

České vysoké učení technické v Praze,
fakulta strojní

Czech Technical University in Prague,
Faculty of Mechanical Engineering

Doc. Ing. Pavel Ošmera, CSc.

Evoluční algoritmy a jejich aplikace

Evolutionary algorithms and their applications

Summary

Evolutionary algorithms (EAs) are an active research area. In contrast to other optimization techniques, Evolutionary Algorithms base their progress on the performance of the population of candidate solution, rather than on a single candidate solution. A solution is represented by a string of symbols, and is analogous to chromosome. In general a genome representation should be the simplest possible encoding of the optimized problem. Optimization techniques model principles of natural evolution. Evolutionary algorithms apply selection and mutation operators to a population of states to guide the population to an optimal solution of the objective function. An objective function called fitness is evaluated for each chromosome.

Evolutionary computation is generally considered as a consortium of genetic algorithms (GA), genetic programming (GP) and evolutionary strategies (ES). There are new GPs: grammatical evolution and chemical genetic programming.

We have described a parallel system of Parallel Grammatical Evolution (PGE) that can map a binary genotype onto a phenotype which is a program. PGE has been successfully applied for symbolic regression. Parallel GE with a backward processing can increase the efficiency and robustness of systems, and thus they can track better optimal parameters in a changing environment. The parallel grammatical evolution can be used for the automatic generation of programs. We are far from supposing that all difficulties are removed but first results with PGE are very promising.

In the last years there is an exponential increasing number of practical applications of EAs.

Souhrn

Evoluční algoritmy (EA) jsou v současné době v popředí výzkumného zájmu. Na rozdíl od jiných optimalizačních metod EA mají svůj progres založen na vlastnostech populace kandidujících řešení, spíše než na jednom kandidujícím řešení. Hledané řešení je reprezentováno pomocí řetězce symbolů (genů), který je analogií chromozómu. Všeobecně by genová reprezentace problému měla být co nejjednodušší možné zakódování optimalizovaného problému. Evoluční optimalizační techniky modelují principy přírodní evoluce. Evoluční algoritmy používají operátory selekce a mutace na populaci stavů aby populace směřovala k optimálnímu řešení. Účelová funkce, zvaná fitness, slouží k ohodnocení každého chromozómu.

Evoluční výpočty jsou všeobecně považovány jako sdružení genetických algoritmů (GA), genetického programování (GP) a evolučních strategií (ES). Existují však nové metody GP: gramatická evoluce (GE) a chemické genetické programování (CHGP).

Byl popsán paralelní systém gramatické evoluce (PGE), který mapuje binární genotyp na fenotyp, který je vlastně generovaný program. PGE byla úspěšně použita na problém symbolické regrese. Paralelní GE se zpětným zpracováním může zvýšit efektivnost a robustnost systému a také může lépe sledovat optimální parametry měnícího se prostředí. PGE může být použita pro automatické generování programů. Jsme daleko toho prohlásit, že všechny problémy byly odstraněny, přesto první výsledky s PGE jsou velice slibné.

V posledních době dochází k exponenciálnímu nárůstu počtu praktických aplikací evolučních algoritmů.

Klíčová slova: evoluční optimalizace, evoluční algoritmy, paralelní gramatická evoluce, hierarchická struktura

Key words: evolutionary optimization, evolutionary algorithms, parallel grammatical evolution, hierarchical structure

Obsah

Úvod.....	6
1. Biologické kořeny evolučních algoritmů.....	7
2. Genetické algoritmy.....	9
3. Rychlost evoluce (doba výpočtu EA).....	13
4. Paralelní genetické algoritmy.....	14
5. Paralelní gramatická evoluce.....	17
5.1 Zpracování gramatiky.....	21
5.2 Překlad genotypu do fenotypu.....	22
6. Aplikace. evolučních algoritmů.....	23
Závěr.....	23
Literatura.....	24
Odborné CV.....	27

Úvod

Evoluční algoritmy patří mezi moderní metody optimalizace systémů, a to nejen technických, ale i biologických, ekonomických atd. Optimalizační problémy se vyskytují v každém oboru lidské činnosti. Každodenně řešíme řadu problémů, jak něco udělat co nejlepším způsobem. Optimalizační problém vznikne v situaci, kdy je nutno vybrat nějakou variantu řešení. Pochopitelně se snažíme vybrat tu variantu řešení, která je pro nás nejvýhodnější. Hledáním nejlepší varianty řešení vznikne optimalizační problém, který lze obvykle řešit několika optimalizačními metodami. Abychom mohli optimalizační problém matematicky formulovat je třeba sestavit matematický model. Dále potřebujeme mít možnost porovnat různé varianty řešení a vybrat pak nejlepší variantu. Model reálného systému (reálného objektu) je vždy zjednodušen, tj. matematicky zpracovatelný model nepopisuje věrně realitu a naopak, skutečnosti blízký a složitý model nebude výpočetně realizovatelný.

Klasické genetické algoritmy (GA) používají operace selekce, křížení a mutace pro simulaci procesu reprodukce. Evoluční strategie obvykle vytvářejí potomky pouze modifikací (např. mutací) jednoho rodiče, tj. nemusí používat operaci křížení. U genetického programování mohou jednotlivé geny kódovat proměnné i funkce, zatímco geny u GA kódují pouze parametry účelové funkce. Hledanou funkci u genetického programování vyjadřujeme pomocí syntaktických stromů. Genetické programování je vhodné např. pro symbolickou regresi. Vzhledem k rozmanitosti řešených optimalizačních problémů, neexistuje obecně použitelný optimalizační algoritmus. Jedná se vždy o algoritmus, který je problémově závislý, tj. více či méně vhodný pro danou účelovou funkci. Evoluční optimalizační algoritmy nejsou vhodné pro aplikace kdy lze snadno zjistit gradienty účelové funkce, nebo účelová funkce je výpočtově velice náročná. Kombinací neevolučních optimalizačních metod (např. simulované žíhání, metoda zakázaného hledání „tabu search“, horolezecký algoritmus „hill climbing“ atd.) a evolučních optimalizací vzniknou hybridní algoritmy. Jedinec (agent) s nejlepším dosaženým ohodnocením (fitness) je považován za řešení problému.

Evoluční proces prohledávání prostoru potencionálních řešení vyžaduje najít kompromis (rovnováhu) pro dosažení dvou následujících cílů:

- a) co nejrychleji najít nejbližší (většinou lokální) řešení v malém okolí výchozího bodu,
- b) co nejrychleji najít globální řešení.

Jednotlivé metody se liší tím, který cíl preferují. Následující metody jsou seřazeny od metod směřujících k lokálnímu řešení až k metodám prohledávajících celý prostor řešení. Posloupnost metod je následující: „horolezecký“ algoritmus, tabu search, simulované žíhání, evoluční strategie, genetické algoritmy. Vzhledem k řešeným problémům bude dále pozornost věnována zejména genetickým algoritmům a gramatické evoluci.

Další vývoj a aplikace evolučních algoritmů mohou být podstatně ovlivněny pochopením některých, zatím neznámých přírodních principů evoluce, proto je v následujících kapitolách věnována velká pozornost biologickým mechanismům evoluce. Člověk zatím zdaleka není schopen využít všechny principy, které používá příroda, neboť je zatím nepochopil, nebo si neuvědomil jak je použít ke svému prospěchu. V poslední době je snaha využít evoluční principy pro návrh i technických prostředků (hardwaru). Na tuto oblast se zaměřují některé konference (např. evolving hardware systems, evolutionary hardware design, self-replicating hardware, self-repairing hardware, embryonic hardware, DNA computing a pod.).

Zvýšená pozornost je věnována problematice paralelních a hybridních genetických algoritmů.

1. Biologické kořeny evolučních algoritmů.

První evoluční teorii vytvořil Jean Baptiste de Lamarck [3-6]. Evoluci vysvětloval schopností změn živých organismů, a to postupně po mnoho generací zděděním struktury (např. silných svalů), která se stává větší a lépe vyvinutou v důsledku trvalého používání, nebo naopak zmenšenou vlivem nečinnosti. Podle dnešních poznatků většina evolučních změn výše popsaným mechanismem produkována nebyla, existuje však nepřímá možnost evoluce pomocí sekundární mutace. Lamarckova teorie byla v širokém měřítku špatně interpretována jako pouhá dědičnost

získaných znaků, nebo také zesměšňována jako změny, které se organismu vyplnily na jeho přání.

Podle Darwinovy teorie založené na přírodním výběru má vznik většího počtu jedinců, než je okolní prostředí schopné uživit, za následek vznik konkurence a "boj o přežití". Pokud ve skupině existují dědičné odchylky, potom v rámci dané populace přežijí jedinci s výhodnějšími odchylkami a budou se dál rozmnožovat na úkor jedinců s méně vhodnými vlastnostmi. Díky tomu přežijí nejsilnější jedinci, takže daný biologický druh se postupně adaptuje na okolí. Změnám životního prostředí se tak biologické druhy postupně přizpůsobí.

Chybějící článek v Darwinově teorii doplnil Gregor Mendel, který svou práci publikoval ve stejném roce, kdy vyšel Darwinův Vznik druhů. Darwinovi tato publikace přinesla okamžitě slávu, kdežto Mendelova teorie, která ještě lépe zapadá do modelu evoluce, více než čtyřicet let ležela v archivech a byla znovu objevena na přelomu našeho století. Mendelova teorie se obvykle podává ve formě dvou zákonů: zákona segregace, který popisuje rozdělení alel při vzniku gamet, a zákona volné kombinovatelnosti, který vysvětluje, že se alely jednotlivých genů rozdělují do zárodečných buněk zcela náhodným způsobem. Mendelova genetika se vztahuje k vzorům dědičnosti organismů se sexuální reprodukcí a to s více než jedním chromozómem (diploidní organismy). U diploidních organismů chromozómy existují v párech (viz paměť Ia, Ib na obr.1.1). Každý z obou chromozómů pochází od jednoho z rodičů. Nezáleží na tom, zda dominantní alela pochází od matky nebo otce. Pohlaví je určeno rozdíly v chromozómech, přičemž v přírodě existuje v tomto směru velká rozmanitost.

Termín „mem“ se poprvé objevil v roce 1976 v knize „Sobecký gen“ Richarda Dawkinse [3]. Termínem „sobecký“ je myšleno, že geny konají vše pouze pro vlastní prospěch, tj. zajímají se především o vlastní replikaci. Geny pouze chtějí, aby přešly do další generace. Geny se nacházejí v chromozómech, zatímco memy mohou být uloženy v mozku, v knihách, na stránkách Internetu atd. (viz obr.1.1) a jsou šířeny imitací. Hodně memů (kulturního dědictví) je přenášeno z rodičů na děti. Rodiče učí od malička svoje děti pravidlům vlastní společnosti (např. mateřský jazyk, morálku, náboženství atd.). Memy tak mohou být předávány horizontálně a mohou cestovat nezávisle na genech. Dawkins také zavedl důležité rozlišení mezi „replikátory“

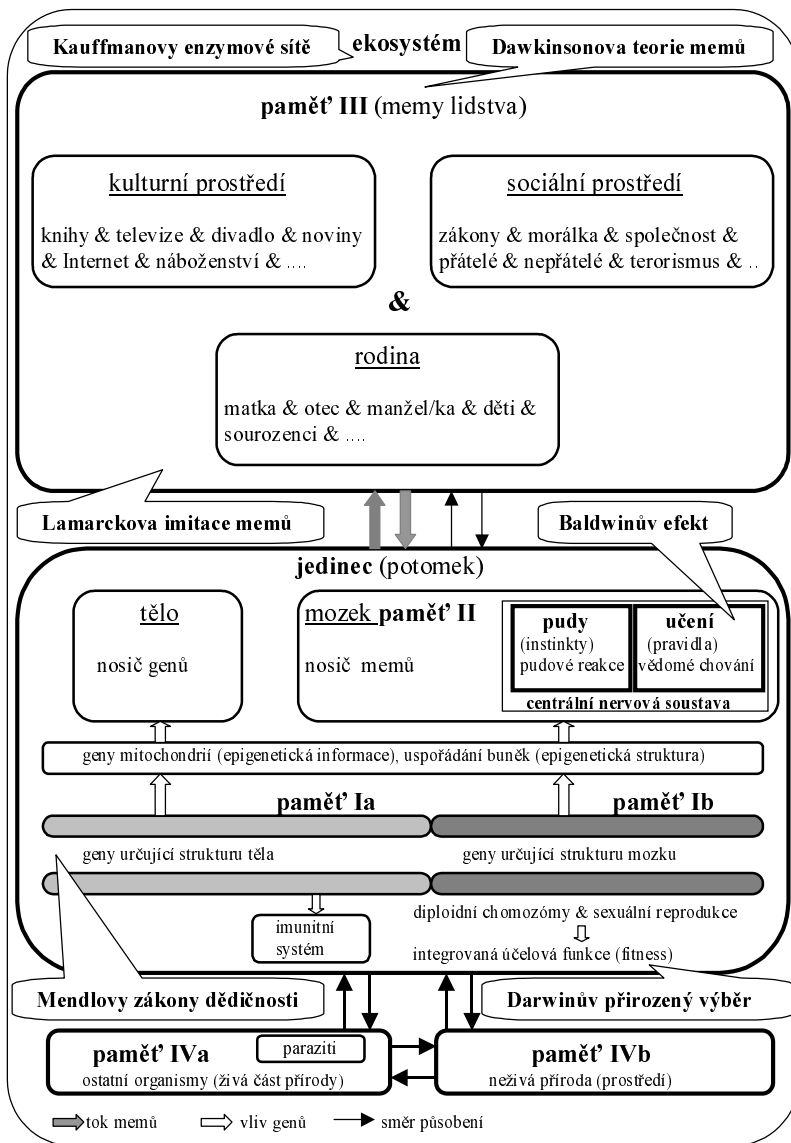
a „nosiči“, tj. geny a tělesnou schránkou. Memy mohou přeskokovat z mozku do mozku pomocí procesu imitace. Memy zahrnují celou slovní zásobu jedince, písničky, příběhy které zná, zvyky, pravidla chování.

Geny jsou instrukce pro tvorbu proteinů, které jsou uloženy v buňkách těla a jsou šířeny reprodukcí (paměť I). Jejich soutěžení řídí evoluci biologického světa. Memy jsou instrukce pro šíření chování, jsou uloženy v mozku (paměti II) a předávají se imitací. Jejich soutěžení řídí evoluci ducha (mysli). Memy mohou být uloženy v paměti lidstva (paměť III), tj. v knihách, obrazech, WWW stránkách na Internetu. Lidé mohou získat nové memy také z televize, divadla, kina, rádia, a to daleko dříve než předají své geny. V současné době existuje celá řada možností jak horizontálně přenášet memy (šířit gramotnost) pomocí klasického a mobilního telefonu, faxů, CD, kazet, počítačové sítě atd. Dawkins chápe organismus (tělo) jako „nosič“ genů, který je fyzicky transportuje a chrání. Chování může být řízeno jednoduchými pravidly uloženými v paměti II (pravá část), jako „Bud' hodný k těm, kteří tě imitují“, „Bud' hodný k dětem“, „Podporuj své nejbližší kulturní příbuzné“ atd. I současné sexuální chování je ovlivňováno kulturou a náboženstvím, tj. je řízeno i na základě memů i genů.

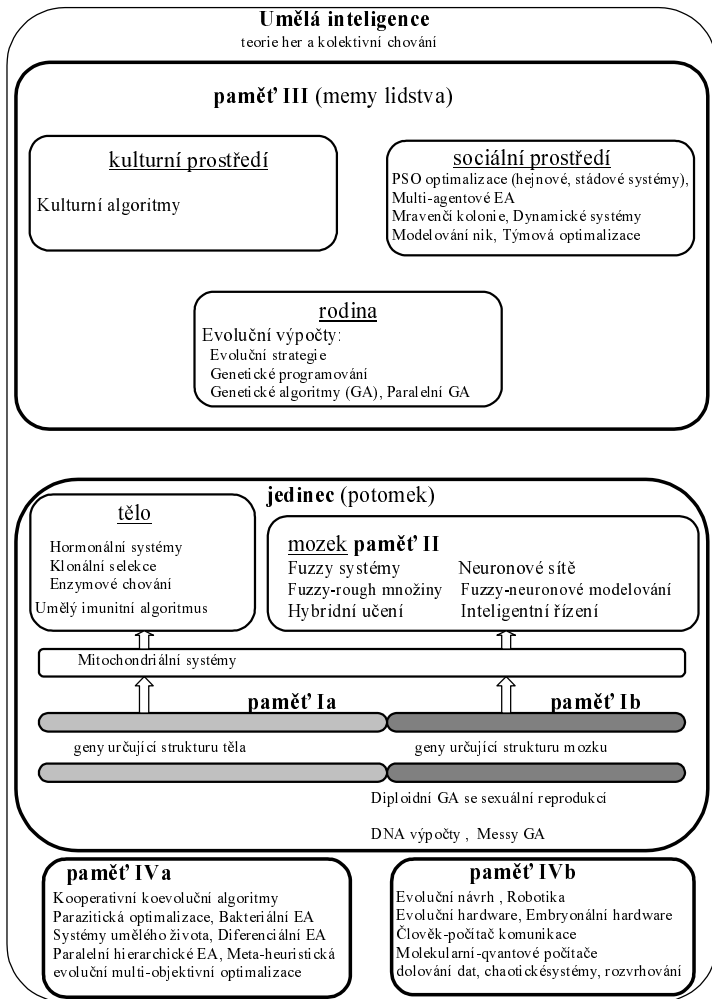
Evoluce memů probíhá v současné době mnohem rychleji než evoluce genů. Z obr. 1.1 je též patrné místo pro teorie jednotlivých autorů. Celkovou evoluci tak nelze vysvětlit pouze jedním principem, ale na základě jejich spojení. Jednotlivé teorie mohou být různě kombinovány a použity pro optimalizaci systémů. Např. evoluce memů se používá v kulturních algoritmech, Baldwinův efekt u systémů s učením.

2. Genetické algoritmy

Myšlenka systémů simulujících přírodní evoluci je stará jako počítače samy. Již geniální americký matematik Friedman v roce 1959 uvažoval o tom, že simulace principů mutace a selekce by mohla být schopna zkonstruovat „thinking machines“. Uvažoval, že na tomto principu by mohl pracovat program hrající hru šachy. V 60. letech prezentoval L. Fogel serii článků obsahujících studie, kde používal metodu evolučního programování pro vývoj systémů inteligence.



Obr. 1.1 Komplexní struktura evoluce genů a memů



Obr.1.2 Transformace struktury na obr. 1.1 do struktury umělé inteligence.

Genetické algoritmy prezentoval poprvé J. Holland jako účinný prohledávací mechanismus pro adaptivní systémy umělé inteligence. Definoval operátor křížení (crossover operátor) a operátor inverze.

Operátor křížení je považován za hlavní rozlišovací znak genetických algoritmů, které tento rekombinační operátor považují za primární. Ten pravý rozruch, který vzbudil značný zájem o GA, způsobilo první vydání knihy Hollandova studenta Davida Goldberga *Genetic Algorithms in Search, Optimization, and Machine Learning* [1]. Tato kniha, ve které je detailně popsán standardní GA, včetně příkladů implementace, je považována za „bibli“ genetických algoritmů a odkaz na ni najdeme téměř ve všech dřívějších publikacích. Po jejím vydání v roce 1989 prudce narostl zájem o tuto oblast vědy zvané *evolutionary computing*, která je nyní součástí *soft computing* (rozšířené o fuzzy logiku, neuronové sítě a fraktály). Po roce 1989 nastává exponenciální nárůst publikací, které jsou věnovány problémům řešitelným evoluční optimalizací [7].

Abychom mohli popsat činnost GA musíme si uvědomit spojení GA s řešeným problémem. To představuje dva mechanismy:

- a) způsob přiřazení řešeného problému (zakódování pomocí genů) k chromozómům,
- b) ocenění každého chromozómu, které udává cenu chromozómu vzhledem k řešenému problému (*fitness*), což umožňuje provést výběr chromozómu pro další populaci.

Slovní popis GA podle [1] je následující:

- krok 1. Vytvoř náhodně počáteční populaci N jedinců (chromozómů),
- krok 2. Oceň každý chromozóm dané populace (stanovení kvality jedince pomocí účelové funkce),
- krok 3. Vytvoř nové chromozómy procesem reprodukce; tj. aplikuj operátor selekce, mutace a křížení (rekombinace).
- krok 4. Vymaž staré jedince v staré populaci, aby se vytvořil prostor v paměti pro novou populaci.
- krok 5. Oceň nové chromozómy a vlož je do populace.
- krok 6. Jestliže je splněna ukončující podmínka, tak vyber nejlepší chromozóm jako řešení problému, jinak běž na krok 3.

Genetický algoritmus začíná inicializací populace. Obvykle bývá všech N chromozómů inicializováno náhodně jako bitový řetězec. Slovem „náhodně“ je vytvoření počáteční populace pomocí pseudonáhodného generátoru náhodných čísel. V některých případech je však vhodné zvolit jiný druh inicializace, neboť zcela náhodné počáteční nastavení může způsobit prodloužení výpočtu, neboť mohou být generovány nefunkční chromozómy, které nemají

smysl pro řešený problém. Jako typický příklad lze uvést problém obchodního cestujícího (TSP), kde je vhodné zabránit duplicitě stejných genů v chromozómu. V případě, že tak neučiníme, je třeba do fitness funkce zakomponovat postih za tento prohřešek (penalty function). Musíme ošetřovat duplicitu nejen v inicializaci, ale i v operátorech křížení a mutace. Nejběžnější používané operátory jsou následující: selekce, křížení a mutace [11].

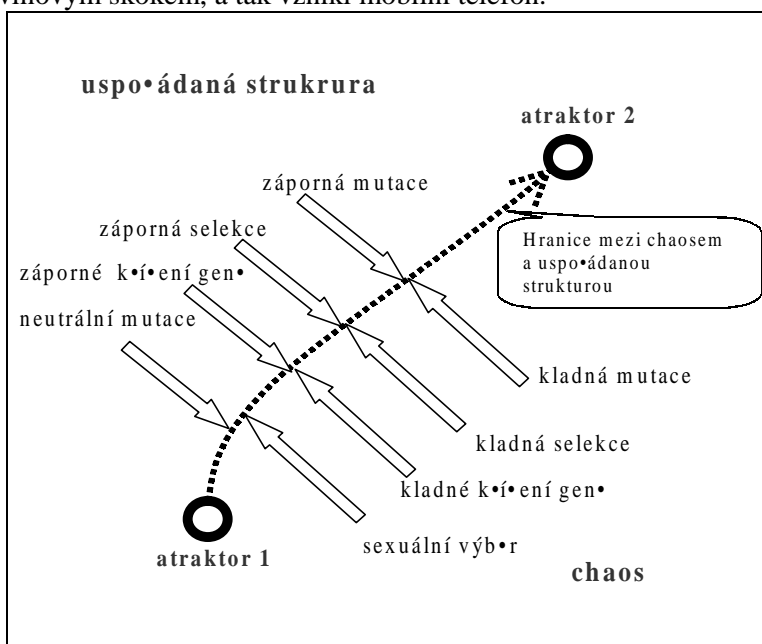
3. Rychlost evoluce (doba výpočtu EA)

Nejrychleji probíhá proces evoluce nachází-li se na hranici mezi chaosem a uspořádanou oblastí (on the edge of chaos, viz obr.3:1) [12]. Na tuto hranici evoluční proces z oblasti chaosu tlačí kladná selekce (Darwinovský přírodní výběr), kladná mutace (zlepšující úspěšnost organismu), křížení genů s kladným účinkem, sexuální výběr. Naopak z uspořádané a těžko proměnné struktury na hranici s chaosem tlačí záporná mutace (mutace se zhoršující se kvalitou organismu), záporná selekce (např. u imunitního systému), křížení genů s negativním výsledkem a neutrální mutace se zpožděným účinkem. Doba výpočtu EA je závislá na rychlosti evoluce.

U nelineárních dynamických systémů se malá změna počátečních podmínek projeví odlišným chováním, takže predikce chování těchto systémů je dlouhodobě nemožná. Složitě nelineární systémy mají svoji historii, ale z daného aktuálního okamžiku nelze udělat ani dlouhodobou předpověď chování systému ani vypočítat stav systému v minulosti.

U složitých systému se často jedná o „sít“ s velkým počtem „agentů“ pracujících paralelně [2]. Např. v mozku agentem je nervová buňka. Každý agent je součástí prostředí na které působí, a které jej také ovlivňuje. Jedná se o interakci s ostatními agenty, tj. neustálé působení a reagování na to, co ostatní agenti dělají. Dojde často ke vzniku emergentních vlastností (anomálií), interakcí mnoha částí, které skupina agentů může vytvořit jen kolektivně, něco co jednotlivý agent dělat nemůže. Např. imunitní systém je řízen lokální interakcí mezi buňkami a antigeny (protilátkami). Imunitní systém nemá centrální řízení. Podobné chování lze vysledovat v rozvoji Internetu. Paralelní evoluce agentů může možná vysvětlit vznik oka. Darwinova teorie postupné evoluce není nesprávná, je pouze neúplná. Evoluce pracuje paralelním způsobem. Evoluce probíhá

postupnou změnou v paralelně se vyvíjejících podsystemech, které v důsledku vzájemného ovlivňování mohou přejít v důsledku vzájemné rezonance (v důsledku kladných zpětných vazeb) lavinovým skokem z jednoho stabilního stavu do druhého. Většinou každá část paralelní evoluce má Darwinovský postupný vývoj, ale v důsledku kladných zpětných vazeb se komplexní struktura může změnit lavinovitým skokem. Proto paleontologové nemohou najít některé chybějící články pozvolné Darwinovské evoluce, tj. chybějící články řetězce ve vývoji druhů. Vývoj mobilního telefonu proběhl podobným způsobem. Na začátku to byly paralelní nezávislé vývoje telefonu a bezdrátového přenosu, které se v určitém okamžiku spojily lavinovitým skokem, a tak vznikl mobilní telefon.



Obr.3.1 Proces evoluce na hraně mezi chaosem a uspořádanou strukturou

4. Paralelní genetické algoritmy

Paralelní genetické algoritmy (PGA - Parallel Genetic Algorithms) jsou výkonné stochastické prohledávací strategie inspirované přírodou, které dovolují řešit větší a složitější problémy, přičemž

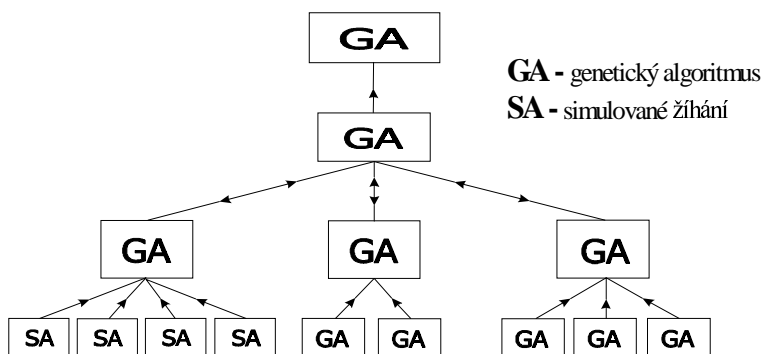
často vedou rychleji k řešení. Používají se tři modely PGA: „farmářský“ model (farming model), migrační model a difusní model. Někdy se pro poslední dva typy PGA používá název distribuované genetické algoritmy (DGA). PGA pracuje s nezávislými podmnožinami populace v nichž probíhá evoluce částečně izolovaně (dále jen podpopulace). Pojem paralelní nebo sekvenční se vztahuje k populačním strukturám, nikoli k hardwaru, na kterém jsou genetické algoritmy implementovány. Při simulaci PGA mohou být procesy nad podpopulacemi prováděny současně v procesorech multiprocesorového zařízení nebo multiplexně v programových blocích počítače s jedním procesorem. V následujícím pojednání se pod pojmem procesor rozumí jednotka provádějící proces nad populací či její podmnožinou bez ohledu na použitý způsob.

Migrací rozumíme smísení určité populace s jinou populací. Jejím následkem je vnášení (import, tok) cizích alel do genového fondu dané populace. Tím se v něm mění frekvence alel. Migrace může být jednosměrná nebo vzájemná (u sousedících populací), jednorázová, periodická nebo trvalá. Kombinovaným účinkem migrace, selekce a genového posunu může v populaci nastat genetická rovnováha se stálým rozdělením genotypových četností.

Jednotlivé migrační modely se liší strukturou propojení jednotlivých podpopulací. Struktura migračního modelu může být s centralizovaným, hierarchickým (stromovým), kruhovým, kruhovým centralizovaným, maticovým (mřížkovým) a toroidním uspořádáním. Velice dobré výsledky se dosahují při toroidním uspořádání, kdy každá podpopulace má stejný počet sousedů, např. čtyři. Pro toto uspořádání neexistují migrační problémy u okrajových podpopulací. Srovnání vlastností hierarchických (víceúrovňových) struktur je např. v literatuře [11]. Na základě našich výpočtů lze konstatovat, že stejný výkon výpočetní techniky (např. 15 počítačů propojených počítačovou sítí) dává lepší výsledky při použití víceúrovňové struktury. Např. při přechodu z dvouúrovňové struktury na tříúrovňovou se doba výpočtu může zkrátit asi třikrát [11].

Každý hierarchický (vícestupňový, víceúrovňový) systém je složen z elementů jednodušších (podsystemů). Tyto podsystemy mohou být ovšem tvořeny z dalších podsystemů nižšího řádu, tyto z dalších podsystemů ještě nižšího řádu atd. Jednotlivé organizační

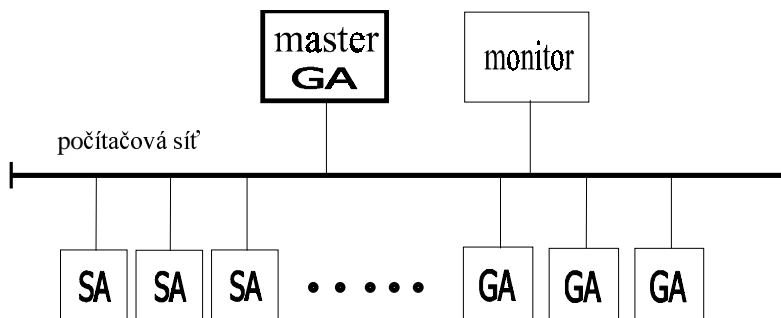
stupně nazýváme též hladiny (úrovně) systému. Vztahy uvnitř hierarchického systému můžeme rozdělit na horizontální a vertikální. Horizontální vztahy jsou vztahy mezi podsystémy ležícími na stejné organizační úrovni, vztahy vertikální pak vztahy mezi podsystémy na různých organizačních úrovních. Definování různého charakteru vztahů na různých organizačních hladinách má zásadní význam. Se zvyšující se vertikální organizovaností se objevují (emergují) nové vztahy, které na nižších organizačních hladinách neexistovaly, tzn. objevují se i kvalitativně nové vlastnosti, které nemají podsystémy. Pro různé organizační úrovně je nutné použít metody, které jsou dané hladině jevů adekvátní. Hierarchické uspořádání umožňuje uplatnění tzv. stavebnicového principu, kdy z týchž prvků můžeme tím, že je dáme do různých vztahů, vytvořit systémy kvalitativně odlišné. Význam stavebnicového principu je hlavně ekonomický. Např. buňce stačí pouze dvacet metabolických drah pro syntézu dvaceti hlavních aminokyselin k tomu, aby vytvořila prakticky neomezené množství bílkovin. Totéž platí pro nukleotidy, které kódují celou rozsáhlou genovou paměť.



Obr.4.1 Hierarchická struktura paralelního GA

Tvorba složitě organizovaných hierarchických systémů probíhá stupňovitě, postupným vytvářením relativně stabilních, stále vyšších celků (obr. 4.1, 4.2). Doba potřebná k vytvoření konečného složitého celku se zkrátí nebo je postupná výstavba dokonce podmínkou pro jeho vytvoření v reálném čase. Při postupné výstavbě je dána

možnost eliminovat chyby, které náhodně na úrovni nějakého podsystému vznikají. Pravděpodobnost, že vznikne systém o správné konečné struktuře se tak podstatně zvyšuje oproti „jednorázové“ syntéze systému.



Obr.4.2 Praktická realizace logické struktury z obr. 4.1

Analýza hierarchických systémů má mimořádný význam i pro popis regulačních obvodů. Regulační podsystém, který leží na vyšší úrovni struktury je vždy nadřazen podsystémům nižšího řádu. Nejvyšší řídicí centrum získává zpravidla omezené množství informací, které jsou pro řízení systému jako celku důležité, tj. ty, které určují především stabilitu celého systému a jeho chování navenek.

5. Paralelní gramatická evoluce

Gramatická evoluce (GE) [8-11] může být chápána jako typ genetického programování (GP) založeného na gramatice. Genetické programování navržené Kozou bylo původně programováno v jazyce LISP. Pomocí gramatické evoluce můžeme vytvořit programy v libovolném jazyku, pokud použijeme Backus-Naurovu formu (BNF). V BNF autoři Backus a Naur definovali programovací jazyk ALGOL. Gramatiky BNF se skládají z terminálů, což jsou objekty, které se mohou vyskytovat v daném jazyku, např. +,-,sin,log atd. a neterminálů. Neterminály mohou být nahrazeny jedním nebo více terminály a neterminály. Neterminální symbol je každý symbol, který může být přepsán na jiný řetězec symbolů. Naopak terminální

symbol nemůže být již přepsán. Hlavní výhoda GE oproti GP je dána možností generování víceřádkových funkcí v libovolném programovacím jazyku. Programy v GE nejsou zapsány přímo ve stromové struktuře jako je tomu u GP, ale pomocí lineárního genomu, což může být např. posloupnost celých čísel. Převod z genotypu do fenotypu se provádí pomocí operací modulo n , kde n je určeno maximálním smysluplným výběrem, např. daným počtem funkcí. Gramatikou budeme nazývat čtveřici $G = \{N, T, P, S\}$, kde

N je konečná množina neterminálních symbolů,

T je konečná množina terminálních symbolů, přičemž $N \cap T = \emptyset$,

S je počáteční symbol, $S \in N$,

P je množina přepisovacích pravidel.

Chceme-li např. identifikovat jednoduchou funkci $\cos 2x$ v rozsahu $[0, 2\pi]$ pomocí identické goniometrické funkce $1 - 2\sin^2(x)$, můžeme použít následující gramatiku [1]:

$N = \{\text{expr}, \text{op}, \text{pre_op}\}$

$T = \{\sin, \cos, \log, +, -, *, /, X, 1.0, (\,)\}$

$S = \langle \text{expr} \rangle$

a přepisovací pravidla P :

$\langle \text{expr} \rangle ::= \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle \quad (0)$

nebo $(\langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle) \quad (1)$

nebo $\langle \text{pre_op} \rangle (\langle \text{expr} \rangle) \quad (2)$

nebo $\langle \text{var} \rangle \quad (3)$

$\langle \text{op} \rangle ::= + \quad (0)$

nebo $- \quad (1)$

nebo $/ \quad (2)$

nebo $* \quad (3)$

$\langle \text{pre_op} \rangle ::= \sin \quad (0)$

nebo $\cos \quad (1)$

$\langle \text{var} \rangle ::= x \quad (0)$

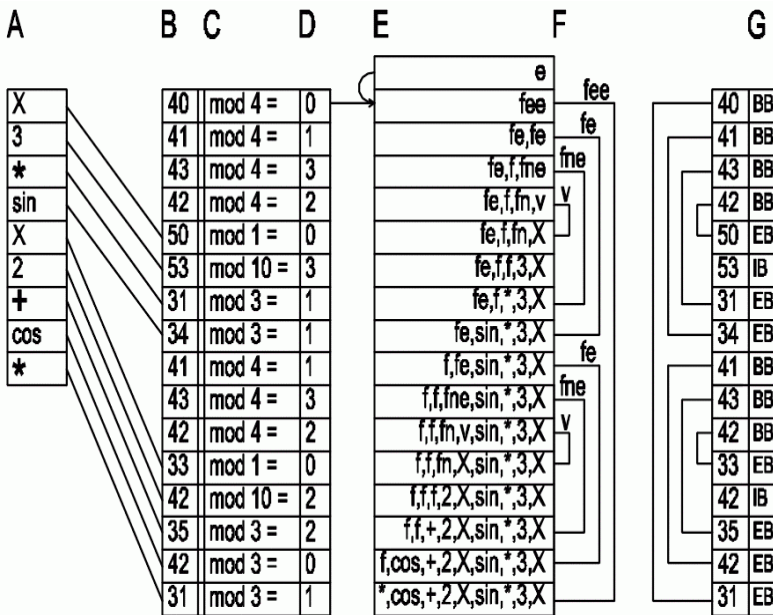
nebo $1.0 \quad (1)$

Tento způsob zápisu pravidel se mírně liší od GP. První pravidlo má čtyři možnosti výběru, druhé tři, třetí dvě a čtvrté dvě. Neterminální symboly jsou v lomených závorkách $\langle \rangle$. Začíná-li posloupnost genotypu např.:

220,240,220,203,101,53,202,.....atd.,

pak aplikací prvního pravidla, které má čtyři možnosti, na počáteční symbol $\langle \text{expr} \rangle$ provedeme tak, že číslo výběru použité možnosti dostaneme operací modulo n , kde $n = 4$, neboť výběr začíná již možností (0) a např. 220 modulo 3 = 0. Počáteční symbol S je tedy nahrazen pomocí prvního pravidla, a to variantou (0): $\langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle$. Nahrazování pak pokračuje aplikací pravidla vždy na neterminální symbol, který je nejvíce vlevo, a to tak dlouho, dokud nejsou všechny neterminální symboly nahrazeny terminálními symboly. Algoritmus končí např. výrazem:

$$1.0 - \sin(x) * \sin(x) - \sin(x) * \sin(x).$$



Obř.5.1 Vztah mezi genotypem (sloupec B) a fenotypem (sloupec A).

Paralelní gramatická evoluce PGE vychází z gramatického evolučního algoritmu, popsaného v [8]. Algoritmus je rozšířen o paralelní populační systém, který např. rozlišuje pohlaví. Každá podpopulace s rozdílným pohlavím používá jiný typ selekce a mutace. Dále obsahuje zpětnou vazbu fenotypu na genotyp.

Chromozom představuje řetězec celých nezáporných čísel, v počáteční populaci náhodně vygenerovaných. Hodnoty genů slouží k rozhodování, který prvek (terminál/neterminál) vybrat při generování těla jedince. První aplikované pravidlo má sadu čtyř možností, proto při výběru pravidla je vybraná možnost určena: hodnota_genu mod 4. Při výběru proměnné má sada proměnných pouze jeden prvek, výsledek mod 1 je vždy 0, vybere se tedy vždy první prvek (proměnná X). Gen se přečte vždy, i pokud není potřeba udělat žádné rozhodnutí (vybírání se z jednoprvkového seznamu). Toto vnáší do chromozomu jedince redundanci, kdy některé hodnoty genů mohou být libovolné.

Na obr.5.1 je schéma překladu chromozomu jedince na strukturu funkce. Tělo jedince je znázorněno jako lineární řetězec, ve skutečnosti je však uloženo v jednosměrném stromu (podřízený prvek nemá vazbu na nadřízený). Jednotlivé sloupce obrázku 5.1 mají následující význam:

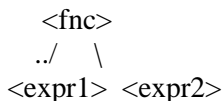
- A. Objekty hotového jedince (výstup překladu),
- B. Geny použité části chromozómu,
- C. Operace modulo: zbytek po celočíselném dělení; hodnota dělitele je daná počtem prvků v seznamu prepisovacích pravidel,
- D. Výsledek: zbytek po celočíselném dělení označuje vybraný terminál/pravidlo,
- E. Stav těla jedince po zpracování genu na daném řádku.

Protože do operace modulo vstupují dva operandy, je výsledek zpracování každého genu daný jednak hodnotou genu a současně kontextem genu. Kontext genu je počet možností mezi kterými gen rozhoduje a ten se řídí tím pro co bude gen použit. Např. gen s hodnotou 42 dává pokaždé jiný výsledek operace modulo v závislosti na tom, jaký je dělitel, tedy pro který seznam je gen použit.

5.1 Zpracování gramatiky

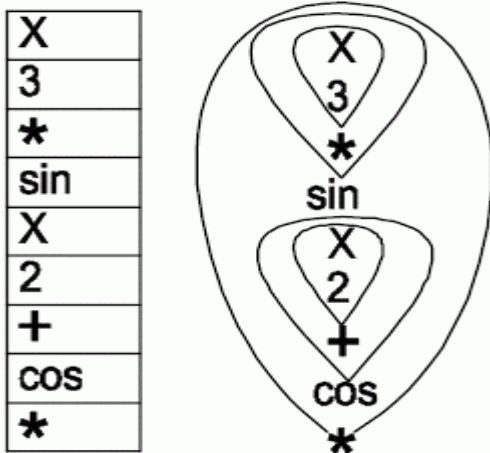
Zpracování produkčních pravidel probíhá v opačném pořadí, odpředu dozadu. Např. pravidlo `<fnc><expr1><expr2>` se zpracovává jako `<expr2><expr1><fnc>`. Zásadní rozdíl mezi neterminály `<fnc>` a `<expr>` spočívá v tom, že se symbol `<fnc>` rozbálí na právě jeden terminální symbol. Naproti tomu neterminál `<expr>` může obecně představovat libovolný počet členů funkce na různých úrovních stromu výsledného fenotypu. Je-li fenotyp

realizovaný pomocí stromu objektů, pak objekt, který vznikne překladem neterminálu <fnc> je zodpovědný za zpracování částí fenotypu, které vzniknou zpracováním <expr1> a <expr2>. Pravidlo lze tedy znázornit také jako strom



Pro výběr pravidla (výběr typu stromu) je potřeba právě jeden gen, pro jeho zpracování je potřeba n-genů a pro výběr terminálního symbolu funkce je potřeba opět jeden gen. Pozn.: při zpracování pravidla v pořadí <fnc><expr1><expr2> je kořen zpracováváný jako druhý a poslední zpracováváný výraz je jeden z listů podstromu.

Pozn: konstrukce algoritmu umožňuje velmi jednoduché přidání libovolné fce. Např. přidání funkce (fnc) tří proměnných vyžaduje pouze přidání nové funkce do množiny funkcí a přidání pravidla <expr> <expr> <expr> <fnc> nebo <num> <num> <num> <num> nebo jejich kombinací. Výběr funkcí z množiny je implementován pomocí virtuálního dvojrozměrného pole, tzn., že všechny funkce se vkládají do jednoho seznamu, ale jejich výběr probíhá tak, jakoby byly oddělené seznamy funkcí stejného počtu proměnných.



Obr.5.2 Způsob kódování stromové struktury pro zpětný výpočet

Při klasickém zpracování v pořadí $\langle \text{fnc} \rangle \langle \text{expr} \rangle \langle \text{expr} \rangle$ popsaném v [8] nelze jednoduše najít stromovou strukturu z genotypu, protože části $\langle \text{expr} \rangle \langle \text{expr} \rangle$ spotřebovávají předem neznámý počet genů a poslední zpracováváný neterminál je (z hlediska struktury) nevýznamný list podstromu. Navržený způsob kódování stromové struktury pro zpětný výpočet je zobrazen na obr.5.2.

5.2 Překlad genotypu do fenotypu

Při aplikaci tohoto obráceného systému kódování začíná pak každé zpracování podstromu fenotypu výběrem pravidla a končí kořenem tohoto podstromu (v našem případě vlastní funkce). Z pohledu genotypu je pak gen použitý pro výběr pravidla následovaný n geny s různou interpretací (použitými pro zpracování pravidla), které jsou však následovány genem určujícím terminální symbol $\langle \text{fnc} \rangle$. Tedy gen určující pravidlo je párový s genem určujícím kořen podstromu pravidla. Tyto geny lze při překladu chromozomu na tělo jedince označovat. Nejjednodušší značení může vypadat takto (viz obr.5.1 sloupec G):

BB - Begin Block

EB - Inside Block

EB - End Block

Značky EB, BB jsou párové, vymezují v chromozomu úseky, které se mohou vnořovat, nemohou se však křížit (jako závorky). Značka IB párová není, ale je vždy obsažená uvnitř nějakého páru. Nalezení příslušného EB genu k zadanému BB genu lze tedy provést pomocí jednoduchého zásobníku (LIFO).

Část chromozomu vymezená jedním párem BB-EB genů pak kóduje jistý podstrom fenotypu. Takovýto blok chromozomu je zcela samostatný a lze ho vyměnit s jiným blokem. Všechny BB geny (a pouze BB geny) určují strukturu těla jedince. EB a IB geny určují terminály, které se objeví ve výsledném fenotypu. BB geny určují výsledný strom jedince (počet podstromů, větví a listů), jsou tedy strukturální, jejich změna ovlivní výběr pravidla a tedy interpretaci ostatních genů. Změna (např. mutace) BB genu způsobí změnu ve výběru pravidla (výsledků operace modulo) mnoha ostatních genů.

Toto označování bloků zavádí zpětnou vazbu fenotypu na genotyp a přitom nesnižuje obecnost algoritmu. Nezávisí totiž na použitých

terminálech ani pravidlech, ale pouze na stromové struktuře. Díky tomuto systému je možné podstatné zvýšení výkonu algoritmu nedestruktivním křížením a mutací.

6. Aplikace evolučních algoritmů

V poslední době dochází k exponenciálnímu nárůstu publikací a praktických aplikací EA [7]. Používají pro optimalizaci ekonomických problémů, tvarů různých zařízení, dopravních a energetických problémů atd. My jsme použili EA pro optimalizaci homogenity magnetického pole NMR, struktury a parametrů neuronových sítí, neuronových regulátorů [22], fuzzy regulátorů [11] a identifikaci systémů, optimalizaci rozvrhů, trajektorie mobilních robotů a větvené mazání. V poslední době je paralelní gramatická evoluce [18-21] aplikována na automatický návrh elektrických obvodů a automatické generování programů [24].

Závěr

Cílem práce bylo podat ucelený pohled na problematiku evolučních algoritmů, zejména paralelních EA a to jak ve vztahu k přírodě, která je nositelkou inspirací, tak i k možnostem praktického využití EA. V úvodní části je provedeno všeobecné seznámení s problematikou. První kapitola je zaměřena na biologické kořeny genetických algoritmů, neboť první GA byly použity pro simulaci biologických systémů. Teprve později se začaly GA používat pro optimalizaci technických systémů. Komplexní pohled na evoluci je na obr. 1.1, který představuje vývoj obsahu čtyř pamětí a souvislosti mezi geny a memy. Podsystemy tohoto komplexního evolučního systému tvoří základ pro vznik různých optimalizačních metod na bázi evolučních algoritmů. Práce byla pojata z širšího hlediska, neboť obecná teorie systémů vychází z předpokladu, že všechny reálně existující objekty mají některé vlastnosti společné, bez ohledu na to, jaká je materiální podstata těchto objektů nebo zda jde o objekty neživé (např. stroj, technologie výrobního procesu), živé (organismus, ekosystém), či sociální (výrobní proces jako společenský jev, organizace školství). Cílem teorie systémů, do které patří i problémy jejich optimalizace, je tedy hledání formálně identických (izomorfních) zákonů, podle nichž se chovají systémy

materiálně odlišných typů. Nalezení formálních analogií mezi systémy umožňuje dále modelování jevů (simulační či matematické), což je prostředkem studia i těch procesů, které lze empirickými metodami studovat jen velice obtížně. Systémový přístup je důležitou podmínkou pro počítačové modelování složitých procesů.

Se zvyšující se organizovaností (vertikální) se u hierarchických systémů objevují (emergují) nové vztahy, které na nižších organizačních hladinách neexistovaly. Objevují se i kvalitativně nové vlastnosti oproti vlastnostem subsystémů. Myšlení jako kvalitativně nový jev vzniká až po dosažení určité organizovanosti soustavy (mozku). Na úrovni jedné buňky nebo jednoduché soustavy nervových buněk je existence myšlení nemožná. Chování agentů se jako nová kvalita projeví až na chování distribuovaného řídicího systému jako celku. Obdobně některé nové jevy se při optimalizaci pomocí GA objeví až na úrovni genofondu (např. vliv diploidních chromozómů nebo vliv migrace).

I přes dosažené pozitivní výsledky optimalizací pomocí evolučních algoritmů je jasné, že příroda používá ještě rafinovanější a přitom ne zcela odhalené principy. Většina dosud popsanych EA odpovídá pouze základním principům živé přírody. Paralelní GA našly uplatnění při návrhu spojů uvnitř složitých VLSI obvodů, včetně mikroprocesorů (topologické úlohy). Zdokonalování technických prostředků (počítačů) vede ke zdokonalování optimalizačních prostředků na bázi PGA, čímž dochází k vzájemnému podporování. Dochází tak k prudkému nárůstu výkonu jak počítačů tak i EA, přičemž se objevují nové výhodné vlastnosti, které nemohly vzniknout u jednoduchých izolovaných systémů.

Využití evolučních algoritmů je nepochybně velice perspektivní a nabízí řadu možností nejen ve zde zmíněných aplikacích. Nabízí se možnost obecného použití k řešení těch nejobtížnějších optimalizačních problémů.

Literatura

- [1] Goldberg, D.E.: Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, 1989
- [2] Kauffman, S.A.: The origin of order. Self-organization and selection in evolution, Oxford University Press, New York, Oxford 1993

- [3] Dawkins, R.: Sobecký gen. Praha: Mladá fronta 1998.
- [4] Ho, M.W.: Genetické inženýrství – naděje nebo hrozba? Praha: Alternativa 1998.
- [5] Ridley, M.: Červená královna - sexualita a vývoj lidské přirozenosti, Mladá fronta, 1999
- [6] Rosypal, S.a kol.: Přehled biologie, Státní pedagogické nakladatelství Praha 1987, 1. vydání
- [7] Alander J.T.: Bibliography: GAs in industry, Proceedings of INWGA, Vaasa, pp.3132-3141, Finland 1995
- [8] O'Neill, M., Ryan, C.: Grammatical Evolution: Evolutionary Automatic Programming in an Arbitrary Language Kluwer Academic Publishers 2003.
- [9] O'Neill, M., Brabazon, A., Adley C.: The Automatic Generation of Programs for Classification Problems with Grammatical Swarm, Proceedings of CEC 2004, Portland, Oregon (2004) 104 – 110
- [10] Piaseczny, W., Suzuki, H., Sawai, H.: Chemical Genetic Programming – Evolution of Amino Acid Rewriting Rules Used for Genotype-Phenotype Translation, Proceedings of CEC 2004, Portland, Oregon (2004) 1639 - 1646.
- [11] Li Z., Halang W. A., Chen G.: Integration of Fuzzy Logic and Chaos Theory; paragraph: Osmera P.: Evolution of Complexity, Springer, 2006 (ISBN: 3-540-26899-5) 527 – 578.
- [12] Waldrop, M.M: Complexity – The Emerging Science at Edge of Order and Chaos, Viking 1993
- [13] Ošmera, P., Roupec, J., Matoušek, R.: Genetic Algorithms with Diploid Chromosomes and Sexual *Reproduction*, in the book P. Sinčák, J. Vaščák: Quo Vadis Computational Intelligence – new Trends and Approaches in Computational Intelligence Physica – Verlag, Springer - Verlag Company (2000) 317-323.
- [14] Ošmera, P., Roupec, J., Matoušek, R.: *Genetic Algorithms with Sexual Reproduction*, Proceeding of World Multiconference on Systemics, Cybernetics and Informatics SCI 2000, Orlando, Florida, USA, July 23-26 (2000) 306-311.
- [15] Ošmera, P., Roupec, J.: *Limited Lifetime Genetic Algorithms in Comparison with Sexual Reproduction Based GAs*, Proceedings of the 6th International Conference on Soft Computing MENDEL 2000, June 7-9, Brno, Czech Republic (2000) 118-126.

- [16] Ošmera, P., Kvasnička, V., Pospíchal, J.: *Genetic Algorithms with Diploid Chromosomes*. Proceedings of MENDEL'97 (1997) 111 – 116.
- [17] Ošmera, P., Roupec, J., Matoušek, R.: *Energie, entropie a evoluce živé hmoty*, sborník konference Kognice a umělý život, Smolenice, Slovensko (2001) 203-223
- [18] Ošmera, P., Popelka, O.: The Automatic Generation of Programs with Parallel Grammatical Evolution, , 13th Zittau Fuzzy Colloquium, Zittau, Germany (2006) 332-339
- [19] Ošmera, P., Popelka, O., Pivoňka, P.: Parallel Grammatical Evolution with Backward Processing, ICARCV 2006, 9th International Conference on Control, Automation, Robotics and Vision, Singapore (2006) 1889-1894
- [20] Pivoňka, P., Veleba, V., Ošmera, P.: Adaptive Controllers using Neural Network Based Identification for Short Sampling Period, 9th International Conference on Control, Automation, Robotics and Vision, Singapore (2006) 521-526
- [21] Ošmera, P., Panáček, T., Pivoňka, P.: Parallel grammatical Evolution, WCECS2007, proceedings of World Congress on Engineering and Computer Science, San Francisco (2007) 897-902
- [22] Pivoňka, P.; Veleba, V.; Ošmera, P.: Using of Neural Network Based Identification for Short Sampling Period in Adaptive Control. In Systems Theory and Applications. Vol. 2. Řecko: WSEAS 2007, ISBN: 978-960-8457-90-4 (2007) 217-222.
- [23] Ošmera, P., Popelka, O., Pivoňka, P.: Two Level Parallel Grammatical Evolution, ICARCV 2006, CEC2008, Hong Kong (2008) 748-756
- [24] Ošmera, P., Popelka, O.: Parallel Gramatical Evolution for Circuit Optimization, Proceedings of the 8th International Conference on Evolvable Systems: From Biology to Hardware, Prague, Czech Republic (2008) v tisku

Odborné CV:

Doc. Ing. Pavel Ošmera, CSc.

Ústav automatizace a informatiky
VUT v Brně – Fakulta strojního inženýrství
E-mail: osmera@fme.vutbr.cz
Tel.: +420 541 142490

Datum narození: 22. 7. 1946

Vzdělání:

2002	Doc. pro obor Řízení strojů a procesů (habilitace)
1987	Doc. pro obor Regulační a automatizační technika.
1978	CSc. VAAZ Brno, formou interní aspirantury, disertační práce na téma „Číslicové metody určení základního tónu řeči“.
1969	Ing. VUT v Brně, fakultě elektrotechnické, obor Technická kybernetika, s vyznamenáním.
1964	Maturita - Střední průmyslová škola elektrotechnická v Brně, obor měřicí a řídicí technika, s vyznamenáním.
1960	Základní devítiletá škola Brno

Přehled vědecko-výzkumné, odborné a pedagogické činnosti

Odborná činnost po roce 1987:

- řešení státního úkolu III-4-1/0506 „Modelování dynamických systémů“, dílčí úkol „Adaptivní číslicová regulace“. Úkol ukončen obhajobou zprávy v roce 1990.
- návrh adaptivního číslicového regulátoru AR-100 pro Sigmu Brno a pomoc při jeho zavedení do výroby v roce 1989, regulátory vystaveny a předvedeny firmou Sigma na Mezinárodním strojírenském veletrhu v Brně v roce 1989
- řešení úkolu „Automatizace cínovací linky pomocí mikropočítače“ pro VD Didaktik Skalica na Slovensku, úkol dokončen předvedením funkčního vzorku v roce 1990
- v 1991 zahájena práce na řešení grantového úkolu C1/VUT, jehož cílem je postupné zavádění číslicového distribuovaného řídicího systému pro vytápění a vzduchotechniku budov strojní fakulty v Brně, aby se zvýšila spolehlivost a snížily náklady na energii. V roce 1992 byl úspěšně oponován

- v roce 1992 jsem se podílel na grantovém úkolu VUT FS č. 211 „Výpočetní metody mechatroniky
- v roce 1992 jsem se podílel na řešení grantového úkolu VUT Brno č. A5 „Mechatrické systémy“
- v roce 1992 zahájena a obhájena práce na řešení grantového úkolu č. 212 „Simulace provozu výtahů na strojní fakultě“
- od roku 1992 do 1994 zapojen do projektu TEMPUS
- od roku 1995 do roku 1997 odpovědný řešitel projektu GAČR č.102/95/0467:
- “Využití genetických algoritmů pro řízení homogenity magnetického pole v magnetu spektrometru NMR”
- od roku 1998 až dosud zapojen do projektu CEEPUS
- v roce 1993 byl řešen grantový projekt č. 1266/93 Fondu dynamického rozvoje vysokých škol: „Nasazení distribuovaného řídicího systému pro vytápění“, praktickým výsledkem je zkušební provoz řídicího systému vzduchotechniky poslucháren P3 a P6 strojní fakulty
- od roku 1995 organizování mezinárodních konferencí MENDEL 1995-2007
- 1995-1996 spoluřešitel projektu Fondu rozvoje vysokých škol: Příspěvek k rozvoji matematických metod projektového řízení
- 1995-1996 spoluřešitel grantů VUT:
 - hlavní řešitel Doc.Jan Mičín: Katalog charakteristických dešťů sestavený pomocí genetických algoritmů
 - hlavní řešitel Doc.Josef Koláčný: Optimalizace parametrů reluktančního motoru
- od roku 1995-1997 řešitelem grantu GAČR: Využití genetických algoritmů pro řízení homogenity magnetického pole v magnetu spektrometru NMR
- člen programového výboru 11 mezinárodních konferencí GALESIA 95, GALESIA 97, DAAAM 95, DAAAM 96, DAAAM 97, 1-st Slovak Neural Network Symposium 96
- CEC 00, CEC 01, 7th až 13th Zittau Fuzzy Colloquim, MENDEL 95, MENDEL 96, MENDEL 97, MENDEL98, MENDEL 99, MENDEL 2000, MENDEL 2001, MENDEL 2002, MENDEL 2003, MENDEL 2004-2007
- Od roku 1998 do roku 1999 návrh a realizace systému tísňového volání pro penziony a pečovatelské služby. Realizováno v penzionu v Boskovicích v roce 1999.
- Od 1999 spoluřešitel výzkumného záměru Automatizace technologií a výrobních procesů ev. č.CEZ J22/98: 260000013. Následoval výzkumný záměr do konce roku 2004 s ev. č. MSM 260000013.

- Projekt FRVŠ v roce 2001 č.:IS10128/2001, spoluřešitel: Vývojové centrum informačních technologií signálových procesorů
- Grant GAČR č.103/05/0400, Rehabilitace kanalizačních staveb na poddolovaném území, 2005-2007
- Grant GAČR č.102/06/1132, Softcomputingové metody v řízení, 2006-2008
- Výzkumný záměr MSM0021630529: Inteligentní systémy v automatizaci, 2007-2009

Kvalifikační a odborné zaměření:

Mikroprocesorová technika, zařízení výpočetní techniky, neuronové sítě, teorie informačních systémů, evoluční optimalizace. Metodika technického vysokoškolského vzdělávání, návrh a inovace studijních programů z oblastí automatického řízení a inženýrské informatiky.

Pedagogické působení a schopnosti:

- odborná pedagogická praxe 36 let, z toho vedení přednášek, konzultací ve 14 předmětech (1971 - 2007 ve všech formách studia (bakalářské, inženýrské, doktorské), včetně vedení diplomových prací (72 diplomantů), výuky včetně výuky v doktorském studiu a v anglickém jazyce.
- autor a spoluautor 13 titulů skript
- Zavedení a přednášení předmětů postupně se vyvíjejících do současné podoby: Informační systémy (3. ročník bakalářského studia), Počítačový hardware (1. ročník bakalářského studia), Neuronové sítě a fuzzy řízení (5. ročník navazujícího magisterského studia), Informační systémy Evoluční optimalizace (doktorské studium, výuka také v angličtině).
- Vytvoření řady laboratorních úloh k předmětům Počítačový hardware a Informační systémy.
- člen **3** komisí pro státní závěrečné zkoušky (VUT FSI- 19 let, VŠB Ostrava 2 roky, EPI Kunovice od 2006)
- Předseda Rady studijního oboru Aplikovaná informatika a řízení 1999-2002
- školitel pro doktorské studium - celkem 10 posluchačů (3 již ukončili úspěšně studium: Ing.Chládek, Ing.Matoušek, Ing.Roupec), 3 po dokt. zkoušce (Ing. Weisser, Ing. Pernes, Ing. Kostíha)
- člen Oborové rady doktorského studia boru "Technická kybernetika" na VUT-FSI - od r. 1991 do roku 2004.

- sestavení výukových plánů, zavedení výuky a vedení přednášek 7 nových předmětů na VUT FS v letech 1991 až 2008
- sestavení výukových plánů, zavedení výuky a vedení přednášek 2 nových předmětů na Evropském polytechnickém institutu 2005-2008, s.r.o., Osvobození 699, 686 04, Kunovice
- vybudování a samostatná inovace 1 laboratoře (Laboratoř mikroprocesorové a výpočetní techniky), podíl na zavedení a inovaci dalších 3 laboratoří a počítačových učeben.
- předseda zkušební komise pro státní závěrečné zkoušky, Univerzita Tomáše Bati ve Zlíně, Fakulta aplikované informatiky, od 2005 dosud
- člen vědecké rady na Evropském polytechnickém institutu, s.r.o., Osvobození 699, 686 04, Kunovice, od 2005 dosud
- oponování disertačních prací ze zahraničí (Jouni Lampinen – Finsko 1998, Juhong Liu – Finsko 2006, Nahla Ahmed Mohamed El-Haggar – Egypt, 2004)

Publikační činnost: knihy, učebnice a skripta (**16**), interní učební texty a elektronické opory (**4**), články v odborných časopisech (**7**), patenty (**10**), užité vzory (**7**), zlepšovací návrhy (**27**), příspěvky ve sbornících a příspěvky na konferencích (**162**), populárně-odborné příspěvky v časopisech (**2**), oponované vědecko-výzkumné zprávy a projekty (**11**), ostatní vědeckovýzkumné a technicko-vývojové zprávy (**14**), odborné posudky (**43**).

Uznání odbornou veřejností doložené jako **citace** publikací (**10** v zahraničí, **36** v tuzemsku).

Jazykové znalosti: Angličtina (dobře), ruština (uspokojivě), němčina (pasivně).

Ocenění:

1982–nejlepší zlepšvatel na VAAZ Brno.

Zahraníční studijní pobyty a tématické návštěvy:

- TU Magdeburg (1986,1988,1989) - celkem 3 týdny
- Loughborough University of Technology - Velká Británie (1993) - celkem 3 měsíce
- University of Hamilton, University of Toronto - Kanada (1997) - celkem 3 týdny
- TU Plovdiv, TU Gabrovo, Bulharsko (1998, 1999, 2000) - celkem 6 týdnů
- Shizuoko University of Japan, Japonsko (1999) - celkem 3 týdny