

Jarosław M. SZYMAŃDA*

WARUNKI BRZEGOWE W ALGORYTMACH EWOLUCYJNYCH - PROPOZYCJE DYDAKTYCZNE

W referacie przedstawiono propozycję skojarzenia warunków brzegowych dla wybranego zagadnienia poszukiwania rozwiązania układu równań różniczkowych eliptycznych, z parametrami symulacji środowiska oddziałującego na wybrane cechy osobników w algorytmach ewolucyjnych. W szczególności wykorzystano powiązanie warunków brzegowych oraz wartości rozwiązania zagadnienia Dirichleta z parametryzacją wybranych cech osobników realizujących zadanie przeszukiwania obszaru $D(X)$ zgodnie z określoną funkcją celu identyfikacji obiektów. Zagadnienie prezentowane jest w kontekście możliwych implementacji dydaktycznych dla studentów wydziałów elektrycznych na kierunkach elektrotechnika, automatyka i robotyka oraz mechatronika.

SŁOWA KLUCZOWE: algorytmy ewolucyjne, programowanie, dydaktyka

1. ASPEKT DYDAKTYCZNY

Umiejętność implementacji algorytmów w zakresie programowania podstawowego jest jednym z ważniejszych elementów kształcenia problemowego. Pod pojęciem programowania podstawowego przyjęto nabytą umiejętność programowania niezależnie od poziomu semantycznego technik programowania, innymi słowy poziom zaawansowania projektów uwzględnia indywidualne możliwości studentów. W kontekście realizacji dydaktycznych, referat częściowo nawiązuje do artykułu autora publikowanego w materiałach ubiegłorocznej edycji konferencji ZKWE'2015 „*Modelowanie symulacyjne z pakietem dydaktycznym Si-STlab20*” [3]. W przywołanym artykule przybliżone zostały między innymi najważniejsze postulaty kształcenia problemowego oraz zasady pracy w zespołach projektowych.

2. ALGORYTMY KLASYCZNE I EWOLUCYJNE

Poszukiwanie wariantywnych metod rozwiązywania zagadnień z zakresu optymalizacji zawsze było przedmiotem zainteresowań nie tylko matematyków, ale także inżynierów starających się zastosować takie metody w praktyce. Procesy

*Politechnika Wroclawska.

optymalizacji polegają na przeszukiwaniu przestrzeni potencjalnych rozwiązań danego problemu w celu znalezienia najlepszego rozwiązania.

Klasyczny (*deterministyczny*) problem optymalizacji określany jest jako procedura poszukiwania wartości zmiennej x , zawartej w danym zbiorze X , przy której dana funkcja $f(x)$ przyjmuje najkorzystniejszą wartość. W takim przypadku funkcja $f(x)$, nazywana jest funkcją celu bądź wskaźnikiem lub kryterium jakości, innymi słowy mierzy cel, jaki ma być osiągnięty. W praktyce często występuje wiele rozbieżnych celów. Zbiór X jest wyznaczony przez zbiór ograniczeń danego problemu. Większość klasycznych algorytmów optymalizacyjnych stosuje deterministyczną procedurę, która w sposób systematyczny, punkt po punkcie zbliża się do rozwiązania optymalnego. Procedura taka zwykle zaczyna poszukiwanie rozwiązania optymalnego startując z pewnego wybranego rozwiązania, po czym, na podstawie informacji lokalnych (*środowiska*), określany jest kierunek poszukiwań. W cyklu iteracyjnym każde odszukane rozwiązanie staje się warunkiem początkowym (*startowym*) kolejnego powtórzenia procedury optymalizacji dopóty, dopóki nie zostanie spełnione kryterium zakończenia powtórzeń. Algorytmy klasyczne różnią się między sobą głównie sposobem określania kierunku poszukiwań, przykładem mogą być metody bezpośrednie i gradientowe. Metody bezpośrednie przy wyznaczaniu kierunku poszukiwań wykorzystują tylko wartości funkcji celu i ograniczeń. Metody gradientowe posługują się pojęciem pierwszej i drugiej pochodnej funkcji celu lub zbioru ograniczeń.

W rzeczywistych zastosowaniach przestrzeni potencjalnych rozwiązań jest bardzo często tak duża, że nie jest możliwe w *dostatecznie krótkim* czasie sprawdzenie wszystkich możliwych rozwiązań w celu wybrania najlepszego optymalnego rozwiązania. W takich przypadkach uzasadnione staje się zastosowanie technik probabilistycznych (*niedeterministycznych*), które używają wyboru losowego jako narzędzia do ukierunkowania procesu poszukiwań. Algorytmy ewolucyjne są właśnie jedną z takich technik, stosowaną z powodzeniem w praktyce inżynierskiej. W bardzo wielu publikacjach naukowych porównuje się algorytmy ewolucyjne ze zbiorem metod optymalizacji inspirowanych analogiami biologicznymi [1,2]. Z analogii biologicznych przejęto do teorii algorytmów ewolucyjnych także pewne określenia, które ułatwiają bardziej sformalizowane ujęcie problemu optymalizacji. Do terminologii tej zaliczamy między innymi następujące pojęcia:

- **Osobnik**: podstawowa jednostka podlegająca ewolucji. Każdy osobnik jest propozycją rozwiązania postawionego problemu. Zakładamy zwykle, że dany osobnik przebywa w pewnym środowisku (*warunki brzegowe ewolucji*), do którego może być *lepiej* lub *gorzej* przystosowany. W środowisku każdemu osobnikowi *często* przypisywana jest wartość liczbowa, określająca jakość reprezentowanego przez niego rozwiązania. *Celem* ewolucji jest stworzenie osobnika możliwie dobrze przystosowanego do danego środowiska.
- **Populacja osobników**: zbiór (*grupa, zespół*) osobników „konkurujących” o zasoby wspólnego środowiska.

- **Genotyp**: kompletny i jednoznaczny opis osobnika zawarty w jego *genach*. W przypadku algorytmów ewolucyjnych cechy rozwiązania (z przestrzeni poszukiwanych rozwiązań) kodowane są w określony sposób, np. za pomocą ciągów binarnych ustalonej długości (odpowiednikiem genotypu osobnika jest w tym przypadku ciąg bitów).
- **Fenotyp**: cechy danego osobnika, utworzone podług genotypu (inaczej: zestaw cech określanych przez genotyp lub kodowanie fenotypu przez genotyp). Cechy podlegają ocenie *środowiska*; wartość liczbowa tej oceny nazywana jest *przystosowaniem osobnika*.
- **Chromosom**: miejsce przechowywania genotypu osobnika. Najczęściej jeden osobnik zawiera jeden chromosom.
- **Środowisko**: wzajemne oddziaływanie *warunków brzegowych ewolucji*¹ z osobnikiem poprzez funkcję przystosowania, za pomocą której osobnikowi przypisuje się przystosowanie do warunków na podstawie *fenotypu*. W niektórych publikacjach funkcja przystosowania osobnika nazywana jest *krajobrazem adaptacyjnym* [1, 2].
- **Mutacja**: operator genetyczny polegający na niezależnej perturbacji genotypu każdego osobnika w celu zwiększenia różnorodności osobników w populacji. Takie działanie ma zapobiegać przedwczesnej zbieżności algorytmu do lokalnego optimum.
- **Krzyżowanie**: operator genetyczny łączący cechy różnych osobników w danej populacji. Najczęściej operacja ta wykonywana jest na parze osobników nazywanych w takim przypadku *rodzicami* lub *osobnikami rodzicielskimi*.

Działanie algorytmu ewolucyjnego sprowadza się do następującej procedury: algorytm ewolucyjny rozpoczyna proces przeszukiwania od utworzenia populacji potencjalnych rozwiązań nazywanych osobnikami. Każdy osobnik reprezentowany jest przez swój *genotyp*. W każdym kroku jego *fenotyp* jest oceniany zgodnie z *funkcją przystosowania*, a następnie przeprowadzana jest selekcja w celu eliminacji osobników ocenionych jako najgorsze. Osobniki wykazujące wysokie przystosowanie podlegają *mutacji* oraz *krzyżowaniu*, w szczególności rekombinacji podlegają *genotypy* tych osobników. Dzięki krzyżowaniu, ewolucyjny proces może się przybliżać w kierunku oczekiwanych obszarów w przestrzeni poszukiwanych rozwiązań. Mutacja natomiast zmniejsza prawdopodobieństwo zbieżności do lokalnego optimum. W wyniku działania operatorów krzyżowania i mutacji tworzone są nowe rozwiązania, z których budowana jest populacja następnej generacji. Warunkiem zakończenia algorytmu jest osiągnięcie zadawalającego poziomu przystosowania. Należy jednak zaznaczyć, że współczesna postać elementów składowych algorytmów ewolucyjnych ma już niewiele wspólnego ze swoimi biologicznymi odpowiednikami [2].

¹ Termin wprowadzony przez autora referatu w celu możliwości rozdzielenia warunków zewnętrznych środowiska od warunków przystosowania się osobnika do tego środowiska

3. WARUNKI BRZEGOWE ZAGADNIENIA DIRICHLETA

Przedmiotem referatu jest przedstawienie propozycji skojarzenia parametrów kształtujących i opisujących środowisko z elementami algorytmu ewolucyjnego. Dla ustalenia uwagi przyjęto, że środowisko zewnętrzne jako zbiór wartości wejściowych do algorytmu ewolucyjnego, będzie kształtowane poprzez wartości otrzymywane podczas rozwiązywania wewnętrznego zagadnienia Dirichleta. Wewnętrzne zagadnienie Dirichleta, nazywane także pierwszym zagadnieniem brzegowym, polega na poszukiwaniu funkcji spełniającej wewnątrz danego obszaru równanie eliptyczne Laplace'a lub Poissona. Tak jak przy rozwiązywaniu każdego równania różniczkowego zawsze muszą być określone właściwe warunki brzegowe i początkowe, tak i w tym przypadku jako warunki brzegowe przyjmowane są wartości funkcji której rozwiązania poszukujemy. Na potrzeby przedstawienia problemu procedura rozwiązywania równania Laplace'a określonego zależnością: $\Delta\varphi(x,y,z) = 0$ została dołączona do procedur numerycznych jako jeden z elementów algorytmu ewolucyjnego.

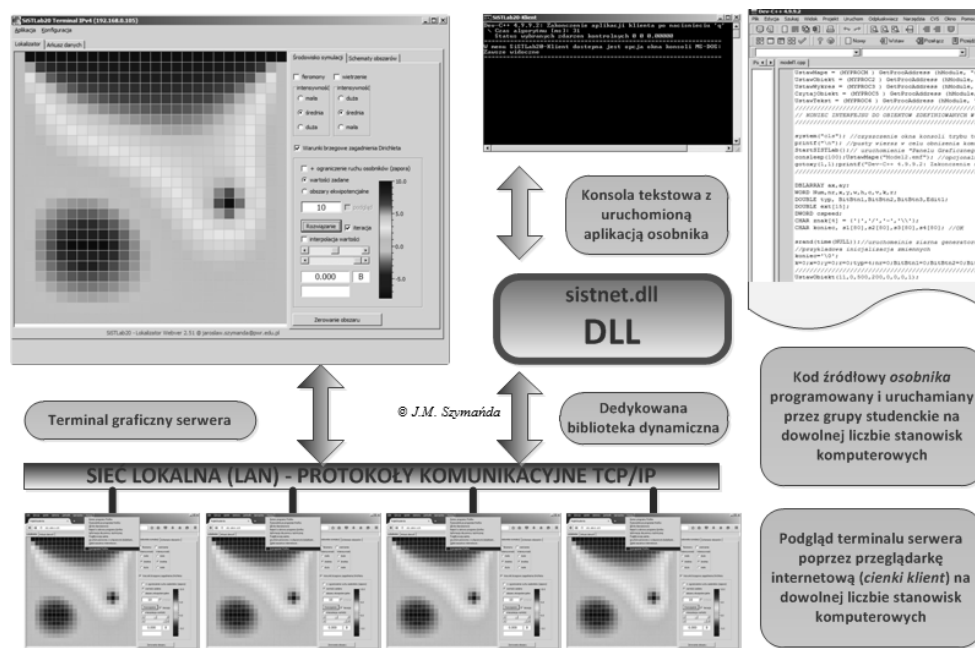
4. WARUNKI BRZEGOWE ALGORYTMU EWOLUCYJNEGO

W algorytmach ewolucyjnych, *środowisko* jest podstawowym obszarem wzajemnych oddziaływań tego środowiska z osobnikiem poprzez *funkcję przystosowania*. Aktywność interakcji w tym zakresie oceniana jest poprzez jakość *fenotypu*. Selekcja wyklucza osobniki, których *genotyp* nie wytworzył odpowiednich cech spełniających założone kryteria. Przedstawiony w referacie model (schemat) warunków brzegowych algorytmu ewolucyjnego obejmuje warunki dwojakiego rodzaju. Pierwsze, to wartości rozwiązania zagadnienia Dirichleta, przybliżone w punkcie 3, drugie to uwarunkowania *logistyczne* określające możliwości przeszukiwania obszaru pożądaných rozwiązań. W algorytmie wprowadzono ograniczenia dla ruchu osobników (zapory). Stanowią one swoisty podzbiór ograniczeń dla danego problemu. Połączenie odrębnych w ocenie funkcji przystosowania osobników do danego środowiska było zadaniem wymagającym zastosowania złożonych algorytmów decyzyjnych, w tym wykorzystania niektórych elementów z teorii algorytmów heurystycznych, statystyki kwalitatywnej, optymalizacji dyskretnej i innych. Ze względu na przeglądowy charakter referatu i ograniczenia redakcyjne, szersze przedstawienie przyjętych rozwiązań oraz ujęcie matematyczne można odszukać w publikacjach [3, 4].

5. PROJEKT NUMERYCZNY

Podczas przygotowywania założeń projektowych przyjęto następujące ogólne wytyczne jego realizacji:

- algorytm ewolucyjny ma wykorzystywać rozproszony sieciowy system przetwarzania danych,
- podstawowe elementy algorytmu ewolucyjnego, takie jak *osobniki*, powinny przyjąć ze względu na aspekt zastosowań dydaktycznych formę umożliwiającą ich pogładową wizualizację,
- *osobniki populacji* powinny mieć zaimplementowaną możliwość komunikacji wzajemnej oraz detekcji zdarzeń podczas interakcji ze *środowiskiem*,
- *osobniki* nie mogą być zaprogramowane bezpośrednio w aplikacji *środowiska*,
- komunikacja *osobników ze środowiskiem* i innymi *osobnikami* może odbywać się wyłącznie w trybie komunikacji sieciowej *klient-serwer* poprzez kwerendy,
- identyfikacja przez *osobniki* parametrów *środowiska* oraz innych zdarzeń nie może być realizowana w sposób niezgodny z przyjętymi zasadami komunikacji i wymiany informacji.



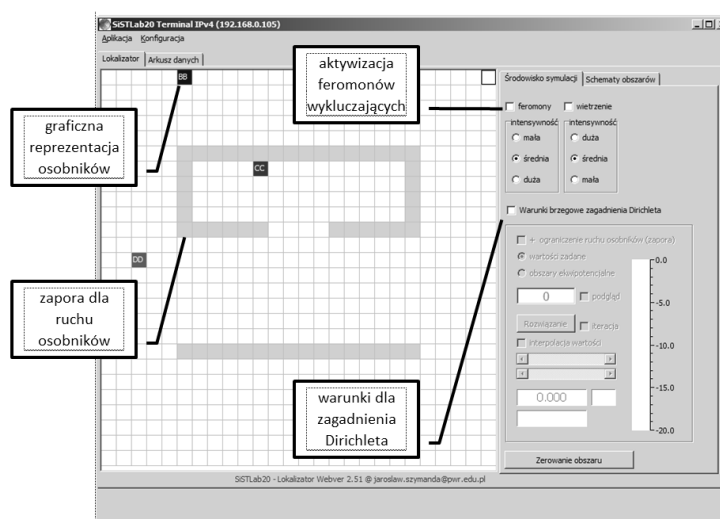
Rys. 1. Poglądowy diagram komunikacji *osobniki-środowisko (klient-serwer)*

Na rysunku 1 przedstawiono poglądowy diagram wymiany informacji pomiędzy *środowiskiem* algorytmu ewolucyjnego a *populacją osobników*. Przyjęte zasady komunikacji umożliwiają na wzajemną interakcję osobników poprzez identyfikację wszystkich zdarzeń występujących w środowisku. Cechy i zachowanie każdego osobnika definiowane są w niezależnym algorytmie przy czym dopuszcza się możliwość grupowania osobników ze względu na przynależność do tej samej ge-

neracji, posiadających zbliżone cechy (*fenotypy*) ocenione w podobnym zakresie wartości itp. Wymiana informacji w sieci LAN realizowana jest poprzez odpowiednio zdefiniowane kwerendy i procedury, które zamieszczono w dedykowanej bibliotece pakietu aplikacji. W kontekście zastosowań dydaktycznych opracowany system umożliwia poszczególnym grupom studenckim na podgląd zachodzących w środowisku zdarzeń, w tym obserwację „własnych” *osobników* realizujących zaprogramowane zadania. Obserwacja *środowiska* dostępna jest poprzez dowolną przeglądarkę internetową na dowolnej liczbie stanowisk komputerowych.

5.1. Edytor środowiska populacji osobników

W wariantcie podstawowym, obszar poszukiwania rozwiązań optymalnych formowany jest przy wykorzystaniu warunków *logistycznych* modelujących ograniczenia ruchu osobników.

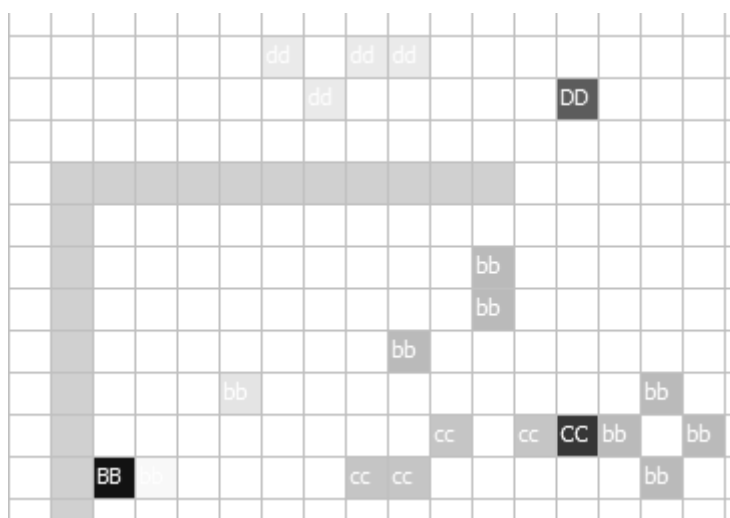


Rys. 2. Terminal graficzny *środowiska* algorytmu ewolucyjnego

Ograniczenia tego rodzaju wprowadzane są poprzez edytor graficzny. Przykładowe *zapory* przedstawiono na rysunkach 2 i 3. Do parametrów definiujących *środowisko* dołączono także warunki modyfikujące zachowanie osobników w czasie działania algorytmu ewolucyjnego. Aktualna implementacja takiego wariantu obejmuje możliwość aktywacji i dezaktywacji tzw. *feromonów wykluczających*². Feromony wykluczające są oznaczane przez każdego osobnika indywidual-

² Termin wprowadzony przez autora referatu w celu możliwości określania dynamicznej modyfikacji obszaru poszukiwania rozwiązań optymalnych

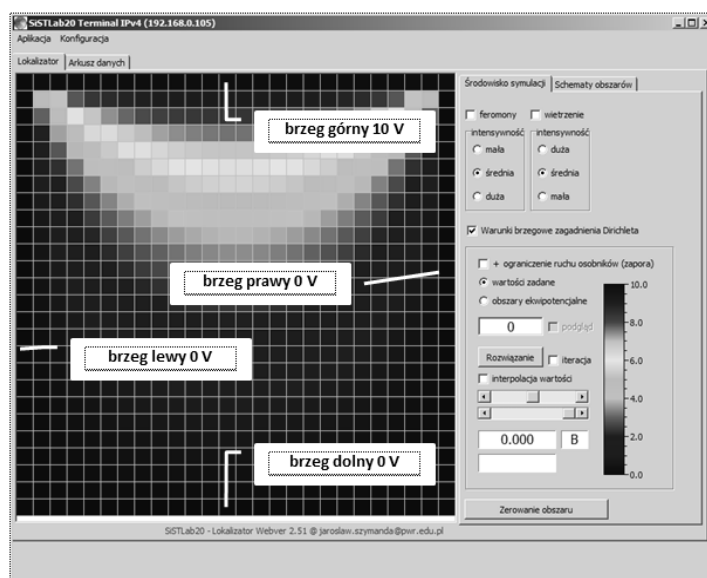
nie i podobnie jak *zapory* ograniczają przestrzeń poszukiwanych rozwiązań (*ruchu*) przez dane osobniki. Swoista dynamiczna blokada ograniczająca swobodę poszukiwania rozwiązań, zdejmowana jest automatycznie po określonym dla *środowiska* czasie wietrzenia feromonów. Czas wietrzenia ściśle powiązany jest z indywidualnie oznaczonym i pozostawionym feromonem danego *osobnika*. Doceľowo przewiduje się możliwość aktywacji i dezaktywacji feromonów bezpośrednio poprzez zapisanie takiej cechy w *genotypie* osobnika. Przykładowy fragment podglądu *środowiska* z „produkowanymi” feromonami wykluczającymi przedstawiono na rysunku 3.



Rys. 3. Fragment podglądu środowiska z zaporą i feromonami wykluczającymi. Duże litery identyfikują *osobniki*, małe pozostawiane przez osobnika feromony

W punkcie 4 referatu przedstawiono możliwość powiązania warunków *logistycznych* (*ograniczenia ruchu osobników*) z zewnętrznymi warunkami brzegowymi rozwiązania równania Laplace'a a w szczególności rozwiązania zagadnienia Dirichleta. W opracowanym edytorze graficznym *środowiska* dla algorytmu ewolucyjnego wprowadzono taką możliwość. Odpowiednia definicja warunków brzegowych pierwszego rodzaju realizowana jest poprzez zaznaczanie wybranych obszarów środowiska. Należy tutaj podkreślić, że pojęcie warunków brzegowych w wewnętrznym zagadnieniu Dirichleta nie oznacza wyłącznie warunków, które nakładane są na brzegi (kontury) analizowanego obszaru. *Dodatkowe* warunki brzegowe mogą być także wprowadzane do *wnętrza* obszaru. Określenie warunku brzegowego pierwszego rodzaju oznacza przede wszystkim warunek niezmienności wartości funkcji w danym punkcie analizowanego obszaru, tzn. ustalone wartości dla warunków brzegowych przed rozpoczęciem, w *trakcie* i po zakończeniu

obliczeń są zawsze takie same. Oczywiście warunkiem koniecznym rozwiązania jest zapewnienie ograniczenia analizowanego obszaru warunkami brzegowymi. W opracowanym edytorze po wprowadzeniu warunków brzegowych i uruchomieniu procedury rozwiązującej równanie Laplace'a sprawdzana jest poprawność nałożonych warunków. W przypadku stwierdzenia jakichkolwiek nieprawidłowości obliczenia nie zostaną rozpoczęte. Ze względów dydaktycznych, w algorytmie uwzględniono także opcjonalną możliwość zadania warunków brzegowych drugiego rodzaju, nazywanych w literaturze warunkami Neumanna. Warunki brzegowe drugiego rodzaju definiowane są poprzez pierwszą pochodną normalną poszukiwanej funkcji w punktach należących do brzegu ograniczonego obszaru. W zaimplementowanym algorytmie, wprowadzono możliwość definiowania wewnątrz ograniczonego obszaru warunków spełniających równanie dla pochodnej kierunkowej $grad \phi = 0$. Warunek ten sugestywnie prezentuje umieszczenie wewnątrz analizowanego obszaru obiektów ekwipotencjalnych np. metalowych płytek, drutów itp. Studenci podczas zajęć dydaktycznych mogą jakościowo „potwierdzić” wzajemne oddziaływanie wymuszonych obszarów ekwipotencjalnych czyli warunków drugiego rodzaju z warunkami zadanych wartości funkcji, czyli warunkami brzegowymi pierwszego rodzaju, aż do spełnienia zasady zachowania minimum energii. Rozwiązanie takiego zagadnienia poprzez obliczenia symulacyjne może również przybliżyć znane studentom z kursów teoretycznych o elektryczności, magnetyzmie i inne interesujące zależności.



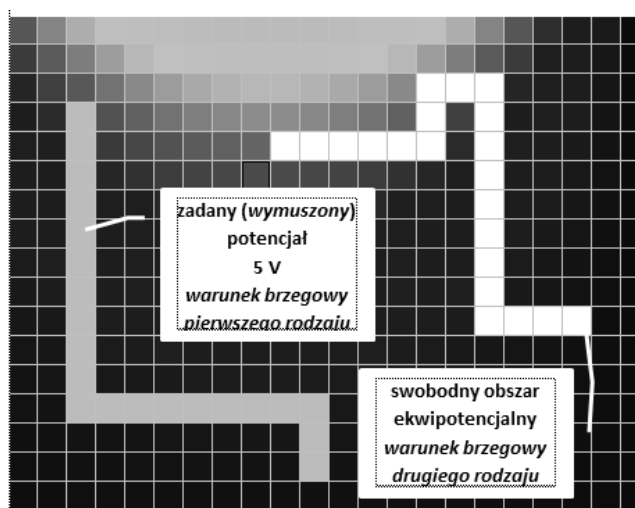
Rys. 4. Przykład zaprogramowania warunków brzegowych ograniczających obszar dla wewnętrzne-go zagadnienia Dirichleta (*wersja czarno-biała publikacji nie oddaje niestety pełnych efektów wizualnych podczas wprowadzania ograniczeń*)

Aplikacja edytora *środowiska* zawiera także wiele dodatkowych opcji technicznych wykorzystywanych podczas modelowania środowiska jak i jego udostępniania osobnikom poprzez lokalną sieć teleinformatyczną. Z punktu widzenia organizacji zajęć dydaktycznych, nauczyciel pełni funkcję operatora terminala, programuje warunki brzegowe dla algorytmu ewolucyjnego oraz dodatkowo przydziela kody identyfikacyjne dla populacji osobników poszczególnym grupom (zespołom) laboratoryjnym. Kody identyfikacyjne zabezpieczają przed możliwością dostępu do opisu genotypów i fenotypów „obcych” osobników, których cechami zarządzają inne grupy studenckie. Wymienione ograniczenie wypełnia postawione wymaganie ogólne w zakresie niedopuszczenia do jakichkolwiek identyfikacji zdarzeń w sposób niezgodny z przyjętymi zasadami komunikacji i wymiany informacji.

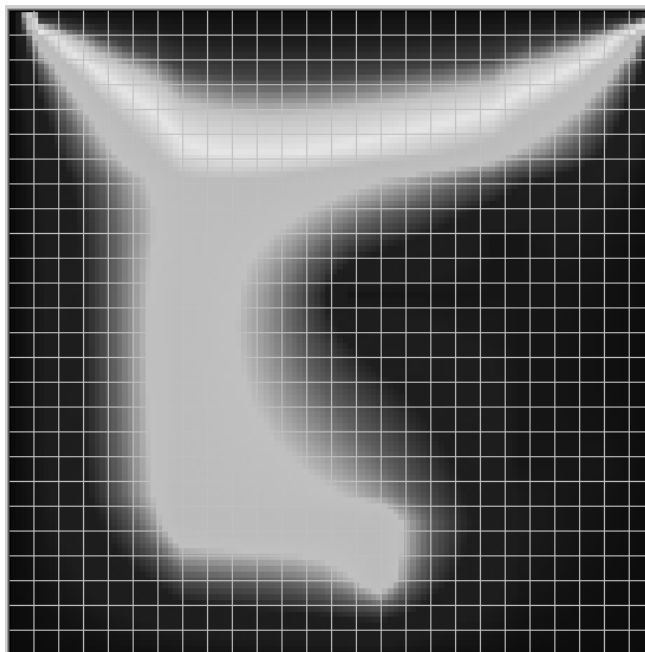
Przykład graficznego interfejsu umożliwiającego programowanie warunków brzegowych przedstawiono na rysunkach 4 i 6.

	0	1	2	3	4	5	6	7	8	9	10	11
0	000.000											
1		004.999	006.973	007.898	008.397	008.695	008.884	009.005	009.083	009.125	009.149	009.157
2		003.023	004.997	006.220	006.994	007.499	007.835	008.057	008.198	008.280	008.317	008.319
3		002.097	003.770	004.992	005.861	006.473	006.899	007.188	007.372	007.477	007.519	007.513

Rys. 5. Opcja wewnętrznego arkusza aplikacji dla podglądu obliczanych wartości



Rys. 6. Fragment obszaru z zaprogramowanymi warunkami brzegowymi pierwszego i drugiego rodzaju „wewnątrz” obszaru dla wewnętrznego zagadnienia Dirichleta

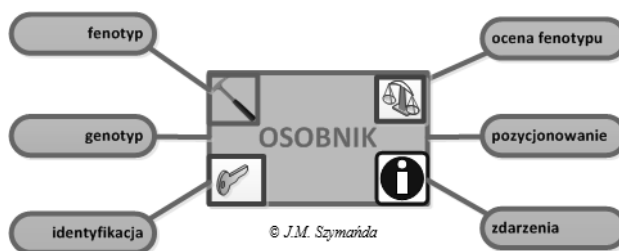


Rys. 7. Mapa potencjalowa po rozwiązaniu równania Laplace'a dla przykładowo zaprogramowanych warunków brzegowych (wersja czarno-biała publikacji nie oddaje niestety pełnych efektów wizualnych wyznaczanych map potencjalowych)

5.2. Ogólne zasady programowania *osobników*

Uwzględniając wymagania programowe i plany dydaktyczne obowiązujące na Wydziale Elektrycznym Politechniki Wrocławskiej, w założeniach projektowych przyjęto, że aplikacje reprezentujące *osobniki* będą programowane przez studentów w językach programowania: *ANSI C* w połączeniu z pakietem *Dev-C++* oraz *PASCAL* w połączeniu z pakietem *FreePascal* [3,4]. Oba pakiety należą do grupy produktów tzw. wolnego oprogramowania i nie wymagają posiadania licencji na ich użytkowanie w celach niekomercyjnych oraz edukacyjnych. Dodatkowo studentom udostępniana jest dedykowana dynamiczna biblioteka *DLL* z przygotowanym interfejsem procedur realizujących zadania dwukierunkowej wymiany danych w trybie zapytań (kwerend) *osobnik-środowisko-osobnik*. W tym obszarze wykonywane są także niektóre „nie dostępne” dla *osobnika* wewnętrzne procedury algorytmu ewolucyjnego (zob. pkt. 5). Aplikacja *osobnika*, niezależnie od wariantu języka programowania przyjmuje jednakową strukturę programu zgodną z zasadą programowania podstawowego opisanymi przez autora w publikacjach [3, 4] i zawiera wszystkie niezbędne informacje umożliwiające poprawne działanie algorytmu w trybie konsoli tekstowej. Sugerowany tryb uruchamiania aplikacji pozwa-

la większą uwagę zwrócić na zagadnienia stabilności algorytmów bez konieczności rozbudowywania programów o elementy i obiekty dodatkowe związane z graficznymi interfejsami konsoli wejściowo-wyjściowej. Na rysunku 8 przedstawiono uproszczony schemat funkcjonalny algorytmu osobnika, w którym przede wszystkim wyróżniono dwie grupy parametrów. Pierwsza to parametry przypisujące cechy podstawowe osobnikowi (*genotyp*, *fenotyp*, *identyfikacja*), druga to parametry określające bezpośrednie oddziaływanie osobnika ze środowiskiem (*pozycjonowanie*, *zdarzenia*, *ocena fenotypu*). Pozycjonowanie to sterowanie przemieszczaniem (ruch) osobnika w środowisku zdarzenie natomiast to informacja zwrotną jaką otrzymuje osobnik ze środowiska np. po napotkaniu innego osobnika, zaporę, wartości z przestrzeni rozwiązań zagadnienia Dirichleta i wielu innych, ważnych ze względu na właściwe działanie algorytmu ewolucyjnego.



Rys. 8. Uproszczony schemat funkcjonalny osobnika dla projektów dydaktycznych

6. PODSUMOWANIE - OCZEKIWANE EFEKTY DYDAKTYCZNE

Zaprezentowana tematyka modelowania algorytmów ewolucyjnych ze szczególnym uwzględnieniem wpływu warunków brzegowych i rozwiązania wewnętrznego zagadnienia Dirichleta, planowana jest do wprowadzenia do realizacji podczas zajęć projektowo-laboratoryjnych w semestrze letnim w roku akademickim 2015/2016 w ramach kursów „Metody numeryczne w technice” dla studentów II roku studiów stacjonarnych na kierunku Elektrotechnika oraz kursu „Elementy sieci komputerowych” dla studentów II roku na kierunku Mechatronika. Planuje się również, że studenci podczas zajęć projektowych w 3-4 osobowych grupach, z odpowiednio określonymi zadaniami i rozdziałem kompetencji, będą oceniani w kontekście założonych w „Karcie przedmiotu” tzw. „Przedmiotowych efektów kształcenia” w tym efekcie: „PEK_W02 - jest w stanie zaproponować odpowiedni algorytm numeryczny do rozwiązania zadania inżynierskiego”. Rozwiązywanie tematów problemowych związanych z podstawowym programowaniem złożonych algorytmów optymalizacyjnych, będzie dobrym miernikiem oceny efektów nabywania przez studentów umiejętności diagnozy problemu. Innymi słowy, studenci w takim wariantcie kształcenia będą mogli większą swoją uwagę skierować na isto-

tę realizacji algorytmu numerycznego i jego prawidłowego zaprogramowania bez konieczności rozbudowywania programu i wykorzystywania systemowych procedur (obiektów) prezentacyjno-graficznych .

LITERATURA

- [1] Golberg D.E.: Algorytmy genetyczne i ich zastosowania: WNT, Warszawa 1995.
- [2] Michalewicz Z: Algorytmy genetyczne + struktury danych=programy ewolucyjne: WNT, 2003.
- [3] Szymańda J.M.:Modelowanie symulacyjne z pakietem dydaktycznym SiSTLab20: Poznan University of Technology Academic Journals. Electrical Engineering: ISSN 1897-0737: Nr 82, s. 253-260, Poznań 2015.
- [4] Szymańda J.M.: Pakiet dydaktyczny *SiSTLab20* – instrukcja użytkownika: Skrypt dydaktyczny PWR/W5/K1, 2014.

BOUNDARY CONDITIONS IN EVOLUTIONARY ALGORITHMS – PROPOSALS FOR TEACHING

In this paper the proposal of the association of boundary conditions for the issue to seek a solution of the elliptic differential equations with simulation parameters affecting the environment on selected subjects in evolutionary algorithms. In particular, the association used the boundary conditions and the solution of the Dirichlet problem with the parameterization of selected characteristics of individuals performing the task of searching the area $D(X)$ in accordance with a specific function to identify objects. This issue is presented in the context of possible implementations of teaching for students directions engineering electrical, automation and robotics and mechatronics.

(Received: 31. 01. 2016, revised: 7. 03. 2016)