

Tomasz Dominik Gwiazda

ALGORYTMY GENETYCZNE

kompendium

TOM I

Operator krzyżowania
dla problemów numerycznych



TOMASZ DOMINIK GWIAZDA

**ALGORYTMY
GENETYCZNE
KOMPENDIUM
TOM 1
OPERATOR KRZYŻOWANIA
DLA PROBLEMÓW NUMERYCZNYCH**



[Kup książkę](#)

Projekt okładki: **Michał Rosiński**
Redakcja: **Krystyna Knap**
Skład komputerowy: **Tomasz Dominik Gwiazda**

Zastrzeżonych nazw firm i produktów użyto w książce wyłącznie w celu identyfikacji.

Copyright © by Wydawnictwo Naukowe PWN SA
Warszawa 2007

ISBN 978-83-01-15168-3

Wydawnictwo Naukowe PWN SA
00-251 Warszawa, ul. Miodowa 10
tel. (0 22) 69 54 321
faks (0 22) 69 54 031
e-mail: pwn@pwn.com.pl
www.pwn.pl

Wydawnictwo Naukowe PWN SA
Wydanie pierwsze
Arkuszy drukarskich 24
Druk ukończono w maju 2007 r.
Druk i oprawa ZWP HEL,
04-007 Warszawa,
ul. Grenadierów 77

[Kup książkę](#)

Spis treści

1. Wstęp.....	9
2. Krótko o algorytmach genetycznych	13
Historia	13
Kanoniczna postać AG	14
Generowanie populacji początkowej	15
Ocena populacji	18
Selekcja rodziców	21
Operatory genetyczne	23
Kreowanie nowej populacji	27
Pełna postać kanonicznego AG.....	31
Literatura.....	39
3. Standardowe operatory krzyżowania.....	41
Krzyżowanie jednopunktowe (1-Point Crossover)	41
Krzyżowanie wielopunktowe (k-Point Crossover)	41
Krzyżowanie tasujące (Shuffle Crossover).....	43
Krzyżowanie zastępujące (Reduced Surrogate Crossover).....	44
Krzyżowanie równomierne (Uniform Crossover).....	45
Krzyżowanie niszczące (Heuristic Uniform Crossover/Highly Disruptive Crossover)	46
Krzyżowanie uśredniające (Average Crossover)	47
Krzyżowanie ziarniste (Discrete Crossover).....	48
Krzyżowanie płaskie (Flat Crossover).....	48
Krzyżowanie heurystyczne-1 (Heuristic Crossover /Intermediate Crossover)	49
Krzyżowanie mieszające (Blend Crossover)	50
4. Operatory krzyżowania dla problemów kodowanych liczbami binarnymi	53
Krzyżowanie powielające podobieństwa (Random Respectful Crossover).....	53
Krzyżowanie oparte na dominacji (Masked Crossover)	55
Krzyżowanie wyboru operatora (1bit Adaptation Crossover).....	57
Krzyżowanie wielowymiarowe (Multivariate Crossover)	61
Krzyżowanie homologiczne (Homologous Crossover)	63
Krzyżowanie zliczające-1 (Count-preserving Crossover-1)	65
Krzyżowanie elitarne (Elitist Crossover)	67
Krzyżowanie skanujące (Scanning Crossover)	68
Krzyżowanie częściowe (Partial Copy Crossover)	71
Krzyżowanie nierównomierne oparte na wiedzy (Knowledge-Based Nonuniform Crossover).....	72
Krzyżowanie średnicą (Circle-ring Crossover).....	74
Wystarczająca wymiana (Sufficient Exchanging)	75
Ewolucja powiązań (Linkage Evolving Genetic Operator)	77
Krzyżowanie łańcuchami-1 (2N-parent Parameter Wise Crossover)	79

Krzyżowanie diagonalne (Diagonal Crossover)	81
Krzyżowanie pulą genów-1 (Gene Pool Crossover-1).....	83
Krzyżowanie hierarchiczne (Hierarchical Crossover)	84
Krzyżowanie iloczynem/sumą logiczną (Randomized and/or Crossover).....	88
Krzyżowanie jednokierunkowe (Simple Conjugation Operator)	89
Strategia wyboru operatora-1 (Adaptive Strategies of Mixing Crossovers)	91
Krzyżowanie ortogonalne-1 (Orthogonal Crossover)	93
Krzyżowanie mikrobiologiczne (Microbial Crossover).....	95
Krzyżowanie selektywne-1 (Selective Crossover-1)	98
Krzyżowanie powiązań (Exchange Crossover/ Linkage Crossover)	100
Krzyżowanie skanujące wielopłciowe (Multi Sexual Scanning Crossover).....	101
Krzyżowanie różnicami (Differences-Based Crossover)	103
Krzyżowanie kumulujące (Fusion Crossover)	105
Krzyżowanie wielochromosomowe (Multiple Chromosomes Crossover).....	106
Krzyżowanie ograniczone (Restricted Crossover)	108
Krzyżowanie selektywne-2 (Selective Crossover-2)	110
Samokrzyżowanie (Self Crossover)	112
Krzyżowanie tnące (Multi-cut Crossover)	115
Transpozycja (Transposition Operator)	116
Transpozycja z selekcją turniejową (Tournament Based Transposition Operator)....	120
Dominujący spłot (Dominant Splice/Symbiotic Combination)	126
Krzyżowanie zdysocjowane (Dissociated Crossover)	127
Krzyżowanie spontaniczne (Spontaneous Crossover)	129
Krzyżowanie metodą bisekcji (Binary Search Point Crossover)	131
Krzyżowanie stabilne (Fixed Crossover)	134
Powielanie z lokalnym dostrajaniem (Common Features/Random Sample Climbing Crossover).....	135
Transpozycja bezpłciowa (Asexual Transposition)	137
Krzyżowanie różnic (Disrespectful Crossover)	141
Krzyżowanie dwupunktowe asymetryczne-1 (Asymmetric Two-point Crossover) ...	143
Krzyżowanie dwupunktowe asymetryczne-2 (Variation of Asymmetric Two-point Crossover).....	145
Najlepsza kombinacja (Best Combinatorial Crossover)	147
Związki mieszane (Hybridization Crossover).....	148
Strategia wyboru operatora-2 (Mixed Crossover).....	149
Krzyżowanie łańcuchami-2 (Direct Design Variable Exchange Crossover)	151
Poszukiwanie schematów (Best Schema Crossover).....	153
Krzyżowanie chromosomów o zmiennej długości (Variable Length Genomes Crossover).....	155
Krzyżowanie trójosobnicze (Three-Parent Crossover)	159
Krzyżowanie częściowo mieszające (Partially Randomized Crossover)	161
Krzyżowanie nierównomierne oparte na statystykach (Statistic-Based Adaptive Non-Uniform Crossover)	162
Krzyżowanie zbiorem operatorów-1 (Multiple Crossover Operators-1)	164
Krzyżowanie z decydem (Half Sibling and a Clone)	167
Adaptacja liczby punktów krzyżowania i mutacji (Adaptive Number of Crossover Points).....	168
Kooperacja/rywalizacja płci (Sexual Selection Crossover)	170

Tasowanie chromosomów (Chromosome Shuffling)	172
Krzyżowanie ortogonalne-2 (Orthogonal Latin Mutli-Parent Crossover)	174
Krzyżowanie z punktową mutacją (Different Location Crossover).....	176
Strategia wyboru operatora-3 (Combined Balanced Crossover).....	177
Krzyżowanie wielokrotne-1 (Hybrid 1-Point Crossover)	180
Krzyżowanie zliczające-2 (Count-preserving Crossover-2)	182
Krzyżowanie ważone przystosowaniem-1 (Fitness Weighted Crossover (Binary)) ...	185
Adaptacja prawdopodobieństwa-1 (Adaptive Probability Crossover-1).....	187
Maksymalizacja podobieństw (Schema-Based Crossover).....	189

5. Operatory krzyżowania dla problemów kodowanych liczbami rzeczywistymi..... 193

Krzyżowanie linearne-1 (Linear Crossover)	193
Krzyżowanie heurystyczne-2 (Heuristic Crossover2).....	194
Krzyżowanie proste (Simple Crossover)	196
Krzyżowanie pojedyncze arytmetyczne (Single Arithmetical Crossover).....	197
Krzyżowanie arytmetyczne (Arithmetical Crossover, Intermediate Crossover2, Linear Crossover, Guaranteed Average Crossover, Convex Crossover)	199
Krzyżowanie wstawiające (Injection Crossover).....	201
Adaptacja prawdopodobieństwa-2 (Adaptive Probability Crossover-2).....	203
Krzyżowanie linearne-2 (Linear BGA Crossover).....	205
Krzyżowanie z simpleksem-1 (Simplex Crossover)	207
Krzyżowanie linearne-3 (Simulated Binary Crossover)	209
Krzyżowanie ze zbiorami rozmytymi-1 (Fuzzy Crossover)	211
Krzyżowanie ze zbiorami rozmytymi-2 (Fuzzy Connectives Based Crossover)	213
Krzyżowanie z wspinaczką-1 (Crossover Hillclimbing).....	215
Krzyżowanie przypadkowe (Random Crossover).....	217
Krzyżowanie łańcuchami-3 (Variable Length Segments Crossover)	218
Krzyżowanie uśredniające zaburzone (Unfair Average Crossover)	221
Krzyżowanie równomierne ciągłe (Continuous Uniform Crossover).....	223
Krzyżowanie geometryczne (Geometrical Crossover).....	225
Krzyżowanie sferyczne (Sphere Crossover)	227
Krzyżowanie z wektorem różnic (Differential Evolution Crossover).....	229
Krzyżowanie z simpleksem-2 (Simplex Crossover-2).....	231
Krzyżowanie wielokrotne-2 (Multiple Crossover Per Couple).....	233
Krzyżowanie pulą genów-2 (Gene-Pooling Crossover-2)	235
Krzyżowanie środkiem masy-1 (Center of Mass Crossover).....	237
Krzyżowanie wieloosobnicze-1 (Multi-parent Feature-wise Crossover).....	238
Krzyżowanie z przystosowaniem częściowo separowalnym (Partially Separable Crossover).....	240
Krzyżowanie drażące (Seed Crossover).....	242
Niewypukła wieloosobnicza kombinacja liniowa (Non-convex Linear Combination of Multiple Parents)	243
Krzyżowanie kierunkowe (Direction-Based Crossover).....	244
Krzyżowanie z pochodną (Derivative-Based Crossover)	246
Krzyżowanie z rankingiem (Rank Based Crossover).....	247
Krzyżowanie paraboliczne-1 (Parabolic Crossover).....	249
Krzyżowanie kierowane (Guided Crossover)	251
Krzyżowanie wielokrotne-3 (Multiple Crossover on Multiple Parents)	253

Krzyżowanie z simpleksem-3 (Simplex Crossover-3).....	255
Krzyżowanie jednorodne (Uniform Design Crossover).....	257
Krzyżowanie środkiem grawitacji (Center of Gravity Crossover).....	258
Krzyżowanie jednopunktowe uśrednione (1-point Average Crossover).....	260
Krzyżowanie dwupunktowe uśrednione (Messy Average Crossover).....	261
Adaptacja prawdopodobieństwa-3 (Adaptive Probability Crossover-3).....	263
Krzyżowanie z wspinaczką-2 (Site-Specific Crossover)	264
Krzyżowanie z przystosowaniem nieseparowalnym (BLX Principal Component Analysis and Independent Component Analysis)	269
Krzyżowanie ortogonalne z kwantyzacją (Orthogonal Crossover with Quantization).....	271
Krzyżowanie równomierne gaussowskie (Gaussian Uniform Crossover).....	272
Krzyżowanie pulą genów-3 (Set-Oriented Crossover)	274
Adaptacja prawdopodobieństwa-4 (Adaptive Probability Crossover-4).....	275
Adaptacja prawdopodobieństwa-5 (Adaptive Probability Crossover-5).....	278
Krzyżowanie pozycyjne (Position Crossover).....	281
Krzyżowanie najlepszy-najlepszy (Best-Best Crossover).....	283
Adapttywne krzyżowanie w losowej liczbie punktów (Adaptive Q-rand Point Crossover).....	285
Krzyżowanie w losowej liczbie punktów (Q-rand Point Crossover)	286
Krzyżowanie wielokrotne-4 (Multiple Crossover)	289
Krzyżowanie arytmetyczne poprawione-1 (Improved Arithmetical Crossover).....	291
Translokacja wzajemna (Reciprocal Translocation Operator).....	293
Krzyżowanie arytmetyczne poprawione-2 (Hybrid Arithmetical Crossover).....	298
Krzyżowanie wielokrotne-5 (Generation of Multiple Descendants and Selection of the Two Best).....	300
Krzyżowanie w otoczeniu-1 (Adaptive Neighborhood-based Multi-parent Crossover).....	302
Krzyżowanie ułomne (Imperfect Crossover).....	304
Adaptacja prawdopodobieństwa-6 (Adaptive Probability Crossover-6).....	307
Krzyżowanie zbiorem operatorów-2 (Multiple Crossover Operators-2)	309
Krzyżowanie samoadaptacyjne (Heuristic-Based Self -Adapting Crossover)	311
Krzyżowanie ze zmiennymi typami danych (Mixed Variable Crossover).....	315
Krzyżowanie cylindryczne (Curved Cylinder Crossover)	319
Krzyżowanie wymienne (Exchanging Information Crossover).....	322
Krzyżowanie arytmetyczne poprawione-3 (Real-Biased Crossover).....	323
Krzyżowanie Taguchiego-1 (Taguchi Crossover/Main Effect Orthogonal Crossover).....	325
Krzyżowanie ortogonalne z interakcją (Interaction Effect Orthogonal Crossover)	328
Krzyżowanie różnorodne (Diverse Crossover).....	331
Krzyżowanie w otoczeniu-2 (Parent Centric BLX- α Crossover)	333
Krzyżowanie Taguchiego-2 (Taguchi Crossover-2)	335
Krzyżowanie w grupie (Extended GA Crossover).....	337
Krzyżowanie jednogenowe (Single Gene Crossover).....	339
Krzyżowanie ważone przystosowaniem-2 (Fitness Weighted Crossover (Real)).....	341
Krzyżowanie zależności (Inheritance Crossover).....	343
Adaptacja prawdopodobieństwa dla genu (Adaptive Probability of Gene Crossover).....	345
Adaptacja prawdopodobieństwa-7 (Adaptive Probability Crossover-7).....	346
Krzyżowanie ze znormalizowanym dystansem przystosowania (Normalized Fitness Crossover).....	348

Krzyżowanie z kontrolą zawartości (Controlled Content Crossover)	350
Krzyżowanie środkiem masy-2 (Center of Mass Crossover-2)	353
Krzyżowanie różnicujące (Parent Differentiation Crossover)	355
Metoda przełączająca (Improved Crossover and Mutation)	356
Adaptacja prawdopodobieństwa-8 (Adaptive Probability Crossover-8).....	358
Krzyżowanie królewskie (King Strategy Crossover).....	361
Krzyżowanie pulą genów-4 (Gene-Pooling Crossover-4)	362
Krzyżowanie sterowane przystosowaniem (Fitness Guided Crossover).....	364
Krzyżowanie w otoczeniu-3 (Continuous Adaptive Culture Model Crossover).....	365
Krzyżowanie paraboliczne-2 (Fitness-Based Parabolic Crossover).....	367
Krzyżowanie wieloosobnicze-2 (Hybrid Panmictic (multi-parent) Crossover)	368
6. Operatory statystyczne	371
UNDX – Unimodal Normal Distribution Crossover.....	371
MM-B – Factorized Distribution Method.....	372
PCX – Parent-Centric Crossover	372
MM-B – Marginal Histogram-Based Method.....	372
EDX – Extrapolation Directed Crossover.....	373
TDX – Trimodal Distribution Crossover	373
CIXL2 – Confidence Interval Based Crossover.....	373
GAQPR – Quantum Probability Representation-Based Method	374
LUNDX-m – Latent Variable Crossover	374
Indeks słów kluczowych, autorów i funkcji testowych.....	375

1. Wstęp

Literatura dotycząca algorytmów genetycznych lub, szerzej, obliczeń ewolucyjnych obfituje w szereg znakomitych książek i artykułów mających charakter tekstów wprowadzających lub tekstów przeglądowych. Teksty te koncentrują się na prezentacji najistotniejszych (lub w danej chwili najpopularniejszych) metod – selekcji, rekombinacji, mutacji etc., jednocześnie ze zrozumiałych względów pomijając (lub tylko wymieniając) większość dorobku z tej dziedziny. Podobnie rzecz ma się zarówno w wypadku serwisów internetowych poświęconych algorytmom genetycznym, jak i oprogramowania wykorzystującego algorytmy genetyczne – zakres prezentowanych lub wykorzystywanych metod jest mocno ograniczony. W tej sytuacji początkujący (choć nie tylko) w tej dziedzinie badacz skazany jest na samodzielne zgłębianie literatury w poszukiwaniu mniej popularnych metod, nowych inspiracji lub odpowiedzi na pytanie, czy metoda, nad którą sam obecnie pracuje, jest nowa. Ostatecznie niezwykle często się zdarza, a liczne tego dowody można znaleźć w tej książce, że publikowana *nowa* metoda powiela inną już istniejącą lub w bardzo niewielkim i mało istotnym stopniu ją modyfikuje. Potrzeba istnienia wyczerpujących opracowań jest więc oczywista i właśnie ta motywacja leżała u podstaw pomysłu napisania tej książki.

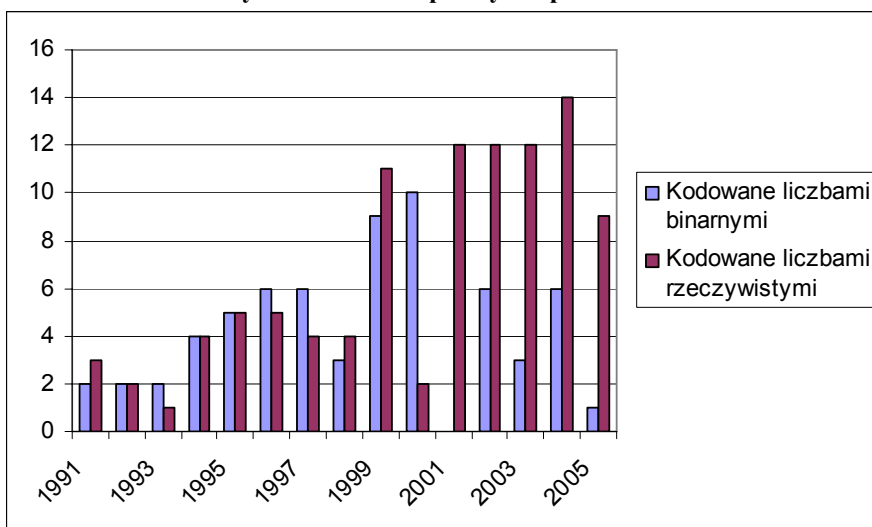
Książka ta jest pierwszą z całej serii książek, nad którą pracuję, a której celem jest dostarczenie możliwie jak najszerszego przeglądu metod opracowanych w dziedzinie algorytmów genetycznych i przedstawienie syntezy obecnego stanu ich teorii. Z charakteru tych książek wynika konieczność ścisłej koncentracji na wybranych obszarach tematycznych. Wybór tych obszarów, aczkolwiek dokonywany przeze mnie arbitralnie, będzie (mam nadzieję) odzwierciedlał stopień ich istotności, jak i popularności. Stąd też w tej książce, która zapoczątkuje serię, przedstawiony zostanie najistotniejszy dla algorytmów genetycznych operator – operator krzyżowania, a obszarem jego stosowania będą problemy optymalizacji numerycznej. Kolejne pozycje z serii będą poświęcone operatorom mutacji, selekcji i reprodukcji w tym samym obszarze stosowania, po czym skoncentruję się na problemach optymalizacji wielokryterialnej, aby na końcu zająć się obszarem problemów kombinatorycznych.

Układ książki jest następujący. Pierwsza część książki stanowi krótkie wprowadzenie do teorii algorytmów genetycznych. Następnie przedstawię 11 *operatorów standardowych*, gdzie pod pojęciem „standardowe” rozumiem te operatory, które najczęściej pojawiały się w literaturze przedmiotu w latach osiemdziesiątych i na początku lat dziewięćdziesiątych jako punkt odniesienia dla nowo publikowanych metod. Operatory standardowe prezentuję w skrócony w porównaniu z pozostałymi sposób, stąd też kilka operatorów, które bez wątpienia można by zaliczyć do grupy standardowych (np. krzyżowanie arytmetyczne), zostało przedstawionych w dalszych częściach książki w celu pełnej ich prezentacji. Trzecia część książki prezentuje 66 operatorów opracowanych dla problemów kodowanych liczbami binarnymi, a czwarta 89 operatorów opracowanych dla problemów kodowanych liczbami rzeczywistymi. Podział ten w przypadku wielu prezentowanych operatorów jest poniekąd sztuczny, bowiem mogą być one stosowane do rozwiązywania zarówno jednej, jak i drugiej klasy problemów, stąd też swoją decyzję o zaprezentowaniu konkretnego operatora w grupie operatorów kodowanych liczbami binarnymi lub

rzeczywistymi opierałem na tekstach źródłowych, w których autorzy zazwyczaj określali obszar zastosowań operatora. Ostatnia część książki zawiera listę operatorów bazujących na metodach statystycznych (operatorów statystycznych). Operatory te nie zostały tu szczegółowo zaprezentowane, jednak z uwagi na fakt, iż wielu autorów określa je także mianem operatorów krzyżowania, dla zainteresowanych czytelników wskazane zostały teksty źródłowe wraz z tekstami dodatkowymi. W toku całej książki, obok już funkcjonujących w polskich publikacjach nazw, zaproponowane zostały nowe polskie nazwy omawianych operatorów krzyżowania. Oprócz tego zachowane zostały oryginalne (angielskojęzyczne) nazwy operatorów głównie z uwagi na fakt, iż poza kilkoma wyjątkami, większość tych nazw nie pojawiła się jeszcze w polskojęzycznej literaturze przedmiotu.

Ilościowe podsumowanie niech stanowi rysunek 1, gdzie przedstawione zostały ilości operatorów opisanych w tej książce, z podziałem na lata ich publikacji.

Rysunek 1. Ilość opisanych operatorów



W drugiej i trzeciej części książki każdy operator prezentowany jest wedle tego samego, następującego schematu:

- Słowa kluczowe – mają pomóc w przeszukiwaniu książki i wzajemnym kojarzeniu prezentowanych w niej operatorów.
- Motywacja – wskazanie motywacji leżącej u podstaw opracowania danego operatora; motywacja ta była albo formułowana przez autorów explicite, albo wskazana jest przeze mnie w sposób arbitralny.
- Źródło – tekst źródłowy wraz ze wskazaniem adresu serwisu internetowego, skąd można ten tekst pobrać; w większości serwisy te są serwisami publicznymi – bezpłatnymi.
- Czytaj także – zalecane dodatkowe teksty, których tematyka jest bezpośrednio związana z omawianym operatorem; wybór tych tekstów, aczkolwiek dokonywany przeze mnie arbitralnie, opiera się głównie na liście bibliograficznej zawartej w tekście źródłowym lub też wskazuje teksty omawiające dalszy rozwój danego operatora lub

innych operatorów ideowo z nim związanych; zalecanym tekstom także towarzyszą adresy serwisów, skąd można je pobrać.

- Patrz także – inne operatory, z którymi w mojej opinii warto się zapoznać w kontekście danego operatora.
- Algorytm – prezentuje w formie pseudokodu omawiany operator, nierzadko w kilku wariantach; zdecydowałem się na tę formę prezentacji operatora, ponieważ w większości przypadków umożliwia ona natychmiastowe zastosowanie operatora w praktyce, z drugiej zaś strony nie chciałem stosować konkretnego języka programowania, ponieważ dodatkowo pojawiające się w kodzie elementy wynikające z gramatyki języka mogłyby utrudnić zrozumienie demonstrowanego operatora. Prezentowany algorytm często może odbiegać od swej oryginalnej postaci przedstawionej w tekście źródłowym. Dzieje się tak wtedy, gdy postać ta miała ścisły związek z problemem, dla którego operator został opracowany, jednak w każdym wypadku kluczowa idea operatora została tu zaprezentowana.
- Komentarze – komentarze lub opis prezentowanego operatora, zależnie od tego, czy w mojej opinii pseudokod algorytmu stanowi wystarczający opis operatora, czy nie.
- Funkcje testowe – lista standardowych funkcji lub problemów testowych, o ile takowe zostały zastosowane w toku eksperymentów z użyciem omawianego operatora.
- Porównano z – lista innych operatorów krzyżowania, z którymi porównywany był (w tekście źródłowym) prezentowany operator.

Aczkolwiek może to się spotkać z dezaprobatą, w toku całego tekstu traktuję jako synonimy takie pary pojęć, jak „rekombinacja – krzyżowanie”, „wektor rozwiązania – chromosom”, „gen – zmienna”, „generacja – iteracja”. Ponadto, o ile nie zostanie *explicite* wskazane, że jest inaczej, w toku całego tekstu stosuję następujące oznaczenia:

t – licznik generacji (iteracji, numeru populacji)

M – maksymalna liczba generacji (iteracji) algorytmu

$P()$ – populacja wektorów rozwiązania (chromosomów)

$P(0)$ – początkowa populacja

$P(t)$ – bieżąca populacja (populacja rodziców)

$P(t+1)$ – następna populacja (populacja potomków)

L_p – licznosc populacji (rozmiar populacji)

p_{cross} , p_c – prawdopodobieństwo krzyżowania

p_m – prawdopodobieństwo mutacji

n – długość wektora rozwiązania (chromosomu)

Rnd – losowa (o rozkładzie równomiernym) liczba rzeczywista, $Rnd \in (0, 1)$; jeżeli Rnd występuje w pętli algorytmu, wtedy w każdym kroku tej pętli jej wartość wyznaczana jest ponownie

Operatory dla problemów kodowanych liczbami binarnymi

$A^{(t)} = (a_1^{(t)}, \dots, a_n^{(t)}) \forall i a_i^{(t)} \in \{0, 1\}$ – binarny wektor rozwiązania (wektor kodowany liczbami binarnymi) (chromosom binarny)

$f(A^{(t)})$ – przystosowanie binarnego wektora rozwiązania $A^{(t)}$

$\{A_1^{(t)}, \dots, A_k^{(t)}\} \in P(t)$ – binarne wektory rozwiązania (chromosomy binarne)

$A_j^{(t)} = (a_{j1}^{(t)}, \dots, a_{jn}^{(t)}) \forall i, j a_{ji}^{(t)} \in \{0, 1\}$

$f(A_j^{(t)})$ – przystosowanie binarnego wektora rozwiązania $A_j^{(t)}$

Operatory dla problemów kodowanych liczbami rzeczywistymi

$X^{(t)} = (x_1^{(t)}, \dots, x_n^{(t)}) \in R^n$ – rzeczywisty wektor rozwiązania (wektor kodowany liczbami rzeczywistymi) (chromosom rzeczywisty)

$\forall i x_i^l \leq x_i \leq x_i^u$, gdzie:

x_i^l – dolne ograniczenie dziedziny i -tej zmiennej (i -tego genu)

x_i^u – górne ograniczenie dziedziny i -tej zmiennej (i -tego genu)

$f(X^{(t)})$ – przystosowanie rzeczywistego wektora rozwiązania $X^{(t)}$

$\{X_1^{(t)}, \dots, X_k^{(t)}\} \in P(t)$ – rzeczywiste wektory rozwiązania (chromosomy rzeczywiste)

$\forall i, j X_j^{(t)} = (x_{j1}^{(t)}, \dots, x_{jn}^{(t)}) \in R^n, x_i^l \leq x_{ji} \leq x_i^u$, gdzie:

x_i^l – dolne ograniczenie dziedziny i -tej zmiennej (i -tego genu)

x_i^u – górne ograniczenie dziedziny i -tej zmiennej (i -tego genu)

$f(X_j^{(t)})$ – przystosowanie rzeczywistego wektora rozwiązania $X_j^{(t)}$

Po wydaniu pełnej serii planowanych kompendiów planuję raz na dwa lata przygotowywać suplementy obejmujące swoją zawartością kolejne dwa lata, licząc od daty wydania danego kompendium. Będę je rozsyłał do każdego chętnego czytelnika. Stąd, jeżeli chcesz w przyszłości otrzymywać taki suplement, daj mi o tym znać e-mailem.

2. Krótko o algorytmach genetycznych

Historia

Karol Darwin (1809-1882), autor teorii ewolucji, Grzegorz Mendel (1822-1884), autor teorii podstawowych zasad dziedziczenia, Hugo de Varis (1848-1935), autor teorii mutacji genów, Walter Sutton (1877-1916) i Teodor Boveri (1862-1915), autorzy teorii chromosomu – to uczeni, którzy położyli podwaliny pod współczesny stan wiedzy, który stał się inspiracją dla twórców teorii algorytmów genetycznych (AG) (ang. *Genetic Algorithms* (GA)). Pierwsze prace, które można uznać za związane z tą teorią, to prace Barricelliego [1][2][3], Boxa [4], Fräsera [5][6], Friedberga [7], Friedmana [8], Bledsoe’a [9], Bremermanna [10], jednak za twórcę teorii algorytmów genetycznych powszechnie uznaje się Johna Hollanda. John Holland w latach sześćdziesiątych ubiegłego stulecia sformułował [11], a następnie rozwijał, wraz ze swym zespołem przez kolejne dwie dekady [12][13], obecnie znaną postać AG.

Gwałtowny wzrost zainteresowania, a co za tym idzie i rozwój badań nad AG, datuje się na przełom lat siedemdziesiątych i osiemdziesiątych. Publikowane są prace obecnie uznane za klasyczne – De Jonga [14], Goldberga [16], Davisa [17]. Organizowane są pierwsze konferencje naukowe, tworzone są stowarzyszenia i publikowane pierwsze periodyki o tematyce ściśle związanej z AG. W Polsce za ośrodki naukowe inicjujące prace poświęcone metodom Sztucznej Inteligencji, a w tym AG, uznaje się Politechnikę Śląską, Politechnikę Warszawską i Akademię Górniczo-Hutniczą w Krakowie.

Równolegle do rozwoju AG rozwijane są metody pokrewne, również wywodzące się z nurtu ewolucyjnego. Do metod tych zalicza się:

- Programowanie ewolucyjne (PE) (ang. *Evolutionary Programming* (EP)) – metoda zaproponowana przez Lawrence’a Fogela [18][19];
- Strategie ewolucyjne (SE) (ang. *Evolution Strategies* (ES)) – metoda zaproponowana przez Ingo Rechenberga [20][21] i Hansa-Paula Schwefela [22][23];
- Programowanie genetyczne¹ (PG) (ang. *Genetic Programming* (GP)) – metoda zaproponowana przez Johna Koza [24][25], przy czym (jak zaznacza sam Koza) jego prace wywodziły się od prac Smitha [26] i Cramera [27].

Współcześnie wszystkie metody wywodzące się z nurtu ewolucyjnego, to jest AG, PE, SE i PG często określa się wspólnym mianem metod obliczeń ewolucyjnych (OE) (ang. *Evolutionary Computation* (EC)). Nie oznacza to wcale, że metody te *zlały się w jedno*, bowiem wywodząc się z tego samego źródła, metody te często przenikają się i inspirują nawzajem, lecz nadal rozwijane są najczęściej indywidualnie.

W kolejnych podrozdziałach bieżącego rozdziału zaprezentowana zostanie kanoniczna postać AG. Forma tej prezentacji będzie zgodna z przyjętą przeze mnie w tej książce zwartą formą opisu: algorytm w postaci pseudokodu plus komentarze. Sądzę, że takie podejście z jednej strony nie utrudnia zrozumienia prezentowanych metod i pojęć, a z drugiej strony pozwala jednocześnie na *naturalne* przejście z fazy nauki do fazy

¹ Niekiedy traktuje się PG jako AG przetwarzające specyficzną postać wektora rozwiązania, jaką zazwyczaj jest wyrażenie języka LISP, jednak większość publikacji postrzega PG jako odrębną od AG metodę.

konkretnej implementacji. Czytelnikom preferującym mniej syntetyczne podejście polecam dodatkowe zapoznanie się z tekstami wskazanymi w części „Czytaj także”, umieszczonej na końcu bieżącego rozdziału.

Podrozdział pierwszy przedstawi ogólną postać AG. Kolejne będą prezentować różne warianty poszczególnych elementów składowych AG, aby w ostatnim pokazać ostatecznie pełną, jedną z wielu możliwych, jak się okaże, postać AG.

Kanoniczna postać AG

Algorytm

1. $t = 1$
2. generuj populację $P(t)$
3. do while *warunek_końca_jest_spełniony* = False
4. oceń populację $P(t)$
5. z populacji $P(t)$ wyselekcjonuj zbiór rodziców I'
6. stosując operatory genetyczne na I' utwórz potomków
7. utwórz nową populację $P(t + 1)$
8. $t = t + 1$
9. loop

Komentarze

- AG przeszukuje przestrzeń rozwiązań pewnego z góry zadanego problemu. Przeszukiwanie to realizowane jest w celu wyznaczenia rozwiązania najlepszego (lub niekiedy wielu rozwiązań), przy czym to, co rozumiemy pod pojęciem *najlepszy*, zależy ściśle od charakteru rozwiązywanego problemu.
- AG nie przetwarza pojedynczego rozwiązania (jak większość metod przeszukiwania przestrzeni rozwiązań), lecz cały ich zbiór zwany populacją². W pierwszym kroku AG (wiersz 2) realizowane jest generowanie *populacji początkowej* losowo lub deterministycznie.
- Kolejne kroki (wiersze 4-7) powtarzane są tak długo, aż nie zostanie spełniony (heurystyczny) warunek zatrzymania AG (wiersz 3). Naturalna ewolucja nie ma końca – trwa wiecznie, my jednak, rozwiązując konkretny problem, nie mamy aż tak dużo czasu.

² Terminologia stosowana w teorii AG wywodzi się z terminologii używanej w naukach biologicznych, co jest swoistym ukłonem w stronę tych nauk, jako że stanowiły one pierwotną inspirację dla AG; należy jednak podkreślić, że współczesna postać elementów składowych AG ma (poza terminologią) niewiele wspólnego ze swoimi *biologicznymi* odpowiednikami – pozostają one stałą inspiracją, lecz nikt (lub prawie nikt) nie próbuje w 100 procentach odzwierciedlać ich faktycznego funkcjonowania w formie sztucznych algorytmów.