

Algorytmy ewolucyjne

1. Charakterystyka

Algorytmy Ewolucyjne (Evolutionary Algorithms) – AE to zbiór metod, których wspólną cechą jest naśladowanie naturalnych procesów związanych z ewolucją istot żywych. Głównym elementem spajającym te metody jest wpływ presji ewolucyjnej na kolejne pokolenia organizmów i ich przystosowywanie się do niszy ekologicznej, w której żyją. Większość z tych metod korzysta także z operatorów naśladujących procesy związane z dziedziczeniem cech przez organizmy potomne, czyli krzyżowania i mutacji. Do algorytmów ewolucyjnych zalicza się m.in. następujące metody: algorytmy genetyczne, strategie ewolucyjne, programy ewolucyjne, programowanie genetyczne, ewolucję różnicową.

Jako przełom w rozwoju algorytmów ewolucyjnych uważa się prace Johna Hollanda z końca lat 60-tych XX wieku. Od lat 80-tych XX wieku metody te stały się bardzo popularne i są przedmiotem implementacji do czasów obecnych, nabierając charakteru metody już klasycznej.

Algorytmy ewolucyjne traktowane są jako uniwersalna metoda optymalizacji globalnej. Do swojego działania nie potrzebują wiedzy na temat pochodnych funkcji celu. Funkcja celu nie musi być zatem funkcją ciągłą. Jediną informacją na temat funkcji celu, jaką wymaga algorytm ewolucyjny, to określenie jej wartości w poszczególnych punktach w przestrzeni rozwiązań. W porównaniu z innymi metodami optymalizacyjnymi, ta cecha dawała zasadniczą przewagę algorytmom ewolucyjnym nad znanymi metodami. Stąd też bardzo duża popularność algorytmów ewolucyjnych, która szczególnie wyraża się w zastosowaniach do optymalizacji problemów o zmiennych dyskretnych.

Ogólna postać struktury algorytmu ewolucyjnego została przedstawiona na rysunku 2.9. Działanie AE rozpoczyna się od utworzenia populacji początkowej osobników, które stanowią potencjalne rozwiązania danego zadania. Najczęściej populację początkową tworzy się w sposób losowy. Znane są także implementacje wykorzystujące pewne dodatkowe algorytmy generujące zbiór rozwiązań początkowych.

Następnie dokonuje się oceny osobników z pokolenia początkowego, czyli w pierwszym kroku oblicza się wartość funkcji celu dla poszczególnych osobników. Na tej podstawie określa się, na ile poszczególne osobniki są przystosowane do sztucznej niszy ekologicznej, którą jest rozwiązywane zadanie. Te, które są lepiej przystosowane, otrzymują wyższą ocenę.

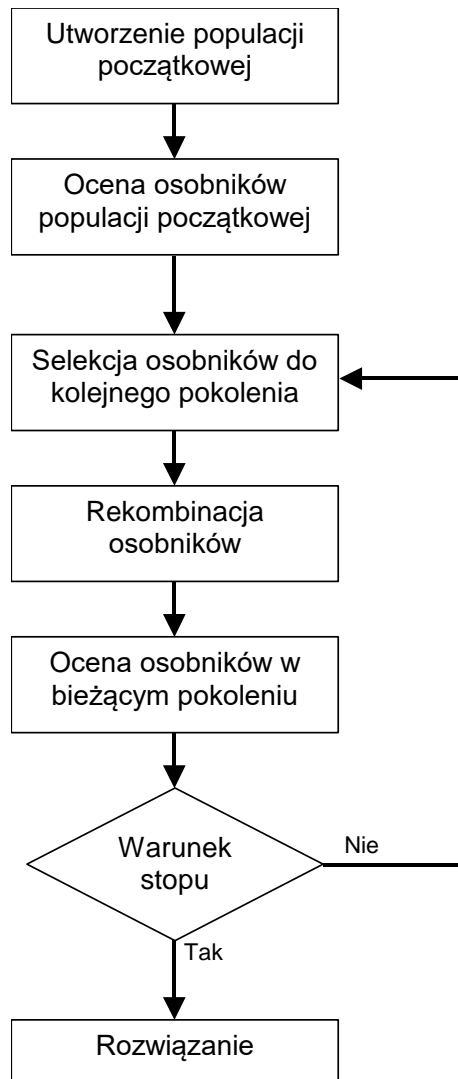
Ocena ta stanowi podstawę do wyboru danego osobnika do następnego pokolenia, czyli przeżycia osobnika.

Kolejnym etapem jest rekombinacja, której celem jest zmiana postaci wybranych osobników. Najczęściej wykorzystuje się w tym celu dwa operatory: krzyżowania i mutacji. Zadaniem operatora mutacji jest wprowadzanie drobnych zmian w losowo wybranych osobnikach. Natomiast celem działania operatora krzyżowania jest tworzenie osobników potomnych, które podobnie jak w naturze, będą stanowić mieszaninę cech osobników rodzicielskich.

Istotnym problemem w konstrukcji algorytmu ewolucyjnego jest sposób kodowania zadania, czyli zapisu potencjalnego rozwiązania. Jest on swego rodzaju kodem genetycznym osobników. Z jednej strony powinien on być możliwie najprostszy ze względu na koszt obliczeniowy, z drugiej zaś powinien zapewniać przetwarzanie przez algorytm ewolucyjny głównie rozwiązań dopuszczalnych. Należy także wziąć pod uwagę wpływ przyjętego rozwiązania na postać operatora krzyżowania. W przypadku zadań o zmiennych rzeczywistych nie stanowi to większego problemu. Natomiast w przypadku innych zadań, szczególnie kombinatorycznych, ten element decyduje o powodzeniu całego procesu optymalizacyjnego. Sposobów rozwiązania tego problemu jest bardzo wiele. W klasycznych algorytmach genetycznych przyjmowano zapis binarny parametrów zadania. W kolejnych generacjach algorytmów ewolucyjnych stosowano kodowanie bardziej naturalne – zbieżne z postrzeganiem problemów optymalizacyjnych przez ludzi.

Następnym etapem działania AE jest ocena przystosowania poszczególnych osobników w pokoleniu, tożsama z oceną osobników w populacji początkowej. Jediną różnicą może być tu uwzględnienie przystosowania osobników z pokoleń poprzednich.

Ostatnim elementem jest weryfikacja warunku zatrzymania algorytmu. Zwykle stosuje się w tym celu określoną liczbę pokoleń lub brak poprawy rozwiązania przez określoną ich liczbę.



Rysunek 3.9 Ogólna struktura algorytmu ewolucyjnego

2 Zastosowania

Pierwsze zastosowania algorytmów ewolucyjnych w elektroenergetyce pojawiły się na przełomie lat 80-tych i 90-tych XX wieku. Metody te od razu zyskały dużą popularność, z uwagi na swoją skuteczność i odporność w stosunku do różnorodnych problemów. Jednak liczba publikacji poświęconych ich aplikacjom w elektroenergetyce jest prawdopodobnie o rząd wielkości mniejsza (10^3) od zastosowań sztucznych sieci neuronowych. Wynika to ze specyfiki rozwiązywania problemów za pomocą algorytmów ewolucyjnych i sposobu ich działania. Powoduje to deficyt gotowych rozwiązań, jakimi dysponujemy w przypadku sieci neuronowych, które mogą być użyte przez badaczy. Praktycznie do każdego nowego zadania należy stworzyć specjalizowaną wersję algorytmu ewolucyjnego, oprogramować ją,

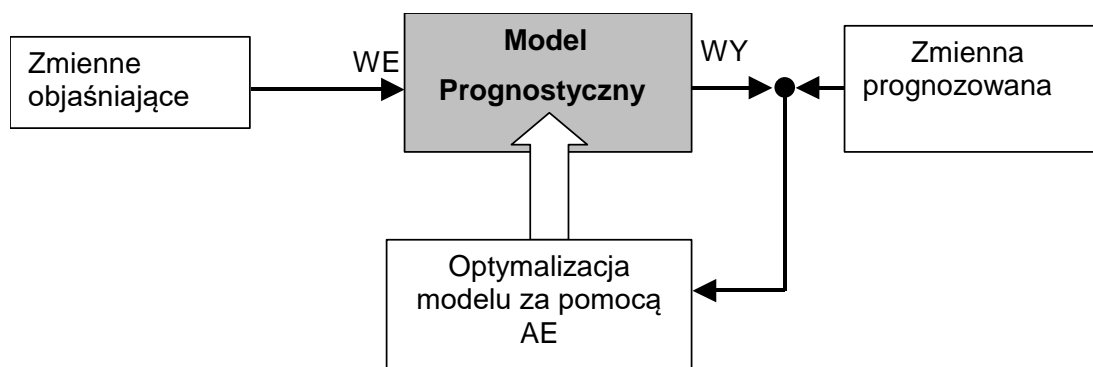
a następnie zoptymalizować parametry jej działania. Jest to proces dość czasochłonny i kosztowny, szczególnie w przypadku zadań o zmiennych dyskretnych.

Algorytmy ewolucyjne są z natury rzeczy uniwersalną metodą optymalizacyjną, która mimo zastosowania w różnych problemach, zawsze służy do znalezienia optimum pewnego zagadnienia szczegółowego. Stąd także dość duża liczba zastosowań hybrydowych, w których algorytm ewolucyjny optymalizuje parametry działania innej metody inteligencji obliczeniowej.

Prognostowanie

Algorytmy ewolucyjne najmniej licznie są reprezentowane w prognozowaniu. Jest to wynik ich optymalizacyjnej natury. Zastosowania algorytmów ewolucyjnych w prognozowaniu polegają na optymalizacji parametrów działania metod typowo prognostycznych. Najczęściej występującym rozwiązaniem jest optymalizacja wag połączeń sztucznej sieci neuronowej za pomocą algorytmu ewolucyjnego. Możliwe jest także wykorzystanie AE do optymalizacji struktury, konfiguracji i parametrów uczenia sieci neuronowej.

W podobny sposób można optymalizować działanie innych metod używanych do prognozowania. Ogólną ideę wykorzystania algorytmu ewolucyjnego w prognozowaniu do budowania modelu prognostycznego przedstawiono na rys 3.10. Przypomina ona uczenie z nauczycielem w przypadku sieci neuronowej, gdzie zadaniem sieci neuronowej jest znalezienie zależności pomiędzy zmiennymi objaśniającymi a objaśnianymi (prognozowanymi). Największą wadą takiego podejścia jest dość kosztowny obliczeniowo proces tworzenia modelu, a w przypadku skomplikowanych modeli - nie zawsze odpowiednia precyzja. Natomiast istotną zaletą takiego rozwiązania jest możliwość stosowania dowolnej funkcji celu określającej jakość modelu lub nawet wielu funkcji w optymalizowaniu wielokryterialnym (np. pareto) takiego modelu.

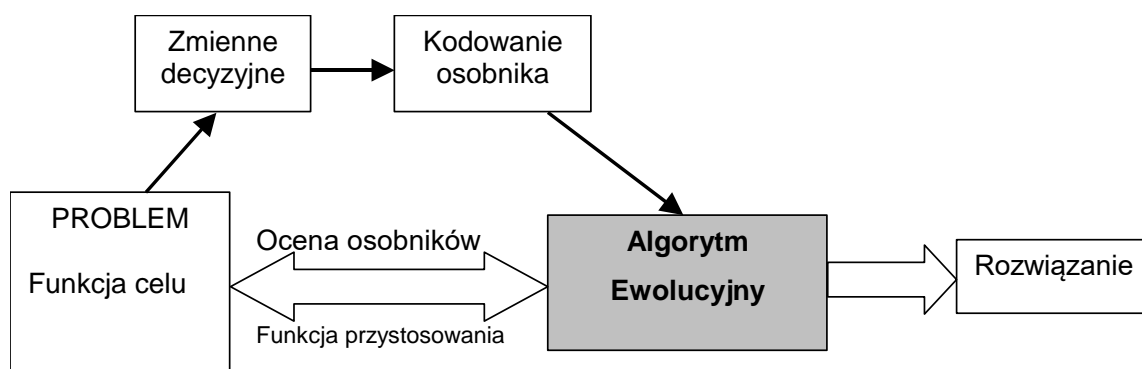


Rysunek 3.10 Ogólna idea wykorzystania algorytmu ewolucyjnego w prognozowaniu

Optymalizacja

Optymalizacja to największa, najobszerniejsza i najbardziej naturalna grupa zastosowań algorytmów ewolucyjnych w elektroenergetyce. Pierwszym etapem wykorzystania algorytmu ewolucyjnego do optymalizacji jest określenie funkcji celu, którą należy zoptymalizować. Przy czym funkcja ta nie musi być ciągła ani różniczkowalna. Określenie zmiennych decyzyjnych najczęściej wynika z postawionego zadania i funkcji celu. Zbiór tych zmiennych staje się zwykle chromosomem/osobnikiem w algorytmie ewolucyjnym, a samo odwzorowanie staje się sposobem kodowania (rys. 3.11).

Niestety takie podejście, szczególnie w przypadku zadań kombinatorycznych, może nie prowadzić do sukcesu – ponieważ rozmiar przeszukiwanej przestrzeni rozwiązań jest zbyt duży. Wtedy najczęściej opracowuje się specjalizowany sposób kodowania, który z jednej strony silnie ogranicza wielkość przeszukiwanej przestrzeni, a z drugiej strony nie wyklucza tych podprzestrzeni, w których potencjalnie może znajdować się optimum globalne.



Rysunek 3.11 Ogólna idea wykorzystania algorytmu ewolucyjnego w optymalizacji

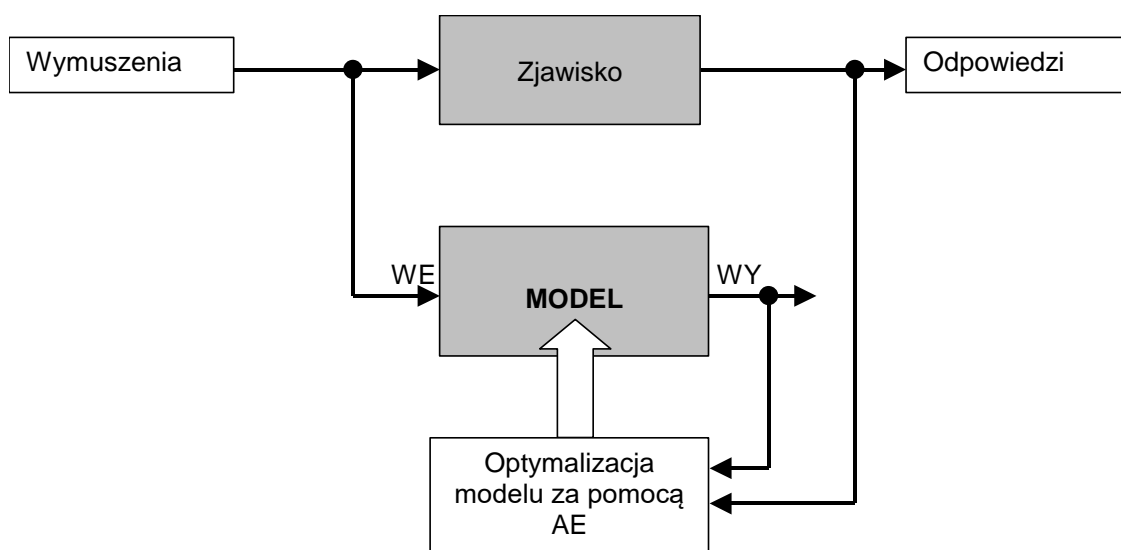
Ze względu na swoje ogólne cechy algorytmy ewolucyjne zostały bardzo szeroko zastosowane w różnych grupach problemów:

1. Optymalizacja struktur sieci elektroenergetycznych
2. Optymalizacja konfiguracji sieci elektroenergetycznych
3. Optymalizacja pracy jednostek wytwórczych klasycznych i odnawialnych
4. Optymalizacja napięć w sieciach elektroenergetycznych
5. Optymalizacja rozptywu mocy czynnej
6. Optymalizacja konstrukcji maszyn elektrycznych
7. Optymalizacja konstrukcji linii elektroenergetycznych
8. Optymalizacja rozmieszczenia generacji rozproszonej
9. Optymalizacja ekonomicznego rozdziału obciążeń

10. Optymalizacja rozptywu mocy biernej
11. Optymalizacja prac eksploatacyjnych
12. Optymalizacja zdolności przesyłowych

Identyfikacja

Zasadnicze wykorzystanie algorytmów ewolucyjnych w identyfikacji polega na optymalizowaniu za ich pomocą parametrów pewnego modelu opisującego interesujące nas zjawisko (rys. 3.12.). W tym zakresie użycie metody jest podobne do zastosowania algorytmów ewolucyjnych w prognozowaniu.



Rysunek 3.12 Ogólna idea wykorzystania algorytmu ewolucyjnego w identyfikacji

Wyżej przedstawione podejście prezentowane jest w aplikacjach algorytmów ewolucyjnych do identyfikacji parametrów modeli matematycznych maszyn elektrycznych. W podobny sposób wykorzystano je do modelowania odbiorów w sieciach elektroenergetycznych, identyfikacji parametrów sieci elektroenergetycznych i estymacji obciążeń.

Innym sposobem wykorzystania algorytmów ewolucyjnych jest optymalizacja innej metody obliczeniowej. Przykładem może być wspomaganie wyznaczania rozptyłów mocy, odporności na zakłócenia, ocena niezawodności, ochrona przeciwporażeniowa, obliczenia elektromagnetyczne.

Sterowanie

Ze względu na specyfikę funkcjonowania, algorytmy ewolucyjne nie wykazują najlepszych własności do działania w czasie rzeczywistym. Stąd też ich zastosowanie w sterowaniu

zwykle polega na optymalizacji parametrów ogólnie pojętego algorytmu sterowania w taki sposób, by jak najlepiej realizował on postawione przed nim zadania. Podejście tego typu prezentowane jest w następujących aplikacjach:

1. Stabilizacja systemu elektroenergetycznego.
2. Automatyka zabezpieczeniowa linii rozdzielczych.
3. Automatyka przeciwwawaryjna i restytucyjna.
4. Regulacja źródeł wytwórczych.

Parametrami algorytmu sterowania są nie tylko parametry pewnej funkcji matematycznej realizowanej przez klasyczny regulator. Mogą to być także listy działań, jakie należy podjąć w przypadku wystąpienia określonych okoliczności. Takie podejście do sterowania wymaga opracowania szeregu optymalnych „scenariuszy” postępowania, które - jako zadanie optymalizacyjne - mają charakter kombinatoryczny i najczęściej bardzo skomplikowany. Właśnie w takich zastosowaniach widać przewagę algorytmów ewolucyjnych nad innymi metodami.

2. Funkcje testowe

- Funkcja Ackley'a

$$f(x) = -20 \exp \left[-\frac{1}{5} \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2} \right] - \exp \left[\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i) \right] + 20 + e$$

gdzie: $n = 1, 2, \dots$; and $-32.768 \leq x_i \leq 32.768$ for $i = 1, 2, \dots, n$. Minimum globalne funkcji $f_* = 0$ osiągnięte jest dla $x_* = (0, 0, \dots, 0)$.

- Funkcja Rosenbrocka:

$$f(x) = \sum_{i=1}^{n-1} \left[(x_i - 1)^2 + 100(x_{i+1} - x_i^2)^2 \right]$$

gdzie: $n = 1, 2, \dots$; and $-30.0 \leq x_i \leq 30.0$ for $i = 1, 2, \dots, n$. Minimum globalne funkcji $f_* = 0$ osiągnięte jest dla $x_* = (1, 1, \dots, 1)$.

- Funkcja DeJonga:

$$f(x) = \sum_{i=1}^n x_i^2$$

gdzie: $n = 1, 2, \dots$; oraz $-100.0 \leq x_i \leq 100.0$ dla $i = 1, 2, \dots, n$. Minimum globalne funkcji $f_* = 0$ osiągnięte jest dla $x_* = (0, 0, \dots, 0)$.

- Funkcja Griewanka

$$f(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \quad (13)$$

gdzie: $n = 1, 2, \dots$; oraz $-600.0 \leq x_i \leq 600.0$ dla $i = 1, 2, \dots, n$. Minimum globalne funkcji $f_* = 0$ osiągnięte jest dla $x_* = (0, 0, \dots, 0)$.

- Funkcja Rastrigina :

$$f(x) = 10n + \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i)] \text{ gdzie: } n = 1, 2, \dots; \text{ oraz } -5.12 \leq x_i \leq 5.12 \text{ for } i$$

$= 1, 2, \dots, n$. Minimum globalne funkcji $f_* = 0$ osiągnięte jest dla $x_* = (0, 0, \dots, 0)$.

- Funkcja Schwefela:

$$f(x) = -\sum_{i=1}^n x_i \sin(\sqrt{|x_i|})$$

gdzie: $n = 1, 2, \dots$; oraz $-500 \leq x_i \leq 500$ dla $i = 1, 2, \dots, n$. Minimum globalne funkcji $f_* \approx -418.9829 \cdot n$ osiągnięte jest dla $x_* = (420.9687, 420.9687, \dots, 420.9687)$.

W celu zapewnienia dodatniości funkcji dla wszystkich numerów wymiarów oraz zmiennych, następująca modyfikacja funkcji powinna podlegać zastosowaniu.

$$f(x) = -\frac{1}{n} \sum_{i=1}^n x_i \sin(\sqrt{|x_i|}) + 500$$

Minimum globalne funkcji $f_* \approx 81.0171$ występuje w tym samym punkcie, co dla funkcji oryginalnej, tj. $x_* = (420.9687, 420.9687, \dots, 420.9687)$.

Celem optymalizacji każdej funkcji jest jej minimalizacja. Algorytmy były jednak budowane tak, aby maksymalizować funkcję przystosowania. Konieczne było więc przejście między funkcją celu a przystosowania, dokonywane docelowo za pomocą:

$$F_{fitness} = \frac{1}{F_{obj} + 1}$$

3. Przebieg ćwiczenia

- Oprócz funkcji domyślnej zespół wybiera 3 inne funkcje testowe
- W pliku PROTOTYP.goi należy ustawić odpowiednią dla danej funkcji testowej liczbę zmiennych oraz zakresy zmiennych wejściowych
- Wybór kombinacji parametrów wejściowych w pliku auto_EA.bxt, umożliwiających zbadanie wpływu każdej zmiennej na efekty symulacji. Kombinacje pozostają takie same dla każdej funkcji testowej.
- Porównanie wyników na podstawie plików report_EA_auto.txt, w tym ocena jakości rozwiązania znalezionego oraz analiza zbieżności