

Piotr Łętkowski

*Instytut Nafty i Gazu – Państwowy Instytut Badawczy*

## Zastosowanie metod programowania genetycznego w procesie maksymalizacji wydobycia węglowodorów przy zastosowaniu symulatora złożowego

Artykuł poświęcono zastosowaniu metody programowania genetycznego dla celów optymalizacji wydobycia ropy naftowej na przykładzie testowego złoża węglowodorowego. Prezentowane zagadnienie optymalizacyjne jest prostym przykładem problemu optymalnej kontroli i polega na doborze wydajności wydobycia ropy naftowej w przyjętych przedziałach czasowych w taki sposób, aby w zadanym całkowitym czasie eksploatacji uzyskać maksymalne wydobyte sumaryczne przy minimalnym wydobyciu wody. Problem rozwiązano przy zastosowaniu algorytmu genetycznego, kodującego dozwolone wartości wydajności wydobycia z listy wartości dozwolonych. Z jednej strony działanie takie jest charakterystyczne dla metod programowania genetycznego, zaś z drugiej redukuje istotnie przestrzeń rozwiązań. W artykule zastosowano algorytm genetyczny Hollanda, dla którego zaimplementowano krzyżowanie wielopunktowe oraz adaptację prawdopodobieństw krzyżowania i mutacji na podstawie tzw. współczynnika zróżnicowania populacji. Działanie tak zdefiniowanego mechanizmu adaptacji jest następujące: jeżeli zróżnicowanie populacji rośnie, liniowo zwiększane jest prawdopodobieństwo krzyżowania, a zmniejszane prawdopodobieństwo mutacji; w przeciwnym wypadku (zróżnicowanie populacji maleje) działa mechanizm odwrotny, tzn. zmniejsza się prawdopodobieństwo krzyżowania, a zwiększa prawdopodobieństwo mutacji. Taka metoda z jednej strony gwarantuje różnorodność populacji, z drugiej zaś zapewnia dobrą eksploatację przestrzeni rozwiązań. Przeprowadzono szereg testów mających na celu zweryfikowanie efektywności algorytmu w zależności od liczby punktów krzyżowania (krzyżowanie 1-, 2-, 3-punktowe) oraz długości chromosomu. Wykonane testy wskazują na zadowalającą zbieżność algorytmu, niezależnie od wartości badanych parametrów. Przyjęcie funkcji w określonej postaci spowodowało premiowanie przez algorytm niższych wartości wydobycia, co wynika z nieliniowego przyrostu wydobycia wody dla wyższych wartości wydobycia ropy naftowej.

Słowa kluczowe: algorytmy genetyczne, programowanie genetyczne, optymalizacja wydobycia, symulacje złożowe.

### Application of Genetic Programming Methods for the Optimization of Hydrocarbon Production by using a Reservoir Simulator

The paper addresses the problem of oil production optimization by genetic programming methods. The specific example of the problem presented in the paper belongs to the class of, so called, optimal control problems. It consists in finding the time variable rates of oil production that result in the maximum of the total oil production while keeping the total water production at a minimum available level. The problem is solved by a genetic algorithm, that assumes the production rates from the list of the allowable values. This approach typical for genetic programming methods significantly reduces the space of possible solutions. The article uses the Holland genetic algorithm for which multi-point crossing has been implemented and the adaptation of crossing and mutation probabilities based on so the called coefficient of population variability. The adaptive mechanism makes the crossing probability increase and mutation probability decrease for population variability increasing with time, while the crossing probability decrease and mutation probability increase for the variability decreasing with time. This mechanism guarantees the population variability to be at an appropriate level and at the same time, the extrapolation process for the solution space to be effective. Several tests were performed to verify the actual effectiveness of the algorithm for various number of crossing points (1, 2, 3 – crossing points) and chromosome length. Their results show a satisfactory convergence of the method to the final solution independent of the varying parameters values. Adopting a function in a specific form resulted in an algorithm for lower mining values, resulting from a nonlinear increase in water extraction for higher oil production values.

Key words: genetic algorithm, geneting programming, production optimization, filed simulations.

## Wstęp

Istnieje wiele problemów, których rozwiązanie klasycznymi metodami skazane jest na niepowodzenie. Niektóre z nich powstały jako zagadnienia teoretyczne, jednak ich efektywne rozwiązania posiadają istotne znaczenie praktyczne w wielu dziedzinach nauki i techniki. Przykładem może być problem „komiwojażera” czy „plecakowy”. Niedoskonałość klasycznych metod optymalizacji w zastosowaniu do przedstawionych zagadnień spowodowała, że rozpoczęto poszukiwania metod działających według innych zasad. Szukając inspiracji, skierowano uwagę na świat przyrody i okazało się, że ewolucja rozwiązała już takie problemy. Jednym z najbardziej popularnych zastosowań rozwiązań ze świata biologii wykorzystanym przy wyjaśnianiu zagadnień optymalizacyjnych są algorytmy genetyczne [18] mające źródło w teorii ewolucji gatunków i zaliczane dzisiaj do rodziny ewolucyjnych metod optymalizacyjnych [3, 24, 25]. Spośród wielu zastosowań algorytmów genetycznych na zainteresowanie zasługują metody tzw. programowania genetycznego spopularyzowane przez Johna Koze [22, 23]. Programowanie genetyczne to próba zbudowania algorytmu, który będzie tworzył optymalne z przyjętego punktu widzenia programy (schematy, procedury) w oparciu o metodologię algorytmów genetycznych. Metody programowania genetycznego znalazły na przykład zastosowanie przy rozwiązywaniu praktycznych problemów z dziedziny tzw. *schedulingu*, czyli optymal-

nego zarządzania procesami [1, 8, 19, 26–28, 31], analizy i klasyfikacji dużych zbiorów danych [35] czy prognozowania zachowania złożonych układów fizycznych [20]. Szereg prac jest również poświęconych aspektom teoretycznym stosowania metod programowania genetycznego oraz ich łączenia z innymi metodami optymalizacji [4–7, 10, 11, 17, 29, 30, 32–34]. Osobną grupę stanowią zastosowania metod programowania genetycznego do rozwiązywania problemów z dziedziny szeroko rozumianej inżynierii naftowej [9, 12]. Szczególne znaczenie mają tutaj zagadnienia związane z optymalną kontrolą i zarządzaniem wydobyciem węglowodorów [2, 15, 16]. Problemy te są istotne, gdyż decyzja o sposobie eksploatacji złoża ma poważne konsekwencje finansowe i w dużym stopniu przesądza o wartości przedsiębiorstw naftowych.

Tematem prezentowanej pracy jest próba zweryfikowania możliwości zastosowania algorytmu genetycznego (programowania genetycznego) do optymalnego doboru wydajności wydobycia ropy naftowej dla prognozowanego czasu wydobycia oraz przyjętego kryterium optymalizacyjnego. Tak postawione zagadnienie optymalizacyjne jest przykładem prostego problemu optymalnej kontroli. W artykule przedstawiono podstawy teoretyczne analizowanego zagadnienia oraz wyniki działania algorytmu optymalizacyjnego realizującego optymalizację dla przyjętego kryterium.

## Podstawy teoretyczne i pojęcia podstawowe

Niestety nie istnieje żadna spójna teoria systematyzująca wiedzę na temat algorytmów genetycznych (w tym metod programowania genetycznego). Do dyspozycji mamy:

- twierdzenie Hollanda (tzw. twierdzenie o schematach) mówiące, że:

*Wąskie schematy, niskiego rzędu i dobrze dostosowane rozprzestrzeniają się w kolejnych pokoleniach w sposób wykładniczy;*

- hipotezę „cegiełek” (inaczej „bloków budujących”), według której:

*Dobrze dostosowane schematy niskiego rzędu i o małej rozpiętości zwane cegiełkami są nieustannie wybierane, zestawiane i powielane, tworząc ciągi kodowe o potencjalnie wyższym przystosowaniu.*

Powyższe twierdzenia wymagają podania definicji schematu, jego rzędu oraz rozpiętości. Schemat to w teorii algorytmów genetycznych wzorec opisujący podzbiór wszystkich ciągów podobnych do siebie ze względu na ustalone pozycje. Rzędem schematu nazywamy liczbę ustalonych

pozycji w schemacie, natomiast rozpiętość schematu to odległość między skrajnymi, ustalonymi pozycjami w schemacie.

Podane warunki są zgodne z intuicją. Jeśli schemat jest małego rzędu i ma małą rozpiętość, jest mniejsza szansa, że w trakcie krzyżowania zostanie rozdzielony. Lepsze przystosowanie gwarantuje mu natomiast wejście jego reprezentantów do następnego pokolenia.

Powyższe twierdzenia tłumaczą mechanizm działania algorytmów genetycznych jakościowo i nie dają żadnych wskazań praktycznych. Dla ogólnego przypadku nie istnieją dowody ich zbieżności. Twierdzenia takie zostały udowodnione tylko dla prostych funkcji testowych (np. funkcji kwadratowej) i to w przypadku kodowania binarnego.

W literaturze poświęconej algorytmom genetycznym (ewolucyjnym) stosowana jest terminologia mająca swoje źródło w teorii ewolucji gatunków. Poniżej zestawiono podstawowe pojęcia stosowane w niniejszej pracy.

**Osobnik** – dowolne, możliwe rozwiązanie problemu.

Zadaniem algorytmu jest znalezienie w wyniku reprodukcji i rekombinacji (krzyżowania i mutacji) osobnika realizującego ekstremum globalne. W opisywanym przypadku osobnikiem jest plik wsadowy do symulatora złożowego opisujący w postaci procedury zmiany wydajności wydobywania ropy naftowej ze złoża.

**Populacja** – rodzina osobników.

**Fenotyp** – parametry rozwiązania podlegające ocenie.

**Genotyp** – kompletny i jednoznaczny zapis parametrów osobnika w postaci zakodowanej.

**Gen** – pojedynczy element genotypu.

**Chromosom** – miejsce przechowywania genotypu osobnika, czyli jego numeryczna reprezentacja.

**Funkcja przystosowania** – funkcja pozwalająca określić jakość osobnika z punktu widzenia rozwiązywanego problemu.

**Kodowanie rozwiązań** – sposób zapisu dowolnego dopuszczalnego rozwiązania problemu w postaci genotypu osobnika.

Jakkolwiek w teorii programowania genetycznego znajduje zastosowanie powyższa terminologia, to jednak ze względu na swoją specyfikę metody te posiadają pewne cechy wyróżniające je spośród metod ewolucyjnych:

- populację stanowią programy komputerowe realizujące zadanie optymalizacyjne;
- funkcja przystosowania ma zdecydować, czy dany program, chociaż w przybliżeniu, wykonuje zadane działanie. Na przykład, jeśli chcemy stworzyć program, który zawsze będzie znajdował drogę wyjścia z labiryntu, funkcja oceny będzie sprawdzała, jak blisko wyjścia program zakończył działanie. Te programy, który znajdują wyjście, mają najwyższą wartość funkcji przystosowania, natomiast te, które nie oddalają się od wejścia, najmniejszą;
- konieczne jest określenie alfabetu dozwolonych symboli i wartości, jakie mogą pojawić się w chromosomie. Mogą to być np. liczbowe wartości zmiennych, zmienne, operatory logiczne, operatory arytmetyczne, słowa kluczowe itd.

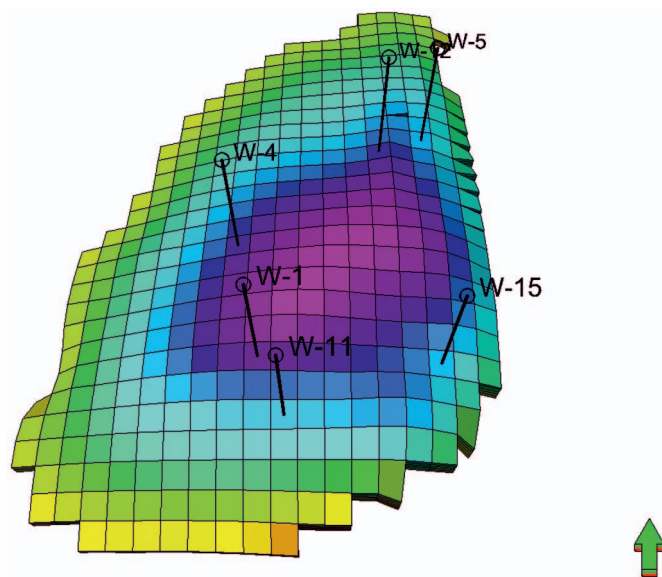
W klasycznym przypadku programowania genetycznego programy są przedstawiane jako struktury drzewiaste będące graficzną reprezentacją działania programu. Dla naszych celów wystarczająca będzie jednak reprezentacja liniowa. Główną różnicą pomiędzy klasycznym algorytmem genetycznym a programowaniem genetycznym jest w tym przypadku fakt, że w tym ostatnim gen może przybierać tylko wartości z pewnej grupy wartości dopuszczalnych.

### Sformułowanie problemu

Cel pracy zrealizowano na przykładzie następującego problemu optymalizacyjnego: *Poszukujemy takich wartości wydajności wydobywania ropy naftowej w przyjętych przedziałach czasowych, aby w zadanym całkowitym czasie eksploatacji uzyskać maksymalne wydobycie sumaryczne przy minimalnym wydobywaniu wody.*

Testy działania zbudowanego algorytmu optymalizacyjnego przeprowadzono dla trójfazowego (ropa, woda, gaz) modelu złoża PUNQ-S3 (rysunek 1). Model symulacyjny złoża PUNQ-S3 został udostępniony przez firmę Elf dla celów testowych i jest dostępny na zasadach Open Source. Model składa się z 2660 bloków (siatka bloków  $19 \times 28 \times 5$ ), z czego 1761 bloków jest aktywnych. Opisany niejednorodny model symulacyjny charakteryzuje się następującymi właściwościami petrofizycznymi: (1) średnia porowatość – 14,3%, (2) średnia przepuszczalność pozioma – 278,8 mD, (3) średnia przepuszczalność pionowa – 130,6 mD. W modelu uwzględniono uskoki stanowiący południowo-zachodnią granicę złoża oraz dwa aktywne akifery typu Cartera-Tracy’ego (przepuszczalność – 137,5 mD, porowatość – 21%, promień – 3000 m, miąższość – 19,6 m) okalające złożo od wschodu i południa.

Złożo jest eksploatowane trzema odwiertami strefy gazowej (W-1, W-4, W-12) oraz trzema strefy ropnej (W-5, W-11 i W-15) zlokalizowanymi wokół kontaktu ropa-gaz.



Rys. 1. Złożo PUNQ-S3. Widok 3D

## Schemat działania algorytmu ewolucyjnego

W celu zbudowania efektywnego algorytmu optymalizacyjnego bazującego na metodzie programowania genetycznego konieczne było przyjęcie szeregu założeń dotyczących sposobu jego działania. Algorytm zastosowany w pracy oparto na strategii ewolucyjnej zaproponowanej przez Hollanda w 1975 r., której schemat zapisany w pseudokodzie przedstawiono poniżej. W celu uporządkowania opisu poszczególne elementy schematu oznaczono liczbami arabskimi.

### Algorytm Hollanda

#### begin

(1)  $t = 0$

(2) inicjalizacja  $P^0$

(3) ocena  $P^0$

(4) **while** [not warunek końca] **do**

#### begin

(5)  $T^t$  = reprodukcja  $P^t$

(6)  $O^t$  = krzyżowanie i mutacja  $T^t$

(7) ocena  $O^t$

$$P^{t+1} = O^t$$

$$t = t + 1$$

end

#### end

gdzie:

$P^t$  – populacja bazowa,

$T^t$  – populacja tymczasowa,

$O^t$  – populacja potomna.

Algorytm w przedstawionej postaci przetwarza dwie populacje: populację bazową  $P^t$  oraz populację potomną  $O^t$ . Wykorzystywana jest również populacja tymczasowa  $T^t$  służąca do przechowywania kopii osobników populacji bazowej. W populacjach tych zawarta jest jednakowa liczba osobni-

ków. W kroku 2 populacja bazowa jest wypełniana losowo wygenerowanymi osobnikami. Dla każdego z nich wyznaczana jest w kroku 3 wartość funkcji przystosowania. Po przygotowaniu populacji bazowej uruchamiana jest główna pętla programu realizującego proces sztucznej ewolucji (4). Na początku następuje reprodukcja (5) polegająca na skopowaniu do populacji tymczasowej losowo wybranych osobników z populacji bazowej. Prawdopodobieństwa wylosowania osobników nie są jednak równe – osobniki o większej wartości funkcji przystosowania mają większe szanse reprodukcji; ponadto stosuje się losowanie ze zwracaniem. W konsekwencji w populacji tymczasowej w wyniku reprodukcji znajdzie się większa liczba kopii lepiej przystosowanych osobników. Następnie osobniki z populacji tymczasowej poddawane są operacjom genetycznym: krzyżowaniu i mutacji (6). Są one kojarzone w pary i dla każdej z nich podejmowana jest decyzja o krzyżowaniu. Jeżeli jest ona pozytywna, to następuje krzyżowanie. W przeciwnym wypadku para nie ulega zmianie. Następnie każdy z osobników populacji tymczasowej jest poddawany mutacji jego genotypu. Decyzja o mutacji genu podejmowana jest niezależnie dla każdego genu z zadaniem arbitralnie prawdopodobieństwem. Utworzone w ten sposób osobniki są poddawane ocenie poprzez wyznaczenie wartości funkcji przystosowania (7) i stanowią populację potomną, która w kolejnym przebiegu pętli staje się populacją bazową (4). Pętla jest wykonywana do chwili spełnienia warunku zatrzymania. Może nim być przeprowadzenie określonej liczby iteracji lub znalezienie osobnika o odpowiednio dużej wartości funkcji przystosowania (funkcji celu).

W wersji podstawowej opisany powyżej algorytm posiada dwa parametry: prawdopodobieństwo krzyżowania  $p_c$  oraz prawdopodobieństwo mutacji  $p_m$ .

## Kodowanie

Ponieważ algorytm genetyczny operuje na zakodowanej postaci rozwiązania, konieczne jest określenie sposobu jego reprezentacji. W ogólnym przypadku metody programowania genetycznego do opisu genotypu wykorzystują struktury drzewiaste. Dla naszych celów wystarczająca jest reprezentacja liniowa zdefiniowana jako ciąg liczb całkowitych identyfikujących wartość wydajności wydobywania dla kolejnych odcinków czasowych na podstawie przyjętego alfabetu wartości dopuszczalnych. Na przykład chromosom postaci 1, 3, 4, 2, 5 dla alfabetu wartości dopuszczalnych w postaci 300, 600, 900, 1200, 1500 oznacza, że złożo eksploatuje w kolejnych przedziałach czasowych z wydajnościami 300, 900, 1200, 600, 1500 m<sup>3</sup>/dobę.

Przyjęty sposób kodowania gwarantuje spełnienie poniższych warunków koniecznych do poprawnego działania algorytmu:

- każde rozwiązanie można przedstawić w formie zakodowanej – w przeciwnym wypadku część rozwiązań byłaby z definicji nieosiągalna,
- operacje kodowania i odkodowania są wzajemnie jednoznaczne – w przeciwnej sytuacji możliwe byłoby uzyskanie rozwiązania (genotypu) nieposiadającego interpretacji fizycznej (fenotypu),
- operator kodowania nie wprowadza dodatkowych ekstremów funkcji przystosowania poza tymi, które są odpowiednikami ekstremów rzeczywistego problemu.

### Funkcja przystosowania

Ze względu na przyjęte sformułowanie problemu maksymalizacji w przeprowadzonych testach zbieżności wykorzystano funkcję przystosowania w postaci:

$$F = \frac{f \cdot N}{W} \quad (1)$$

gdzie:  
 $f$  – współczynnik szcerpania złoża,  
 $N$  – sumaryczne wydobycie ropy,  
 $W$  – sumaryczne wydobycie wody.

### Reprodukcja

Operator reprodukcji jest odpowiedzialny za to, które osobniki populacji wejdą do tzw. puli rodzicielskiej, czyli będą podlegać rekombinacji (krzyżowaniu i mutacji), w wyniku czego powstanie pokolenie potomne. W niniejszej pracy zastosowano reprodukcję rangową, dla której prawdopodobieństwo reprodukcji każdego osobnika jest podane w sposób jawny na podstawie jego rangi – liczby charakteryzującej jakość osobnika na tle populacji. Jest ona sortowana według niemalejącej wartości funkcji przystosowania; ranga jest numerem osobnika w takim

uszeregowaniu. Po określeniu rang definiuje się zmienną losową, przypisując każdemu osobnikowi prawdopodobieństwo reprodukcji na podstawie jego rangi zgodnie z poniższą formułą:

$$p_{r,k} = \frac{r(X_k)}{\sum_1^N r(X_i)} \quad (2)$$

gdzie:  
 $p_{r,k}$  – prawdopodobieństwo reprodukcji  $k$ -tego osobnika,  
 $r(X_k)$  – ranga  $k$ -tego osobnika w populacji.

### Rekombinacja

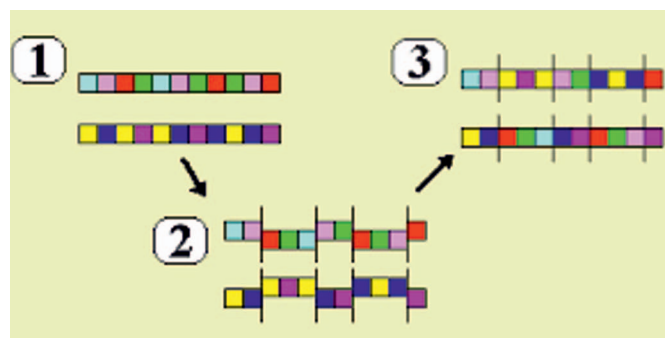
Wybór operatorów ewolucyjnych oraz parametrów ich działania ma podstawowe znaczenie dla efektywności pracy całego algorytmu. Trudność polega na tym, że nie istnieją operatory bezwzględnie „lepsze” i „gorsze”. Oznacza to, że przyjęty sposób kodowania, który wraz z rozbudowanym operatorem selekcji, krzyżowania i mutacji wykazuje bardzo dobrą zbieżność dla np. problemu „komiwojażera”, może być zupełnie bezużyteczny w przypadku kalibracji symulacyjnego modelu złożowego. Specyfika rozwiązywanego problemu i wynikająca z niej ograniczona możliwość wykorzystania istniejących w literaturze rozwiązań [2, 15], wraz z ilością kombinacji możliwych do zastosowania operatorów i wartości parametrów liczbowych, stanowią w tej sytuacji o złożoności problemu.

Zastosowane w pracy operatory zostały omówione poniżej.

**Krzyżowanie** to proces „mieszania” fragmentów genotypów osobników rodzicielskich w celu wykreowania lepiej przystosowanych osobników potomnych z prawdopodobieństwem  $p_c$ . W pracy zastosowano metodę krzyżowania  $n$ -punktowego. Polega ona na wygenerowaniu losowo  $n$  punktów z zakresu od 1 do liczby genów w chromosomie. W efekcie, na podstawie wygenerowanych punktów, każdy

chromosom jest podzielony na  $n + 1$  fragmentów. Krzyżowanie polega na budowaniu dwóch osobników potomnych przez krzyżowanie dwóch osobników rodzicielskich w wyniku przemiennego kopiowania fragmentów chromosomu z osobników rodzicielskich (rysunek 2).

**Mutacja** polega na perturbacji genotypu jednego osobnika, przy czym zmiana ta następuje z określonym prawdopodobieństwem  $p_m$ . Decyzja o mutacji podejmowana jest dla każdego genu każdego osobnika populacji potomnej. Przyjmuje się ponadto, że niewielkie perturbacje są bardziej prawdopodobne niż duże.



Rys. 2. Zasada krzyżowania  $n$ -punktowego

### Przykłady optymalizacji

#### Adaptacja parametrów

Jednym z głównych problemów wynikających ze stosowania algorytmów genetycznych, a w szczególności metod

programowania genetycznego, jest występowanie „przedwczesnej zbieżności” (będącej niczym innym jak szybkim osiągnięciem przez algorytm pewnego ekstremum) i „zablo-

kowania” dalszego procesu optymalizacji. W odróżnieniu od metod klasycznych algorytmy genetyczne posiadają mechanizmy zapobiegania tego rodzaju sytuacjom. Prawie zawsze stosowana jest mutacja umożliwiająca wprowadzenie losowego, ale kontrolowanego zaburzenia procesu krzyżowania. Drugim wykorzystywanym często zabiegiem jest adaptacja prawdopodobieństw krzyżowania  $p_c$  i mutacji  $p_m$ . Polega ona na tym, że w każdym pokoleniu  $p_c$  i  $p_m$  są wyznaczane od nowa, w zależności od wartości wybranego parametru charakteryzującego populację.

W prezentowanym algorytmie zastosowano adaptację sterowaną współczynnikiem zróżnicowania  $Z_d$ :

$$Z_d = 1 - \frac{f_{min}(X)}{f_{mean}(X)} \quad (3)$$

W powyższej formule przez  $f_{min}(X)$  oraz  $f_{mean}(X)$  oznaczono odpowiednio najgorsze i średnie przystosowanie w populacji. Parametr ten jest tym większy, im większe jest zróżnicowanie osobników. W granicznym przypadku, gdy populacja składa się z identycznych osobników,  $Z = 0$ .

## Wyniki symulacji

Zbudowany algorytm ewolucyjny zaimplementowano w kompilatorze Parallel Studio XE 2011 firmy Intel. Przy wyznaczaniu wartości funkcji przystosowania użyto symulatora złożowego ECLIPSE 100 firmy Schlumberger.

Dla każdego z przedstawionych poniżej przykładów optymalizacji przyjęto:

- maksymalna liczba generacji: 35,
- liczba osobników w pokoleniu bazowym: 10.

Ponadto dla prawdopodobieństwa krzyżowania  $p_c$  oraz mutacji  $p_m$  przyjęto następujące wartości graniczne:

$$p_{c\ min} = 0,5; p_{c\ max} = 0,9$$

$$p_{m\ min} = 0,01; p_{m\ max} = 0,3$$

oraz formuły opisujące ich zmienność (3–5).

W trakcie przeprowadzonych testów wykonywano 30-letnie prognozy wydobywania, maksymalizując sumaryczne wydobywanie ropy naftowej ze złoża zgodnie z formułą (1).

Testy prowadzono dla trzech różnych długości chromosomu dla krzyżowania 1-punktowego (test – 1):

- długość chromosomu: 5 (5 przedziałów czasowych po 6 lat każdy),
- długość chromosomu: 10 (10 przedziałów czasowych po 3 lata każdy),
- długość chromosomu: 15 (15 przedziałów czasowych po 2 lata każdy),

oraz dla 1, 2 i 3 punktów krzyżowania dla długości chromosomu równej 10 (test 2).

Dla każdego pokolenia prawdopodobieństwa krzyżowania  $p_c$  i mutacji  $p_m$  wyznaczane były według następujących formuł:

$$p_c = p_{c\ min} + Z_d \cdot (p_{c\ max} - p_{c\ min}) \quad (4)$$

$$p_m = p_{m\ max} - Z_d \cdot (p_{m\ max} - p_{m\ min}) \quad (5)$$

gdzie indeksy *max*, *min* odnoszą się do zadanych maksymalnych i minimalnych wartości parametrów.

Działanie tak zdefiniowanego mechanizmu adaptacji jest następujące:

- jeżeli zróżnicowanie populacji rośnie ( $Z \rightarrow 1$ ), automatycznie zwiększane jest prawdopodobieństwo krzyżowania, a zmniejszane prawdopodobieństwo mutacji,
- jeżeli zróżnicowanie populacji zmniejsza się ( $Z \rightarrow 0$ ), mechanizm działa odwrotnie, tzn. zmniejsza prawdopodobieństwo krzyżowania, a zwiększa prawdopodobieństwo mutacji.

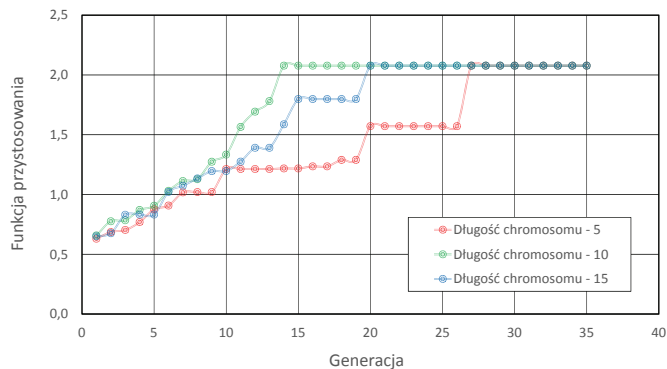
Przedstawione rozwiązanie z jednej strony gwarantuje różnorodność populacji, z drugiej zaś zapewnia dobrą eksploatację przestrzeni rozwiązań.

Zbiór dozwolonych wydajności wydobywania ze złoża składał się z 6 wybranych arbitralnie wartości: 300, 600, 900, 1200, 1500, 1800 m<sup>3</sup>/dobę.

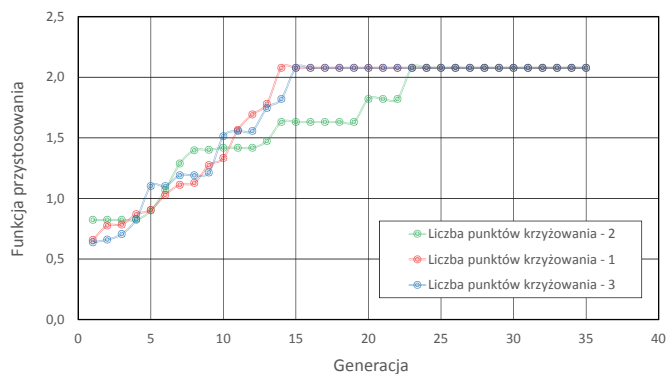
Zadaniem zbudowanego algorytmu był taki dobór wydajności wydobywania w poszczególnych przedziałach czasowych, aby uzyskać maksymalne wydobywanie ropy przy minimalnym wydobywaniu wody. Na podstawie wyników przeprowadzonych symulacji przeanalizowano wybrane aspekty jego działania.

Rysunek 3 przedstawia zależność wyników optymalizacji od długości chromosomu dla 3 wybranych jej wartości (5, 10, 15). W praktyce oznaczało to podziały 30-letniego okresu prognoz na 6-, 3- i 2-letnie okresy, w których wydajność wydobywania była stała. Testy w tym przypadku prowadzono dla krzyżowania 1-punktowego. Dla każdego z przedstawionych przypadków uzyskano identyczną wartość funkcji przystosowania, przy czym nie stwierdzono jednoznacznej zależności szybkości procesu optymalizacji od długości chromosomu. Charakterystyczne jest to, że spośród trzech długości chromosomu najgorszą zbieżność otrzymano dla najkrótszego chromosomu odpowiadającego najdłuższemu, czyli 6-letnim przedziałom czasowym.

Na rysunku 4 przedstawiono zależność zbieżności algorytmu od liczby punktów krzyżowania dla krzyżowania 1-, 2- i 3-punktowego (test 2). Podobnie jak dla testu zależności zbieżności od długości chromosomu (rysunek 3) nie stwierdzono zależności od badanego parametru. W każdym z przy-



Rys. 3. Zależność zbieżności algorytmu od długości chromosomu



Rys. 4. Zależność zbieżności algorytmu od liczby punktów krzyżowania

padków (1, 2, 3 punkty krzyżowania) uzyskano identyczne maksimum, zgodne z wynikiem testu 1.

Należy zauważyć, że brak widocznej zależności pomiędzy zbieżnością a badanymi parametrami (długość chromosomu, liczba punktów krzyżowania) wynika ze stochastycznego charakteru zaproponowanej metody optymalizacji.

Zadaniem artykułu było zweryfikowanie możliwości zastosowania metod programowania genetycznego dla celów optymalizacji wydobycia ropy naftowej przy założeniu minimalizacji wydobycia wody złożowej. W wyniku wykonanych symulacji potwierdzono możliwość zastosowania wybranej metody do optymalizacji wydobycia węglowodorów przy założonych warunkach ograniczających.

Przeprowadzone analizy symulacyjne pozwalają sformułować następujące wnioski i uwagi:

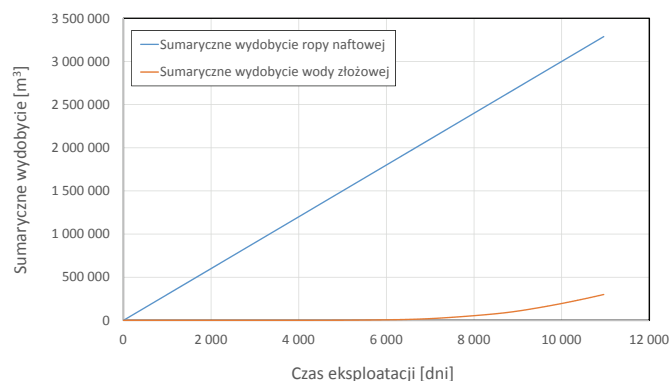
1. Zastosowana metoda okazała się być efektywna dla rozwiązania postawionego problemu optymalizacyjnego. Przeprowadzone testy wykazały zadowalającą zbieżność zarówno dla różnych długości chromosomu, jak i różnej liczby punktów krzyżowania. Co więcej dla każdego z przeprowadzonych testów otrzymano w każdym z ana-

lizowanych wariantów identyczne rozwiązanie, co świadczy o bardzo dobrej powtarzalności działania algorytmu.

2. Przyjęcie funkcji przystosowania w postaci 1 powoduje preferowanie przez algorytm optymalizacyjny niskich wydajności wydobycia ropy ze względu na szybki przyrost wydobycia wody w przypadku zwiększania wydobycia ropy.

3. Operator mutacji ma kluczowe znaczenie dla efektywnego przeszukiwania przestrzeni rozwiązań. Przeprowadzone symulacje wskazują, że zastosowanie prawdopodobieństwa mutacji na zbyt wysokim poziomie powoduje problemy z utrzymaniem zbieżności algorytmu.

4. Wskazane jest stosowanie adaptacji parametrów sterujących działaniem algorytmu. Wynika to z potrzeby różnicowania osobników w zależności od stanu procesu optymalizacji.



Rys. 5. Sumaryczne wydobycie płynów złożowych

### Podsumowanie

Prosimy cytować jako: Nafta-Gaz 2017, nr 10, s. 760–767, DOI: 10.18668/NG.2017.10.05

Artykuł nadesłano do Redakcji 23.12.2016 r. Zatwierdzono do druku 18.07.2017 r.

Artykuł powstał na podstawie pracy statutowej pt.: *Zastosowanie metod programowania genetycznego w procesie maksymalizacji wydobycia węglowodorów przy zastosowaniu symulatora złożowego* – praca INiG – PIB na zlecenie MNiSW; nr zlecenia: 27/KZ/16, nr archiwalny: DK-4100-27/16.

## Literatura

- [1] Abraham J.N., Rao K.S.: *Integration of generative and evaluative models for production scheduling of lube oil plants in a petroleum refinery*. Applied Mathematical Modelling 2009, vol. 33, s. 1213–1227.
- [2] Aissani N., Beldjilali B., Trentesaux D.: *Dynamic scheduling of maintenance tasks in the petroleum industry: A reinforcement approach*. Engineering Applications of Artificial Intelligence 2009, vol. 22, s. 1089–1103.
- [3] Arabas J.: *Wykłady z algorytmów ewolucyjnych*. Wydanie drugie. Wydawnictwo Naukowo-Techniczne, Warszawa 2004.
- [4] Aytak A., Kisi O.: *A genetic programming approach to suspended sediment modelling*. Journal of Hydrology 2008, vol. 351, s. 288–298.
- [5] Banzhaf W.: *Evolutionary Computation and Genetic Programming*. Engineered Biomimicry 2013, Chapter 17, s. 429–447.
- [6] Can B., Heavey C.: *A comparison of genetic programming and artificial neural network sin metamodeling of discrete-event simulation models*. Computers & Operations Research 2012, vol. 39, s. 424–436.
- [7] Chen Y., Ohkawa E., Mabu S., Shimada K., Hirasawa K.: *A portfolio optimization model using Genetic Network Programming with control nodes*. Expert Systems with Applications 2009, vol. 36, s. 10735–10745.
- [8] Dimopoulos C., Zalzal A.M.S.: *Investigating the use of genetic programming for a classic one-machine scheduling problem*. Advances in Engineering Software 2001, vol. 32, s. 489–498.
- [9] Fathinasab M., Ayatollahi S.: *On the determination of CO<sub>2</sub>-crude oil minimum miscibility pressure using genetic programming combined with constrained multivariable search methods*. Fuel 2016, vol. 173, s. 180–188.
- [10] Gao F., Leea T., Caoa W., Leea X., Denga Y., Tonga H.: *Self-evolution of hyperfractional order chaos driven by a novel approach through genetic programming*. Expert Systems with Applications 2016, vol. 52, s. 1–15.
- [11] Garcia-Arnau M., Manrique D., Rios J., Rodriguez-Paton A.: *Initialization method for grammar-guided genetic programming*. Knowledge-Based Systems 2007, vol. 20, s. 127–133.
- [12] Gharbi R.B.C., Mansoori G.A.: *An introduction to artificial intelligence applications in petroleum exploration and production*. Journal of Petroleum Science and Engineering 2005, vol. 49, s. 93–96.
- [13] Gwiazda T.D.: *Algorytmy genetyczne – kompendium*. Tom I: *Operator krzyżowania dla problemów numerycznych*. Wydawnictwo Naukowe PWN, Warszawa 2007.
- [14] Gwiazda T.D.: *Algorytmy genetyczne – kompendium*. Tom II: *Operator mutacji dla problemów numerycznych*. Wydawnictwo Naukowe PWN, Warszawa 2007.
- [15] Hasan A.: *Optimal Control of Petroleum Reservoirs*. 1<sup>st</sup> IFAC Workshop on Control of Systems Governed by Partial Differential Equations 2013, s. 144–149.
- [16] Hasan A., Foss B.: *Optimal switching time control of petroleum reservoirs*. Journal of Petroleum Science and Engineering 2015, vol. 131, s. 131–137.
- [17] Hashimoto N., Kondo N., Hatanaka T., Uosaki K.: *Nonlinear System Modeling by Hybrid Genetic Programming*. Proceedings of the 17<sup>th</sup> World Congress the International Federation of Automatic Control 2008, s. 4906–4611.
- [18] Holland J.H.: *Adaptation in Natural And Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. University of Michigan Press, Ann Arbor, 1975.
- [19] Jakobovic D., Marasovic K.: *Evolving priority scheduling heuristics with genetic programming*. Applied Soft Computing 2012, vol. 12, s. 2781–2789.
- [20] Kashid S.S., Maity R.: *Prediction of monthly rainfall on homogeneous monsoon regions of India based on large scale circulation patterns using Genetic Programming*. Journal of Hydrology 2012, vol. 454, s. 26–41.
- [21] Kaydani H., Mohebbi A., Hajizadeh A.: *Dew point pressure model for gas condensate reservoirs based on multi-gene genetic programming approach*. Applied Soft Computing 2016, vol. 47, s. 168–178.
- [22] Koza J.R., Poll R.: *Genetic Programming*. Chapter 5, MIT Press edition.
- [23] Langdon W.B., Poli R., McPhee N.F., Koza J.R.: *Genetic Programming: An Introduction and Tutorial, with a Survey of Techniques and Applications*. Springer-Verlag Berlin Heidelberg 2008, vol. 115, s. 927–1028.
- [24] Łętkowski P.: *Zastosowanie algorytmu optymalizacji wieloagentowej do automatycznej kalibracji modeli symulacyjnych*. Nafta-Gaz 2016, nr 9, s. 704–712, DOI: 10.18668/NG.2016.09.04.
- [25] Łętkowski P.: *Zastosowanie hybrydowej metody optymalizacji rojem cząstek w procesie automatycznej kalibracji modeli złożowych*. Nafta-Gaz 2014, nr 11, s. 784–793.
- [26] Meiyi W., Xiang L., Lean Y.: *Time-dependent fuzzy random location-scheduling programming for hazardous materials transportation*. Transportation Research Part C 2015, vol. 57, s. 146–165.
- [27] Qin W., Zhang J., Sun Y.: *Multiple-objective scheduling for interbay AMHS by using genetic-programming-based composite dispatching rules generator*. Computers in Industry 2013, vol. 64, s. 694–707.
- [28] Reisi-Nafchi M., Moslehi G.: *A hybrid genetic and linear programming algorithm for two-agent order acceptance and scheduling problem*. Applied Soft Computing 2015, vol. 33, s. 37–47.
- [29] Russo M.: *A distributed neuro-genetic programming tool*. Swarm and Evolutionary Computation 2016, vol. 27, s. 145–155.
- [30] Sato H., Hasegawa Y., Bollegala D., Iba H.: *Improved sampling using loopy belief propagation for probabilistic model building genetic programming*. Swarm and Evolutionary Computation 2015, vol. 23, s. 1–10.
- [31] Tang Y., Liu R., Sun Q.: *Schedule control model for linear projects based on linear scheduling method and constraint programming*. Automation in Construction 2014, vol. 37, s. 22–37.
- [32] Uy N.Q., Hoai N.X., O'Neill M., McKay R.I., Phong D.N.: *On the roles of semantic locality of crossover in genetic programming*. Information Sciences 2013, vol. 235, s. 195–213.
- [33] Xie H., Zhang M.: *Impacts of sampling strategies in tournament selection for genetic programming*. Soft Comput 2012, vol. 16, s. 615–633.
- [34] Yeun Y.S., Yang Y.S., Ruy W.S., Kim B.J.: *Polynomial genetic programming for response surface modeling. Part 1: a methodology*. Struct Multidisc Optim 2005, vol. 29, s. 19–34.
- [35] Zhang Y., Bhattacharyya S.: *Genetic programming in classifying large-scale data: an ensemble method*. Information Sciences 2004, vol. 163, s. 85–101.



Dr inż. Piotr ŁĘTKOWSKI  
 Adiunkt w Zakładzie Symulacji Złóż Węglowodorów  
 i Podziemnych Magazynów Gazu.  
 Instytut Nafty i Gazu – Państwowy Instytut Badawczy  
 ul. Lubicz 25 A  
 31-503 Kraków  
 E-mail: piotr.letkowski@inig.pl