad – umělý mravenec.....	151
16.1 Zadání úlohy	151
16.2 Řešení pomocí genetického programování	153
16.3 Poznámky k průběhu hledání řešení	155

17. Příklad – interaktivní evoluce	163
17.1 Zadání úlohy	164
17.2 Řešení pomocí genetického programování	165
17.3 Možnosti zlepšování navrženého řešení	167
Závěrečné slovo.....	169
Literatura	171
Rejstřík	179

Předmluva

V současné době patří nejrůznější heuristické metody založené na darwinovském principu evoluce mezi velice populární a často používané optimalizační techniky. Každoročně jsou organizovány desítky odborných konferencí zabývajících se problematikou evolučních algoritmů a jejich aplikací, vycházejí stovky publikací a někteří autoři dokonce již začínají varovat před přílišnou popularizací evolučních technik, která sice na jedné straně přitahuje nové uživatele, ale na straně druhé nutně vede k nerealistickým očekáváním.

Evoluční algoritmy jsou snadno aplikovatelné a do značné míry i univerzální, ale bez podrobné znalosti jejich fungování a podrobné analýzy problému nelze automaticky očekávat, že pouhou aplikací evolučních technik na libovolný problém je možné dosáhnout uspokojivých výsledků. Klíč k opravdu efektivnímu nasazení evolučních algoritmů je ukryt ve využívání maxima znalostí o struktuře a charakteru konkrétního problému, ve vzájemném prolínání evolučních algoritmů s tradičními technikami, které byly pro řešení těchto problémů využívány v minulosti, v procesu, který souhrnně nazýváme hybridizací. A právě k takovému pochopení možností praktického využití evolučních algoritmů by ráda přispěla i tato kniha.

Tomuto cíli byla podřízena i vlastní struktura knihy, která se dělí do tří částí, jež jsou dále členěny do jednotlivých kapitol. Po stručném úvodu do problematiky je ve druhé kapitole nejprve vysvětlen princip fungování jednoduchého genetického algoritmu a třetí kapitola objasňuje teoretické základy, o než se genetické algoritmy (byť s jistým zjednodušením) opírají. Čtvrtá kapitola ilustruje způsob aplikace genetického algoritmu na hledání maxima funkce dvou proměnných. Kapitola pátá potom rozšiřuje možnosti dříve prezentovaného jednoduchého genetického algoritmu z hlediska využití rozmanitých způsobů kódování, selekčních mechanismů, rekombinačních operátorů i strategií vytváření nové populace a nastiňuje též možnosti paralelizace příslušných výpočtů. V mnoha publikacích neprávem opomíjené problematice příčin možného selhání genetického algoritmu je věnována kapitola šestá, která současně nabízí některé možnosti, jak se těmto situacím včas a pokud možno i elegantně vyhnout. Sedmá kapitola, která první část této knihy uzavírá, je věnována možnostem hybridizace genetických algoritmů s jinými technikami, neboť právě touto cestou se celá oblast genetických algoritmů a jejich aplikací v současné době ubírá.

Posouzení reálných možností aplikace genetických algoritmů na úlohách s omezujícími podmínkami a ukázkám způsobu konstrukce hybridních evolučních algoritmů je věnována druhá část této knihy. V osmé kapitole je nejprve provedena klasifikace různých optimalizačních problémů a následně jsou zde popsány některé základní techniky pro práci s omezujícími podmínkami. V kapitole deváté až dvanácté jsou postupně popsány možnosti použití evolučních algoritmů při řešení čtyř úloh, které patří do třídy NP. S ohledem na snahu ukázat různé metody práce s omezujícími podmínkami a současně alespoň částečně představit obrovskou pestrost rozmanitých přístupů byl zde pro tyto účely využit problém splňování logických formulí, problém N dam, problém určení minimálního počtu kontejnerů a problém obchodního cestujícího. V každé z těchto kapitol jsou nejprve uvedeny různé heuristické metody, které se pro řešení dané úlohy používají, a poté je krok za krokem ukázán způsob hybridizace evolučního algoritmu s cílem nalezení co nejefektivnějšího výsledného algoritmu.

Třetí část knihy je věnována genetickému programování. Po úvodu do této problematiky v kapitole třinácté následuje kapitola s několika pokročilejšími tématy a návrhy řešení problémů, které jsou pro genetické programování specifické. Následující tři kapitoly ilustrují možnosti genetického programování na příkladu strojového učení, hledání programu pro umělého mravence a na závěr je uveden příklad interaktivní evoluce.

Souhrnně lze konstatovat, že hlavním záměrem v první části knihy bylo dát čtenáři do rukou co nejširší aparát nástrojů, které mu umožní proniknout do světa genetických algoritmů a aplikovat je na konkrétní problémy. Druhá část knihy by pak čtenáře měla inspirovat ke kritickému zamyšlení nad reálnými možnostmi genetických algoritmů a současně motivovat k úsilí využívat při jejich aplikaci hlubších znalostí o konkrétním problému. Třetí část potom zužitkovává principy genetických algoritmů, vyložené v části úvodní, a uvádí čtenáře do problematiky genetického programování, které se vyznačuje nejen odlišnou reprezentací individuí, ale i možnostmi použití.

Rozdílnému cíli všech tří částí této knihy byla přizpůsobena i použitá metodika. První a třetí část knihy vznikala jako produkt rozsáhlé literární rešerše a vlastního pedagogického působení autora knihy, s cílem co nejsrozumitelněji představit oblast genetických algoritmů a genetického programování. Vzhledem k nezbytnosti vyložit pojmy a techniky byla zvolena forma výkladu, která by měla být dobře čitelná i pro případného zájemce, který se snaží v dané oblasti rychle zorientovat.

Druhá část knihy byla psána se záměrem nikoliv porovnat efektivnost různých algoritmů na vybraných úlohách, ale zejména pokusit se u zde prezentovaných úloh vystopovat

evoluci jednotlivých přístupů, jelikož takto zprostředkovaný způsob myšlení může být velice užitečným zdrojem inspirace při návrhu hybridního evolučního algoritmu i pro zcela odlišný problém. Tímto úmyslem byl nutně ovlivněn výběr algoritmů, které jsou diskutovány u jednotlivých úloh, přičemž zejména mezi tradičními metodami řešení byly upřednostňovány takové algoritmy, které se podařilo využít při konstrukci hybridních evolučních algoritmů. V žádném případě nejde a nemůže jít o vyčerpávající přehled všech metod a pokusů, které kdy byly učiněny. Spíše šlo o to, ukázat takové algoritmy (ať již tradiční nebo genetické), které určitým způsobem přispěly k hledání nových postupů a umožnily najít dokonalejší metody pro řešení jednotlivých problémů.

Nejlepší algoritmy uváděné v druhé části této knihy jsou či byly pro daný problém v době psaní této knihy ty nejlepší známé a vzhledem k jejich charakteru je pravděpodobné, že budou dále zdokonalovány. V této souvislosti je třeba zopakovat a zdůraznit, že cílem nebylo najít a popsat nejlepší algoritmy pro zde popisované problémy k určitému datu, ale ukázat postup myšlení, kterým je třeba se ubírat při hledání a návrhu efektivního algoritmu pro určitý problém.

Z důvodu čitelnosti a srozumitelnosti bylo záměrně ustoupeno od přesné formulace algoritmů pro řešení jednotlivých úloh, čímž by výsledný text byl zbytečně zatížen a rozkouskovaný, přičemž hlavní myšlenka hybridizace jako způsobu návrhu skutečně efektivních evolučních algoritmů by se mohla v technických detailech jednotlivých algoritmů snadno ztratit. Každý algoritmus je proto pouze stručně popsán a současně je uveden odkaz, který zájemce o daný algoritmus pro řešení konkrétního problému dovede k pramenům, kde je tento algoritmus jednoznačně a podrobně definován. Analogicky nejsou v této knize publikovány podrobné výsledky relevantních experimentů. Je tomu tak jednak pro jejich rozsáhlost, ale i pro poměrně značnou neporovnatelnost, pramenící z různých hardwarových platforem, různých implementací evolučních algoritmů, odlišné množiny testovacích příkladů či nemožnosti zjistit konkrétní nastavení jednotlivých parametrů. V případech, kde již určité porovnání bylo provedeno, je v textu uveden příslušný závěr včetně relevantního odkazu.

V neposlední řadě je třeba objasnit volbu konkrétních problémů, na nichž jsou v druhé části této knihy možnosti aplikace evolučních algoritmů demonstrovány. Problém splňování logických formulí byl do této množiny zařazen zcela automaticky jako nejstarší a nejnámější NP-úplný problém. Problém N dam je další typickou netriviální úlohou, na které jsou testovány různé algoritmy pro řešení problémů, navíc právě tento problém byl autorem intenzivně zkoumán, přičemž výsledkem byl návrh heuristického algoritmu, který umožňuje řešit i poměrně rozsáhlé instance tohoto problému. Problém určení minimálního počtu kontejnerů byl vybrán jako zástupce celé škály seskupovacích problémů, pro něž se podařilo právě vhodnou kombinací tradičních technik operačního výzkumu a evolučních algoritmů navrhnout velmi efektivní hybridní algoritmus. Problém obchodního cestujícího je jedním z nejčastěji studovaných kombinatorických problémů a do této knihy byl zařazen zejména proto, že v oblasti evolučních algoritmů se celá řada instancí tohoto problému užívá jako obecně akceptovaný benchmark. Volbou uvedených problémů jsou zde navíc stejným dílem zastoupeny problémy splňování omezujících podmínek (*constraint satisfaction problems*) i problémy hledání vázaného extrému (*constraint optimization problems*).

Úlohy, na nichž jsou ilustrovány možnosti genetického programování, byly záměrně voleny tak, aby bylo možné naznačit nejen velmi široké pole aplikací, ale zejména popsat odlišnosti pramenící z charakteru jednotlivých úloh a zvláště pak způsobů hodnocení individuů. Proto zde čtenář nalezne vedle symbolické regrese i úlohu, ve které se hledá program pro navigaci robota, a příklad interaktivní evoluce, kdy hodnocení individuů zprostředkovává sám uživatel.

Jak již bylo uvedeno výše, přestože tato kniha popisuje desítky různých algoritmů, neměla by sloužit jen jako návod, jak tyto algoritmy použít pro řešení zde uvedených problémů. Mnohem významnější a důležitější jsou odhalené způsoby myšlení a naznačené příležitosti, které popisované evoluční algoritmy spolu s rostoucími možnostmi informačních technologií v současné době nabízejí.

1.

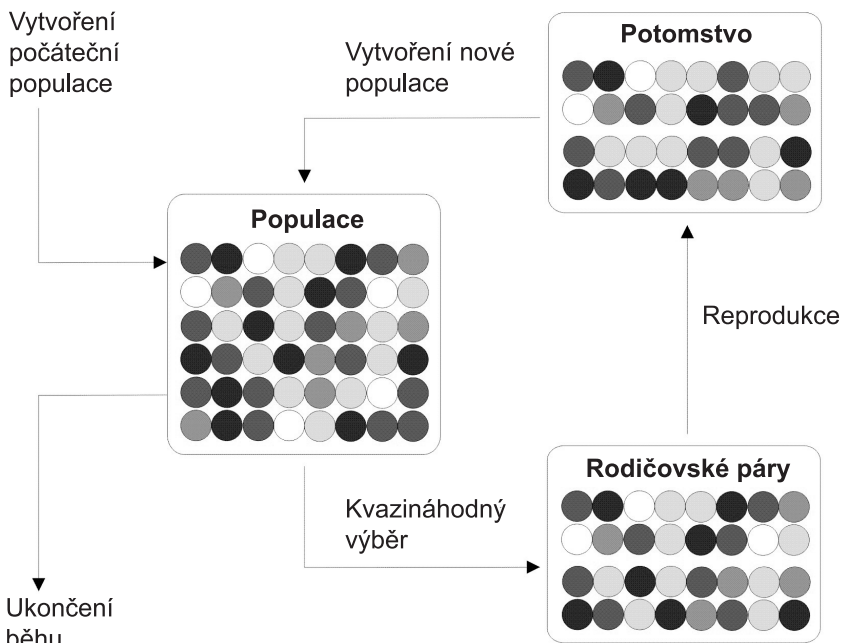
Úvod

Při hledání skutečných kořenů, z nichž vyrůstá celá oblast evolučních algoritků, je třeba se vrátit zpět až do roku 1859, kdy Charles Darwin poprvé publikoval svoji proslulou knihu *O vzniku druhů přirozeným výběrem čili zachováním vhodných odrůd v boji o život*. Darwinova myšlenka, že populace živočichů a rostlin se vyvíjela po mnoho generací podle principu přirozeného výběru a přežití těch nejschopnějších, inspirovala po více než jednom století velice slibně se rozvíjející oblast výzkumu, jehož cílem je tyto přírodní procesy napodobit. Ačkoliv počátky rozmanitých evolučních procesy inspirovaných paradigmat řešení problémů lze vysledovat již na konci 50. let dvacátého století, tato oblast zůstávala další tři desetiletí stranou většího zájmu vědecké komunity. Tento stav byl zapříčiněn nejen nedostatečným výpočetním výkonem tehdejších počítačů, ale též počátečními metodologickými nedostatky jednotlivých přístupů [44]. V posledním desetiletí dvacátého století se však již evoluční algoritmy, což jsou stochastické optimalizační metody založené na darwinovském principu evoluce, staly velice populárními a obecně akceptovanými metodami, které dnes běžně používají vědci i praktičtí specialisté k řešení problémů v celé řadě nejrůznějších oborů. Pod obecný pojem evoluční algoritmy se běžně zařazují jako hlavní metodologie genetické algoritmy, evoluční strategie, evoluční programování a genetické programování. Přestože v této knize bude hlavní pozornost věnována ge-

netickým algoritmům a genetickému programování, někdy bude záměrně použit termín algoritmy evoluční. Je tomu tak zejména proto, že i přes ne zcela přesné vymezení pojmů v této oblasti celá řada algoritmů prezentovaných zejména v druhé části této knihy svým charakterem přesahuje rámec běžného chápání pojmu genetický algoritmus (například porušením principu pevné délky chromozomů atd.).

Průkopníkem v oblasti genetických algoritmů se stal John Holland, který studoval elementární procesy v populacích, jež jsou z hlediska evoluce nepostradatelné. Na základě těchto výzkumů navrhl genetický algoritmus jako abstrakci příslušných biologických procesů [65]. V přírodě obvykle individua v rámci dané populace soupeří o zdroje potřebné k přežití, chrání se před predátory a usilují o možnost vlastní reprodukce. Nejúspěšnější jedinci, kteří obstojí v tomto náročném zápasu, mají potom příležitost žít déle, nalézt a přilákat vhodného partnera a následně společně zanechat relativně větší počet potomků než méně úspěšní jedinci, kteří mají omezenější možnosti se reprodukce účastnit, a počet jejich potomků je potom menší, nebo dokonce zemřou bez potomstva. Použitím tohoto principu je možné dosáhnout toho, že vlastnosti schopnějších či lépe přizpůsobivých jedinců zakódované v relevantní genové výbavě se předávají rostoucímu počtu potomků v dalších generacích. Navíc je zřejmé, že vhodnou kombinací vlastností rodičovského páru lze docílit toho, že jejich potomek (nebo potomci) bude mít lepší vlastnosti než kterýkoliv z obou rodičů a bude tedy lépe přizpůsoben životu v daném prostředí.

Hollandův genetický algoritmus je založen na přímé analogii s právě popsány procesy, které Darwin odhalil v živé přírodě. Genetický algoritmus proto pracuje s populací jedinců, přičemž každý z těchto jedinců reprezentuje vhodným způsobem zakódované řešení daného problému. Každému individu je přiřazeno jisté nezáporné ohodnocení, jež kvantitativně vyjadřuje míru vlastností konkrétního jedince vzhledem k řešenému problému a současně je i měřítkem reprodukční způsobilosti tohoto individua. Použitím imitace náhodného procesu přirozeného výběru spolu s geneticky inspirovanými operátory křížení,



Obrázek 1.1: Schematické znázornění funkce genetického algoritmu

mutace a inverze je dána individuům možnost reprodukce. V závislosti na použité strategii potom z původní populace a jejího potomstva vzniká populace nová (viz obrázek 1.1) a tím je jeden generační cyklus ukončen.

Opakuje-li se tento proces po mnoho generací, zdatnější jedinci (to jest lepší řešení daného problému) jsou při výběru upřednostňováni, jejich vlastnosti se šíří v rámci populace a kombinují se s vlastnostmi dalších zdatných jedinců. Po určitém počtu generací potom tímto způsobem obvykle vznikne populace s jedním či více jedinci, kteří odpovídají přijatelnému (nebo dokonce optimálnímu) řešení daného problému. Samozřejmě: protože tento algoritmus je stochastický, nelze garantovat, že jeho použitím bude nalezeno řešení optimální, ale ukazuje se, že genetické algoritmy jsou obecně vhodné k „nalezení přijatelného řešení v přijatelném čase“ [8].

Při porovnání genetického algoritmu s tradičními optimalizačními technikami lze snadno nalézt několik výrazných odlišností, které významným způsobem přispívají k univerzálnosti a robustnosti genetického algoritmu. Prvním zřejmým rozdílem je využití celé populace řešení z prohledávaného prostoru namísto jediného počátečního řešení, jak je tomu u většiny klasických algoritmů. Tato skutečnost podstatně ovlivňuje celkovou robustnost algoritmu a současně přímo vybízí k paralelizaci vlastního výpočtu.

Genetický algoritmus bez dalších modifikací, kterým bude věnována zejména druhá část této knihy, je algoritmem slepým, protože kromě ohodnocení jedinců v populaci nevyužívá žádnou apriorní informaci či znalost vážící se ke konkrétnímu problému nebo relevantnímu prohledávanému prostoru. To jej na jedné straně činí velmi obecným a univerzálním, na straně druhé je zřejmé, že právě toto ignorování specifických vlastností konkrétního problému bude znamenat, že algoritmus bude jen těžko schopen konkurovat speciálními algoritmy, které byly navrženy přímo pro daný problém.

Shrnutím předchozích řádků lze konstatovat, že síla genetických algoritmů spočívá v jejich robustnosti a univerzálnosti použití. Tento fakt ale rozhodně neznamená, že genetické algoritmy jsou technikou, která je snadno a okamžitě použitelná pro jakýkoliv problém. Jak již zde bylo uvedeno, existuje-li pro konkrétní problém specializovaný algoritmus, který využívá určité znalosti o daném problému, takový algoritmus obvykle předčí genetický algoritmus jak v rychlosti, tak i v kvalitě nalezeného řešení. Doménou genetických algoritmů tedy budou zejména problémy, k jejichž řešení žádná speciální technika není k dispozici nebo není ani známa.

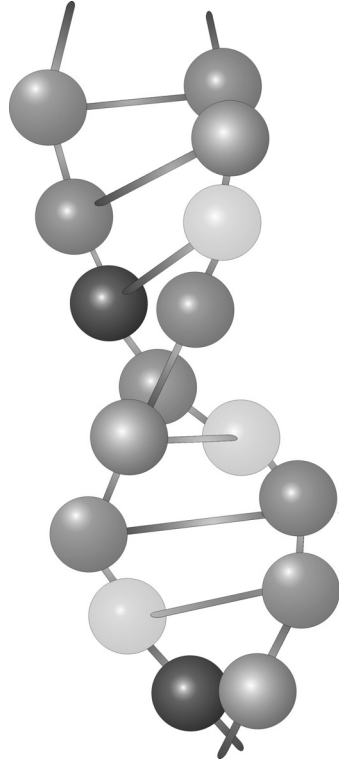
Na straně druhé se v případě některých opravdu obtížných problémů zřetelně ukazuje, že vhodnou kombinací genetických algoritmů a dosud používaných speciálních technik, kdy využitím robustnosti genetického algoritmu, jež je doplněna o know-how konkrétního algoritmu pro daný problém, je možné dosáhnout velmi zajímavých výsledků. Příklady takovýchto postupů uvedeme v druhé části této knihy.

Genetické programování je rozšířením genetických algoritmů v tom smyslu, že zatímco genetické algoritmy pracují s individui, která jsou reprezentována chromozomy pevné délky, v tomto případě se jedná o hierarchicky strukturované počítačové programy, které po spuštění mohou představovat potenciální řešení daného problému. Ačkoliv se tento nápad objevil poprvé již v polovině devadesátých let minulého století [20], za pomyslného otce genetického programování je považován stanfordský profesor John Koza [76, 77]. Právě Koza vytvořil metodologii, která umožňuje pěstovat počítačové programy pomocí podobných principů, jaké používají genetické algoritmy – s tím, že výsledkem evolučního procesu je program, který řeší (či přibližně řeší) konkrétní problém. Kromě odlišného způsobu reprezentace individuí je podstatným rozdílem samozřejmě i oblast aplikací a genetické programování bylo s úspěchem použito k řešení celé řady problémů. Koza [80] uvádí více než dvě desítky příkladů, kdy byl s využitím genetického programování „znovuobjeven“

či dokonce vylepšen už patentovaný vynález, přičemž v šesti případech šlo o vynález patentovaný po 1. lednu 2000. Ve dvou případech genetické programování dokonce vytvořilo zcela nový patentovatelný vynález (v obou případech jde o regulátor, který má lepší vlastnosti než dosud běžně používané regulátory). Pro korektnost je třeba dodat, že většiny z těchto výsledků bylo dosaženo na vysoce paralelních počítačích (například tisíciprocessorový stroj) a potřebný výpočetní čas se pohyboval v řádu hodin až několika dnů (při vynálezu regulátoru šlo dokonce o běh trvající čtyři týdny). Příklady uvedené ve třetí části této knihy mají spíše ilustrativní charakter, ale na druhé straně jsou snadno realizovatelné i na běžném osobním počítači.

Genetické algoritmy

část I.



2.

Genetický algoritmus krok za krokem

Genetické algoritmy jsou založeny na myšlence darwinovského principu evoluce. Hledání optimálního (nebo alespoň dostatečně vyhovujícího) řešení probíhá formou soutěže v rámci populace postupně se vyvíjejících řešení. K tomu, aby bylo možné posoudit, kteří členové populace mají větší šanci podílet se na dalším vývoji hledaného řešení, musí být tato schopnost individuí kvantifikovatelná. V této souvislosti se mluví o ohodnocení, míře kvality, vhodnosti, síle či reprodukční schopnosti individua (v zahraniční literatuře se ve směr používá pojem *fitness*). Jedinci s lepším ohodnocením (vyšší hodnotou *fitness*) mají přirozeně větší šanci přežít déle a podílet se na vytváření následující generace. Použitím rozmanitých technik křížení a reprodukce potom vznikne nová generace individuí, ve které jsou vlastnosti jedinců částečně zděděny po rodičích a částečně ovlivněny náhodnými mutacemi v procesu reprodukce. Opakuje-li se tento evoluční cyklus mnohokrát, obvykle po desítkách až stovkách opakování vznikne populace s jedinci, kteří mají vysoké ohodnocení a mohou představovat dostatečné (či dokonce optimální) řešení daného problému. Protože tento evoluční proces v sobě zahrnuje značný díl náhodnosti, je zřejmé, že každý běh příslušného algoritmu se bude odvíjet odlišným způsobem. Z téhož důvodu se v některých případech poměrně snadno může stát i to, že celá populace v procesu vývoje zdegeneruje a nejlepší jedinec bude reprezentovat pouze lokální optimum, které

může být velmi odlišné od optima globálního. Proto se chování genetických algoritmů na konkrétní úloze obvykle popisuje prostřednictvím různých statistik, které shrnují nejlepší, nejhorší a průměrné hodnoty sledovaných ukazatelů.

V literatuře lze nalézt různé definice genetického algoritmu, které se liší zejména ve způsobu vytváření nové populace, obecné schéma je však ve většině případů následující:

1. Vynuluj hodnotu počítadla generací $t=0$.
2. Náhodně vygeneruj počáteční populaci $P(0)$.
3. Vypočítej ohodnocení (*fitness*) každého individua v počáteční populaci $P(0)$.
4. Vyber dvojice individuí z populace $P(t)$ a vytvoř jejich potomky $P'(t)$.
5. Ohodnoť nově vytvořená individua v $P'(t)$.
6. Vytvoř novou populaci $P(t+1)$ z původní populace $P(t)$ a množiny potomků $P'(t)$.
7. Zvětšij hodnotu počítadla generací o jedničku ($t=t+1$).
8. Vypočítej ohodnocení (*fitness*) každého individua v populaci $P(t)$.
9. Pokud t je rovno maximálnímu počtu generací nebo je splněno jiné ukončovací kritérium, vrať jako výsledek populaci $P(t)$; jinak pokračuj krokem číslo 4.

K podrobnějšímu popisu a vysvětlení tohoto algoritmu je vhodné na tomto místě zavést několik biologických termínů, které budou v jistém přeneseném smyslu dále v této knize používány.

Nejprve je nutné si uvědomit zásadní rozdíl mezi individuem (fenotyp) a jeho reprezentací (genotyp), která zde bude označována též jako chromozom. Toto zjednodušení může být pro někoho částečně matoucí, protože v přírodě každá buňka organismu daného druhu nese určitý počet chromozomů (obvykle více než jeden). V celé této knize však budou uvažována pouze taková individua, která jsou charakterizována jediným chromozomem, a proto je výše uvedené zjednodušení *chromozom = genotyp* zcela korektní a přípustné. V této souvislosti je třeba též konstatovat, že tento způsob abstrakce je vlastně naprosto většinou prací věnovaných problematice genetických algoritmů [96, s. 6].

Chromozom se potom dále dělí na jednotlivé geny, které jsou lineárně uspořádány. To znamená, že jsou-li dva chromozomy stejného typu, potom i -tý gen bude reprezentovat stejnou charakteristiku v obou chromozomech (například třetí gen bude ovlivňovat barvu očí, čtvrtý barvu vlasů atd.). A konečně, aby mohl gen určovat příslušnou vlastnost, musí mít možnost nabývat různých hodnot (nacházet se tedy v různých stavech), kterým se říká alely (např. 0 = bílá, 1 = šedá, atd.).

V předchozím odstavci je obsaženo další záměrné zjednodušení, protože v případě živých organismů je zřejmé, že jednotlivé geny ovlivňují daleko jemnější charakteristiky a teprve společným a vzájemným působením mnoha genů je určena jistá vlastnost organismu. Na straně druhé se však podle posledních výzkumů zdá být pravděpodobné, že při sekvencování lidského genomu vědci izolovali určité geny, které hrají významnou roli v náchylnosti k některým chorobám. Například firma Myriad Genetics prodává testy na mutace v genech označovaných jako *BRCA1* a *BRCA2*, které umožňují včas diagnostikovat predispozice žen k rakovině prsu, gen *apoE* je potom ze zatím neznámého důvodu významným indikátorem výskytu Alzheimerovy choroby v pozdějším věku atd.¹

¹ Lidský genom. Soubor studií převzatý z týdeníku The Economist (zvláštní příloha publikovaná 1. 7. 2000). *Ekonom* 28, 2000.

2. Genetický algoritmus krok za krokem

S tímto pojmovým aparátem je možné přikročit k podrobnějšímu výkladu obecného algoritmu, který byl prezentován výše. Je zřejmé, že prvním úkolem je náhodně vygenerovat počáteční populaci. Z předchozího textu vyplývá, že každé individuum je popsáno svým chromozomem, který má pevnou délku a určené pevné uspořádání genů.

Jak bude ukázáno v dalších částech této knihy, již vlastní způsob, jakým jsou individua zakódována (geneticky popsána), je velmi důležitý pro úspěch či neúspěch genetického algoritmu na konkrétní úloze. Pro účely této úvodní kapitoly předpokládáme binární kódování, tedy že jednotlivá individua budou reprezentována chromozomem určeným dvojkovým řetězcem dané délky. Toto kódování je historicky nejstarší [65] a z celé řady důvodů je stále jedním z nejpoužívanějších druhů kódování [96, s. 156–7]. V dalších částech této knihy budou uvedeny i některé další způsoby kódování jedinců a budou diskutovány i jejich konkrétní výhody oproti tomuto nejjednoduššímu způsobu.

Binární kódování znamená, že každý gen může nabývat jen dvou různých stavů: 0 nebo 1. V tomto případě bude tedy počáteční populaci tvořit množina náhodně vygenerovaných binárních řetězců dané délky. Aby bylo možné další výklad ilustrovat na jednoduchém příkladu, předpokládáme, že náhodně vygenerovaná populace se skládá ze čtyř individuí, přičemž každé z nich je reprezentováno chromozomem délky 8 následovně:

<i>Individuum č.</i>	<i>Chromozom</i>
1	(1, 0, 1, 0, 1, 1, 0, 0)
2	(0, 1, 1, 1, 1, 0, 1, 1)
3	(0, 0, 0, 1, 0, 0, 0, 1)
4	(1, 1, 0, 0, 1, 1, 0, 0)

Dalším krokem ve výše uvedeném algoritmu je ohodnocení každého individua v populaci. Je zřejmé, že jakákoliv smysluplná definice vhodné ohodnocující funkce bude přirozeně dána konkrétním problémem, protože cílem je najít takové individuum či individua, která mají z hlediska příslušného problému co možná nevhodnější (nebo dokonce optimální) vlastnosti.

V následujících kapitolách bude uvedena celá řada příkladů konkrétních a smysluplných ohodnocujících funkcí, ale v zájmu jednoduchosti výkladu předpokládáme, že ohodnocení každého individua bude rovno počtu jednotkových bitů v jeho reprezentaci. Jinými slovy: preferována a považována za zdatnější budou individua obsahující ve svém genotypu více jedniček. Ohodnocení jednotlivých chromozomů ve zde uvažované populaci bude tedy následující:

<i>Individuum č.</i>	<i>Chromozom</i>	<i>Ohodnocení</i>
1	(1, 0, 1, 0, 1, 1, 0, 0)	4
2	(0, 1, 1, 1, 1, 0, 1, 1)	6
3	(0, 0, 0, 1, 0, 0, 0, 1)	2
4	(1, 1, 0, 0, 1, 1, 0, 0)	4

A zde přichází první klíčový okamžik práce genetického algoritmu, kdy je nutné vybrat dvojice individuí (respektive odpovídající chromozomy) z této populace za účelem jejich reprodukce. Tento výběr by měl být imitací přirozeného výběru, jak je popsán v Darwinově teorii o původu druhů. To znamená, že zdatnější jedinci mají větší šanci vzdorovat predátorům, překonat nemoci a další překážky, žít dostatečně dlouho a zplodit potomky.

Zřejmě nejrozšířenější formou implementace přirozeného výběru je použití takzvaného ruletového mechanismu selekce. Na rozdíl od klasické rulety, kde každé z čísel může být kvůli ekvivaletnímu obsahu přidělených výšečí vybráno se stejnou pravděpodobnos-

tí, v tomto případě budou favorizována kvalitnější individua (to jest individua s vyšším ohodnocením). Takovým jedincům je na ruletovém kole přiřazena větší výšeč a existuje tedy větší pravděpodobnost, že se při hře kulička zastaví právě v jejich výšeči. Slabší individua mají přiřazenu menší výšeč, což je sice předem nevylučuje, ale pravděpodobnost, že se kulička zastaví v jim přiřazené výšeči a dojde tak k vybrání příslušného individua, je menší.

Existuje několik běžně používaných způsobů, jak „upřednostňující“ ruletové kolo zkonstruovat. Liší se zejména ve způsobu určování velikosti výšečí pro jednotlivá individua. Velmi rozšířený je tzv. ruletový mechanismus selekce s pravděpodobností výběru individua přímo úměrnou jeho ohodnocení (tzv. *fitness-proportionate selection*), kde – jak již název napovídá – má každé individuum přiřazenou výšeč, jejíž plocha je přímo úměrná ohodnocení tohoto individua.

K vytvoření takového ruletového kola stačí sečíst ohodnocení všech členů populace a potom přiřadit každému jedinci proporcionálně odpovídající kruhovou výšeč. Nechť má uvažovaná populace velikost $N > 0$ a $f_i \geq 0$ ($i = 1, \dots, N$), nechť je ohodnocení i -tého individua (předpokládáme, že ohodnocující funkce je vždy nezáporná), potom pravděpodobnost, s jakou bude i -tý jedinec vybrán (a odpovídající velikost kruhové výšeče), je dána následujícím vztahem:

$$p_i = \frac{f_i}{\sum_{j=1}^N f_j} \quad i \in \{1, \dots, N\}$$

Pro zde prezentovaný ilustrativní příklad to konkrétně znamená rozdělit ruletové kolo na čtyři výšeče odpovídající svou plochou ohodnocení příslušných individuí tak, jak je níže vypočteno a poté naznačeno na obrázku 2.1.

Individuum č.	Chromozom	Ohodnocení (f_i)	% z celkového ohodnocení (p_i)	Kumulované ohodnocení
1	(1, 0, 1, 0, 1, 1, 0, 0)	4	25,0%	0,250
2	(0, 1, 1, 1, 1, 0, 0, 1)	6	37,5%	0,625
3	(0, 0, 0, 1, 0, 0, 0, 1)	2	12,5%	0,750
4	(1, 1, 0, 0, 1, 1, 0, 0)	4	25,0%	1,000
Průměrné ohodnocení populace		4		

Odtud je zcela zřetelně vidět, že jedinec číslo 2 má mnohem větší šanci být vybrán než například jedinec číslo 3. Ale protože jde o náhodný proces, není ještě vše ztraceno ani pro toto – v naší populaci nejslabší – individuum. Stejně jako v přírodě se stává, že některá i velmi silná individua se neprosadí, nenajdou vhodného partnera či partnerku, nebo předčasně zahynou, zatímco i relativně slabý jedinec může přežít a účastnit se reprodukčního procesu.

Je-li ruletové kolo zkonstruováno tímto způsobem, potom stačí pouze při každém výběru jedince z populace vygenerovat náhodné číslo $r \in (0,1)$ a vybrat i -tého jedince, právě když platí vztah:

$$\sum_{j=1}^{i-1} p_j < r \leq \sum_{j=1}^i p_j \quad i \in \{1, \dots, N\}$$

kde N je velikost populace. Jelikož by nebylo efektivní počítat výše naznačené sumy opakovaně pro každý jednotlivý výběr individua, je užitečné zavést souhrnné (kumulativní) ohodnocení, které je definováno jako:

2. Genetický algoritmus krok za krokem