

Autoreferat

1 Imię i Nazwisko

Rafał Biedrzycki

2 Posiadane dyplomy, stopnie naukowe lub artystyczne – z podaniem podmiotu nadającego stopień, roku ich uzyskania oraz tytułu rozprawy doktorskiej

- Stopień doktora nauk technicznych w zakresie informatyki – tytuł rozprawy doktorskiej: „Konstruktywna indukcja w zadaniu klasyfikacji sekwencji DNA”, 2009. Wydział Elektroniki i Technik Informatycznych, Politechnika Warszawska. Rozprawa została wyróżniona.
- Tytuł zawodowy magistra inżyniera – specjalność: Inżynieria oprogramowania i Systemy Informatyczne, 2003, Wydział Elektroniki i Technik Informatycznych, Politechnika Warszawska.

3 Informacja o dotychczasowym zatrudnieniu w jednostkach naukowych lub artystycznych

- październik 2016 – obecnie: Instytut Informatyki, Wydział Elektroniki i Technik Informatycznych, Politechnika Warszawska. Adiunkt.
- październik 2009 – wrzesień 2016: Instytut Systemów Elektronicznych, Wydział Elektroniki i Technik Informatycznych, Politechnika Warszawska. Adiunkt.
- luty 2009 – wrzesień 2009: Instytut Systemów Elektronicznych, Wydział Elektroniki i Technik Informatycznych, Politechnika Warszawska. Asystent.

4 Omówienie osiągnięć, o których mowa w art. 219 ust. 1 pkt. 2 ustawy z dnia 20 lipca 2018 r. Prawo o szkolnictwie wyższym i nauce (Dz. U. z 2021 r. poz. 478 z późn. zm.).

Osiągnięciem, o których mowa w art. 219 ust. 1 pkt. 2 ustawy z dnia 20 lipca 2018 r. Prawo o szkolnictwie wyższym i nauce jest cykl powiązanych tematycznie artykułów naukowych, pt. „Analiza, rozwój i zastosowania wybranych algorytmów z rodziny algorytmów ewolucyjnych”.

[A1] Jarosław Arabas, **Rafał Biedrzycki**. *Quasi-Stability of Real Coded Finite Populations*. Parallel Problem Solving from Nature – PPSN XIII, 2014, Vol. 8672, pp. 872-881. Konferencja międzynarodowa z grupy A (według CORE), 15 pkt MNiSW (obecnie 140 pkt).

W pracy tej mój udział polegał na analitycznej analizie zachowania punktu środkowego populacji, zaplanowaniu i przeprowadzeniu wszystkich badań eksperymentalnych oraz współtworzeniu tekstu artykułu. Mój udział szacuję na 45%.

- [A2] **Rafał Biedrzycki**, Jarosław Arabas, Agata Jasik, Michał Szymański, Paweł Wnuk, Piotr Wasylczyk, Anna Wójcik-Jedlińska. *Application of Evolutionary Methods to Semiconductor Double-Chirped Mirrors Design*. Parallel Problem Solving from Nature – PPSN XIII, 2014, Vol. 8672, pp. 761-770. Konferencja międzynarodowa z grupy A (według CORE), 15 pkt MNiSW (obecnie 140 pkt).

W pracy tej mój udział polegał na poprawieniu implementacji modelu zwierciadła, zapoznaniu się z metodami projektowania pozwalającymi na wygenerowanie rozwiązań początkowych, implementacji procesu generowania rozwiązań początkowych. Opracowałem i przeanalizowałem różne warianty funkcji celu, dokonałem wyboru i konfiguracji zestawu algorytmów optymalizacyjnych do dalszych badań. Przeprowadziłem wszystkie badania eksperymentalne związane z algorytmami optymalizacji. Stworzyłem pierwszą wersję tekstu w części związanej z algorytmami ewolucyjnymi, badaniami i ich wynikami. Brałem udział w tworzeniu kolejnych wersji tekstu całego artykułu. Trudno tu jednoznacznie określić udział procentowy. Objętościowo, część związaną z artykułami ewolucyjnymi artykułu można oszacować na 70%, a część fizyczną na 30%. Jeżeli wziąć pod uwagę wkład w tematykę związaną z algorytmami ewolucyjnymi to mój udział procentowy szacuję na 95%. Większość współautorów odpowiedzialna była za wykonanie i badania zwierciadła będącego wynikiem optymalizacji lub za przygotowanie modelu teoretycznego zwierciadła.

- [A3] **Rafał Biedrzycki**, Dorota Jackiewicz, Roman Szewczyk. *Reliability and Efficiency of Differential Evolution Based Method of Determination of Jiles-Atherton Model Parameters for X30Cr13 Corrosion Resisting Martensitic Steel*. Journal of Automation, Mobile Robotics & Intelligent Systems, vol. 8, no. 4, pp. 63-68, 2014. 8 pkt MNiSW (obecnie 40 pkt).

W pracy tej mój udział polegał na rozwiązaniu zadania optymalizacji tak, aby rozwiązanie to spełniało kryterium jakościowe, a algorytm optymalizacji kryterium wydajnościowe. W tym celu przeanalizowałem funkcję celu, wybrałem algorytmy ewolucyjne dające nadzieję na realizację zadania. Wybrałem również zestaw operatorów mutacji do DE. Zaproponowałem odpowiednie zestawy parametrów algorytmów optymalizacji. Zaplanowałem i przeprowadziłem badania eksperymentalne. Dokonałem analizy danych, wybrałem sposób prezentacji wyników. Byłem wyłącznym autorem tekstu w zakresie dotyczącym algorytmów ewolucyjnych oraz prezentacji i omówienia wyników badań. Trudno tu jednoznacznie określić udział procentowy. Jeżeli wziąć pod uwagę wkład w tematykę związaną z algorytmami ewolucyjnymi to mój udział procentowy szacuję na 95%. Jeżeli wziąć pod uwagę udział na poziomie tekstu artykułu to jest to około 60%.

- [A4] Jarosław Arabas, **Rafał Biedrzycki**. *Improving evolutionary algorithms in a continuous domain by monitoring the population midpoint*. IEEE Transactions on Evolutionary Computation, vol. 21, no. 5, pp. 807-812, 2017. IF=8,124, 50 pkt MNiSW (obecnie 200 pkt).

W pracy tej mój udział polegał na zaplanowaniu i przeprowadzeniu badań eksperymentalnych. Wybrałem zastosowane algorytmy, samodzielnie zaimplementowałem DE, GEA, ES, JADE i SADE. Przeprowadziłem analizę wyników, opracowałem sposób ich prezentacji. Wspomniane wyniki przedstawiłem i opisałem w rozdziałach III i IV. Opracowałem pierwszą wersję tekstu

w zakresie dotyczącym planowania eksperymentów i omówienia ich wyników. Brałem udział w redakcji całości treści artykułu. Mój udział szacuję na 50%.

[A5] **Rafał Biedrzycki**. *A Version of IPOP-CMA-ES Algorithm with Midpoint for CEC 2017 Single Objective Bound Constrained Problems*. IEEE Congress on Evolutionary Computation (CEC), 2017. Konferencja międzynarodowa z grupy B (według CORE), 15 pkt MNiSW (obecnie 70 pkt).

[A6] **Rafał Biedrzycki**, Jarosław Arabas, Dariusz Jagodziński. *Bound constraints handling in Differential Evolution: An experimental study*. Swarm and Evolutionary Computation, vol. 50, no. 100453, 2019. IF=3,818, 140 pkt MNiSW.

W pracy tej mój udział polegał na wykonaniu przeglądu literatury i kodów źródłowych w celu wyszukania metod uwzględniania ograniczeń kostkowymi. Dla każdej metody wybrałem najbardziej w moim mniemaniu odpowiednią nazwę, dokonałem wyodrębnienia cech metod i ich przydziału do kilku grup. Zaplanowałem i przeprowadziłem badania eksperymentalne z użyciem własnych implementacji algorytmów: DE, DE/local2best, SADE, JADE, jSO, BBDE. Badania przeprowadziłem zarówno na funkcji kwadratowej jak i na funkcjach z CEC 2017. Opracowałem i omówiłem wyniki. W artykule znajdują się one na rysunkach od 11 do 18 oraz w tabelach 1-4. Opracowałem tekst artykułu w zakresie omówienia metod uwzględniania ograniczeń, pseudokodu z rysunku 1, opisu wykonanych przeze mnie eksperymentów, ich wyników i wniosków. Brałem udział w redakcji całości tekstu. Mój udział szacuję na 50%.

[A7] **Rafał Biedrzycki**. *Handling bound constraints in CMA-ES: An experimental study*. Swarm and Evolutionary Computation, vol. 52, no. 100627, 2020. IF=7,177, 140 pkt MNiSW.

[A8] **Rafał Biedrzycki**, Kamil Kwiatkowski, Paweł Cichosz. *Compressor Schedule Optimization for a Refrigerated Warehouse Using Metaheuristic Algorithms*. IEEE Congress on Evolutionary Computation (CEC), 2021. Konferencja międzynarodowa z grupy B (według CORE), 70 pkt MNiSW.

W pracy tej mój udział polegał na zaplanowaniu i wykonaniu wszystkiego co związane jest z optymalizacją, a także stworzeniu zawartości artykułu z wyłączeniem fragmentu wprowadzającego opis chłodni i opisującego model chłodni. Poprawiłem wydajność otrzymanego modelu chłodni. Zaproponowałem kilka różnych podejść do zdefiniowania rozwiązania tego zadania, opracowałem i zaimplementowałem heurystykę generującą punkt referencyjny. Zaplanowałem i przeprowadziłem serię eksperymentów numerycznych. Opracowałem i przedyskutowałem wyniki eksperymentów. Swój udział procentowy w rozwiązaniu problemu optymalizacyjnego (zaczynając od opracowania funkcji celu) szacuję na 100%. Swój udział w pracach nad zawartością artykułu (tekst, ilustracje, tabele) szacuję na 95%.

[A9] **Rafał Biedrzycki**. *Comparison with State-of-the-Art: Traps and Pitfalls*. IEEE Congress on Evolutionary Computation (CEC), 2021. Konferencja międzynarodowa z grupy B (według CORE), 70 pkt MNiSW.

[A10] **Rafał Biedrzycki**, Jarosław Arabas, Eryk Warchulski. *A Version of NL-SHADE-RSP Algorithm with Midpoint for CEC 2022 Single Objective Bound Constrained Problems*. IEEE Congress on Evolutionary Computation (CEC), 2022. Konferencja międzynarodowa z grupy B (według CORE), 70 pkt MNiSW.

W pracy tej mój udział polegał na opracowaniu i implementacji szeregu modyfikacji algorytmu NL-SHADE-RSP. Zaplanowałem i przeprowadziłem wszystkie eksperymenty z wyjątkiem użycia algorytmu DES, który to był wykorzystany w artykule jako jeden z algorytmów referencyjnych. Stworzyłem całą zawartość artykułu. Zawartość ta podlegała później modyfikacjom w warstwie tekstowej, które były wprowadzane przez wszystkich współautorów. Mój udział szacuję na 90%.

4.1 Wstęp

Spośród 27 moich artykułów, 22 prace są zgodne z tematyką przedstawionego cyklu publikacji. Do dokładnej oceny wybrałem 10 prac, które według mnie najwięcej wnoszą w rozwój dziedziny i zostały opublikowane w wysokopunktowanych miejscach. Zależności występujące pomiędzy tymi artykułami przedstawiłem na rysunku 1. Prezentowany cykl badań dotyczy trzech przeplatających się elementów: analizy zachowania, rozwoju, a także zastosowań wybranych algorytmów ewolucyjnych. Analiza zachowania algorytmów ewolucyjnych pozwala na lepsze ich zrozumienie, a przez to m.in. na dokonywanie świadomych wyborów przy próbach ich praktycznych zastosowań. Rozwój algorytmów wynika z ich analizy i prowadzi do ich ulepszenia. Metody wybrane na podstawie analizy lub stworzone w wyniku rozwoju stosuje do rozwiązywania rzeczywistych problemów. Dzięki temu ujawniają się problemy, których nie da się zauważyć bez rzeczywistej aplikacji. Często algorytmy uzyskujące dobre wyniki na zestawie funkcji benchmarkowych nie radzą sobie na danej klasie rzeczywistych problemów. Zatem, analiza, rozwój i praktyczne zastosowania napędzają się wzajemnie.

Ponieważ w moim dorobku znajdują się prace wieloautorskie, dalsza część tekstu jest często pisana w pierwszej osobie, aby nie było wątpliwości co do mojego wkładu. Dokładny opis wkładu mojego oraz współautorów znajduje się w załączonych oświadczeniach.

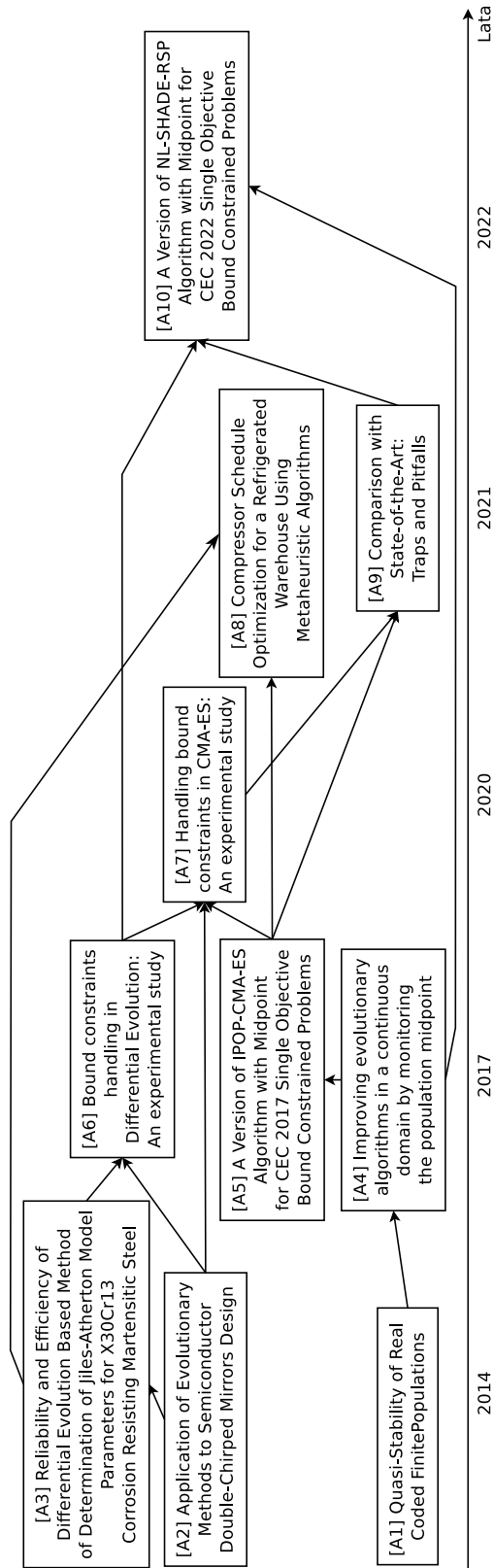
4.2 Analiza zachowania wybranych algorytmów ewolucyjnych

Algorytmy ewolucyjne (EA) to metody optymalizacji globalnej inspirowane ewolucją naturalną. Badania nad nimi prowadzi się już od około 50 lat. Większość wysiłków badaczy skupia się na opracowywaniu nowych algorytmów, tymczasem już obecnie istniejąca ogromna ich liczba utrudnia ich praktyczne wykorzystanie, ponieważ nie wiadomo, który algorytm będzie odpowiedni do konkretnego zadania. W wyborze algorytmu i ustawieniu jego parametrów powinny pomóc wyniki teoretycznej analizy jego zachowania. Niestety, najczęściej analizy takie wykonuje się przy bardzo wymagających założeniach, co prowadzi do niewielkiej praktycznej przydatności wniosków wynikających z tych analiz. Wynika z tego, że należy dążyć do przyjmowania jak najbardziej realistycznych założeń, a wnioski płynące z analizy teoretycznej należy weryfikować eksperymentalnie.

Pogłębienie zrozumienia zachowania rzeczywistych algorytmów ewolucyjnych pozwala na dokonywanie bardziej świadomego doboru odpowiedniego algorytmu do postawionego zadania. Lepsze zrozumienie tych algorytmów umożliwia również dostrzeżenie ich wad i wprowadzanie ulepszeń.

Analiza dynamiki zmian położenia środka populacji i wariancji rozproszenia

O ile w zastosowaniach inżynierskich stabilność jest cechą pożądaną systemów, o tyle w algorytmie ewolucyjnym populacja w stanie quasi-stabilnym skupia się na eksploatacji aktualnego optimum lokalnego (przy rozsądnie małej sile mutacji) lub przypomina przeszukiwanie typu Monte Carlo, przy



Rysunek 1: Powiązania pomiędzy wskazanymi artykułami.

dużej sile mutacji. Wyjście ze stanu quasi-stabilnego pozwala na przejście do obszaru przyciągania innego optimum, co jest pożądane dla algorytmów optymalizacji globalnej.

W moich badaniach analizowałem dynamikę zmian położenia środka populacji w algorytmie ewolucyjnym ograniczonym do selekcji i mutacji. Wyniki tych badań zostały opublikowane w pracy [A1]. Analiza ta doprowadziła do zdefiniowania pojęcia quasi-stabilności jako zdolności do pozostawiania populacji w obszarze przyciągania ekstremum przez wiele kolejnych iteracji. W stanie quasi-stabilnym zmieniają się położenia osobników w kolejnych iteracjach, ale średnia i wariancja tych położzeń fluktuują wokół charakterystycznych wartości. Wartości te zostały wyznaczone analitycznie, używając modelu z nieskończonym rozmiarem populacji. Ponieważ nieskończone populacje nie występują w praktyce, w dalszych badaniach przyjęto założenie o ograniczonym rozmiarze populacji oraz o braku niezależności zreprodukowanych osobników. W związku z odrzuceniem założenia o niezależności, dla zachowania takiej samej wariancji populacji potrzeba więcej osobników. Skorygowany rozmiar populacji został wyznaczony analitycznie i zweryfikowany eksperymentalnie. Dzięki zastosowaniu wyznaczonej korekty średni błąd przewidywań wariancji populacji zmalał z 24 do 1 procenta.

W wyniku badań udało się również analitycznie wyznaczyć jak quasi-stabilność algorytmu zależy od jego parametrów. Wyznaczono obszar quasi-stabilności, w którym, z określonym prawdopodobieństwem, znajdują się stany populacji działającego algorytmu. Wyznaczone zależności są w dobrej zgodności z zachowaniem rzeczywistego algorytmu, co zostało potwierdzone eksperymentalnie.

Zbadano również zachowanie algorytmu w stanach stabilnych wynikających z małej i dużej siły mutacji, oraz w stanach pośrednich. Badania eksperymentalne przeprowadzono na funkcji Galara. Funkcja ta jest sumą dwóch wzgórz gaussowskich tak sparametryzowanych, aby obszar przyciągania optimum lokalnego był szerszy niż optimum globalnego. Populację początkową umieszczono w optimum globalnym. Dla niewielkiej wariancji mutacji populacja znajduje się w stanie quasi-stabilnym w okolicy optimum globalnego. Dla średniej siły mutacji populacja sporadycznie opuszcza obszar stabilności, poruszając się w kierunku drugiego optimum. Dla dużej siły mutacji populacja na początku pozostaje w okolicy optimum globalnego, po czym przesuwa się w kierunku optimum lokalnego, gdzie pozostaje do końca ewolucji. Porzucenie optimum globalnego jest ilustracją zjawiska znanego pod angielską nazwą: „survival of the flattest”, które polega na preferowaniu osobników, których wartość funkcji celu jest mało wrażliwa na zmiany położenia w przestrzeni cech. Dla bardzo dużej siły mutacji populacja przebywa w pojedynczym obszarze quasi-stabilności, a jej środek nie pokrywa się z żadnym optimum. Co ciekawe, zmniejszenie siły mutacji w takiej sytuacji najczęściej nie spowoduje przejścia do optimum globalnego, a spowoduje powrót do optimum lokalnego.

Wyniki tych prac pozwalają na lepsze zrozumienie dynamiki algorytmu ewolucyjnego. Prace te mogą być wykorzystane w praktyce do modyfikacji kryterium zatrzymania istniejących algorytmów oraz do świadomego równoważenia eksploracji i eksploatacji.

Analiza sposobu aproksymacji optimum

W znanych algorytmach ewolucyjnych za rozwiązanie zadania optymalizacji uznaje się najlepszego spośród znalezionych osobników. W ramach badań opublikowanych w pracy [A4] wykazano analitycznie i eksperymentalnie, że najlepszym aproksymatorem szukanego optimum jest środek populacji. Podczas analizy założono, że populacja składa się z niezależnych osobników wygenerowanych z rozkładu normalnego. Ponieważ założenie to w praktyce nie jest spełnione wykonano szereg badań eksperymentalnych. W związku z tym, że populacja rzeczywistego algorytmu jest stosunkowo mało liczna i zawiera zależne od siebie osobniki, oprócz średniej arytmetycznej użyto mediany jako

estymatora środka populacji. Oprócz wykorzystania klasycznego algorytmu ewolucyjnego, wykorzystano również strategię ewolucyjną $ES(\mu + \lambda)$, strategię ewolucyjną opartą na adaptacji macierzy kowariancji (CMA-ES), algorytm ewolucji różnicowej oraz kilka jego nowoczesnych następców (SADE, JADE, b6e6rl).

Dla funkcji kwadratowej wykazano eksperymentalnie, że zastosowanie średniej jako aproksymatora optimum w algorytmie ewolucji różnicowej (DE) dało około 32-krotnie lepsze wyniki niż podejście klasyczne. Dla algorytmu ewolucyjnego (GEA), JADE, ES, i b6e6rl, uzyskano kilkunastokrotną poprawę. W przypadku SADE uzyskano siedmiokrotną poprawę, a dla CMA-ES – około dwukrotną.

Dalsze badania prowadzono na trudnych, wielowymiarowych funkcjach z benchmarków CEC 2005 i CEC 2013. Przyjęto dwa skrajne założenia: 1) ewaluacja jakości punktu środkowego nie zmniejsza dostępnej dla algorytmu liczby ewaluacji; 2) liczba dostępnych algorytmowi ewaluacji jest zmniejszana o 1 w każdej iteracji. W praktycznych zastosowaniach można spodziewać się pośredniego zachowania, można np. przyjąć, że punkt środkowy będzie ewaluowany w co k -tej iteracji lub będzie ewaluowany po wykryciu braku poprawy oceny najlepszego z dotychczas znalezionych osobników.

Przy założeniu zerowego kosztu oceny środka populacji łatwo zauważyć, że większość algorytmów uzyskało znaczną poprawę jakości wyniku, a SADE, JADE i b6e6rl niewielką. W przypadku wzięcia pod uwagę kosztu oceny punktu środkowego dla GEA, ES, CMA-ES i DE można stwierdzić, że częściej występuje poprawa jakości wyniku niż pogorszenie. W przypadku JADE, SADE i b6e6rl liczba popraw była niewielka, była niewiele większa od liczby pogorszeń.

Mediana jako estymator środka populacji dała nieco gorsze wyniki niż średnia, a dodatkowo jej wyznaczenie wiąże się z większym nakładem obliczeniowym dlatego zalecanym estymatorem jest średnia.

Analiza wpływu metod uwzględniania ograniczeń kosztowych na własności algorytmów ewolucji różnicowej

W literaturze opisującej nowe algorytmy zgłoszone do konkursów optymalizacyjnych z serii CEC trudno doszukać się informacji o tym, jakie metody uwzględniania ograniczeń kosztowych zostały użyte. Jeżeli już autorzy podają opis metody, to okazuje się, że stosują metody najprostsze w implementacji, takie jak rzutowanie na ograniczenia. W literaturze związanej z problemem uwzględniania ograniczeń główny nurt prac skupia się na skomplikowanych ograniczeniach funkcyjnych. Zatem nie wiadomo jaką metodę uwzględniania ograniczeń kosztowych wybrać podczas tworzenia nowego algorytmu, czy podczas rozwiązywania praktycznych problemów (np. [A2, A3]), a także jaki będzie miała ona wpływ na zachowanie algorytmów z rodziny ewolucji różnicowej (DE). Aby wypełnić tę lukę, przeprowadziłem badania, w wyniku których powstał artykuł [A6]. W ramach prac dokonałem przeglądu literatury oraz dostępnych kodów źródłowych w poszukiwaniu istniejących metod uwzględniania ograniczeń kosztowych. Okazało się, że część metod występuje pod różnymi nazwami w różnych publikacjach, a część nie ma nazwy. Ostatecznie skatalogowałem 14 metod, zaproponowałem dla nich nazwy, po czym dokonałem ich przydziału do trzech kategorii: metody naprawy, metody zachowujące dopuszczalność, funkcje kary. Dla części metod naprawy zaproponowałem dwie możliwości ich użycia: 1) naprawa modyfikuje genotyp, co w pracy nazwano podejściem lamarkowskim; 2) naprawa modyfikuje fenotyp, co nazwano podejściem darwinowskim. W sumie dało to 17 metod uwzględniania ograniczeń kosztowych. Wpływ zastosowania tych metod na jakość wyników optymalizacji zbadałem dla następujących algorytmów: DE, DE/local2best, SADE,

JADE, jSO, BBDE. Testy przeprowadziłem w dwóch scenariuszach: 1) używając 10-wymiarowej, kwadratowej funkcji celu ze zmienianą odległością optimum od ograniczenia; 2) używając zestawu 30 funkcji benchmarkowych z konkursu CEC 2017 w 10, 30, 50 i 100 wymiarach. Na podstawie analizy wyników badań dla funkcji kwadratowej stwierdziłem, że większość metod uwzględniania ograniczeń pogarsza tempo zbieżności algorytmu, niektóre poprawiają, a niektóre tempa tego nie modyfikują. Przyspieszenie tempa zbieżności należy uznać za niepożądane zachowanie, ponieważ wprowadza ono ukierunkowanie procesu przeszukiwania sprzyjając przejściu na ograniczenia, co w ogólności nie jest dobrym rozwiązaniem. Pożądane zachowanie można zaobserwować dla kary kwadratowej (*additive penalty*). Nie oznacza to automatycznie, że ten rodzaj kary jest właściwy dla wszystkich możliwych funkcji celu. Dla funkcji kwadratowej wspomniana kara po prostu odtwarza kształt funkcji oryginalnej.

Biorąc pod uwagę wyniki wszystkich eksperymentów przeprowadzonych na funkcji kwadratowej można stwierdzić, że wszystkie zbadane algorytmy są wrażliwe na wybór metody uwzględniania ograniczeń. W znakomitej większości przypadków zastosowanie lamarkowskiej wersji zawijania (*wrapping*) i reinicjacji (*reinitialization*) powoduje pogorszenie tempa zbieżności. Obie metody podczas zbliżania do ograniczenia generują coraz bardziej zróżnicowanych mutantów, przez co populacja odsuwa się od optimum. Lamarkowska wersja rzutowania (*projection*) zmniejsza tempo zbieżności, natomiast rzutowanie na punkt środkowy (*projection to midpoint*) zwiększa je. Pierwsza z tych metod powoduje zbliżanie się populacji do rogu hiperkostki wynikającej z ograniczeń, a druga utrudnia zbliżanie do tego rogu.

Wyniki badań wykazały, że algorytmy DE w połączeniu z metodami opartymi na funkcji kary dają dobre wyniki. Podobne wyniki uzyskuje się przy użyciu metod z grupy metod zachowujących dopuszczalność i lamarkowską wersję odbicia (*Lamarckian reflection*). Należy natomiast unikać metod darwinowskich oraz rzutowania (*projection*), zawijania (*wrapping*) i reinicjacji (*reinitialization*) w wersji lamarkowskiej.

Kolejne badania przeprowadziłem na funkcjach benchmarku CEC 2017 w 10, 30, 50 i 100 wymiarach. Ograniczenia kostkowe to $[-100,100]^n$ jednak autorzy benchmarku gwarantują, że optimum jest zawarte w kostce $[-80,80]^n$. Na podstawie tego można przypuszczać, że w przypadku tego benchmarku dobór metody uwzględniania ograniczeń nie jest istotny. Na podstawie wyników eksperymentów stwierdziłem, że wrażliwość na zastosowaną metodę uwzględniania ograniczeń zależy od wersji algorytmu. Najbardziej wrażliwe są CDE i BBDE, na trzecim miejscu jest DES. Różnice powodowane przez zastosowanie różnych metod uwzględniania ograniczeń stają się bardziej widoczne wraz ze wzrostem wymiarowości zadania optymalizacji.

Dla większości algorytmów najlepszą metodą jest powtórzenie mutacji (*resampling*). Warto jednak zauważyć, że dla problemów wielowymiarowych, gdy optimum leży blisko narożnika hiperkostki wynikającej z ograniczeń, może być trudno uzyskać rozwiązanie dopuszczalne. Dlatego dla tej metody zawsze warto zdefiniować maksymalną liczbę podjętych prób powtórzenia mutacji, po przekroczeniu której należy użyć innej metody. Innymi dobrymi metodami są *midpoint target*, *midpoint base* i *rand base*. Najprawdopodobniej użycie cech osobnika bazowego lub docelowego pozwala na dobre zachowanie podobieństw w dystrybucji punktów pomiędzy kolejnymi iteracjami, co pozwala na lepsze dopasowanie przestrzennego rozłożenia osobników w populacji do kształtu funkcji celu.

Spośród metod lamarkowskich najlepszą jest odbijanie (*reflection*). Metoda *substitution penalty* jest jedną z najlepszych dla DES, ale w połączeniu z innymi algorytmami plasuje się w środku rankingu. Metody oparte na karze są nisko w rankingu, szczególnie *additive penalty* użyta w połączeniu z CDE i BBDE. Metody darwinowskie zamykają ranking. Ich zastosowanie znosi wszelkie

mechanizmy zmuszające populację do pozostawania w obszarze dopuszczalnym.

Na podstawie wyników badań można jasno stwierdzić, że nawet w przypadku benchmarku, w którym optimum jest stosunkowo daleko od ograniczeń, wybór metody ich uwzględniania ma znaczenie. Spora grupa prostych, powszechnie stosowanych metod, prowadzi do uzyskania znacznie słabszych wyników niż metody wskazane w omawianej pracy.

Analiza wpływu metod uwzględniania ograniczeń kosztowych na własności CMA-ES

Kontynuacją badań przedstawionych w artykule [A6] są badania wpływu metod uwzględniania ograniczeń kosztowych na algorytm CMA-ES. Jest to jeden z ważniejszych z praktycznego punktu widzenia algorytmów, co potwierdzają również wyniki badań [A2, A5]. CMA-ES i DE opierają się na różnych założeniach, co za tym idzie wnioski wyciągnięte dla jednego z nich nie muszą przenosić się na drugi. Oprócz ogólnych metod uwzględniania ograniczeń kosztowych skatalogowanych w [A6] wyszukałem (zarówno w kodzie źródłowym jak i w artykułach naukowych), zaimplementowałem i zbadałem pięć metod stosowanych w połączeniu z CMA-ES, co dało w sumie 22 metody.

Badania podzieliłem na trzy etapy. W pierwszym z nich zbadałem wpływ metody uwzględniania ograniczeń na zdolności eksploatacyjne CMA-ES. Badania przeprowadziłem na trzech funkcjach jednomodalnych, których optimum znajduje się w obszarze dopuszczalnym: funkcji sferycznej, funkcji elipsoidalnej i funkcji *TwoAxes*. Zaobserwowałem, że CMA-ES w połączeniu z większością metod nie poradził sobie podczas zbliżania optimum do ograniczenia. Najlepszą zbieżnością wykazał się CMA-ES w połączeniu z transformacją (*transformation*). Nieco gorsze wyniki dało się uzyskać używając funkcji opartych na karze oraz *resampling* i *Darwinian reflection*. Wersja *weighted penalty Matlab implementation* używana w oficjalnej matlabowej implementacji CMA-ES poradziła sobie gorzej niż prostsza metoda nazywana *weighted penalty*. Zaskakujące jest to, że *Darwinian reflection*, czyli jedna z najprostszych metod, nie stosowana w kontekście CMA-ES, jest równie dobra jak metody specjalizowane tworzone z myślą o CMA-ES, wykorzystujące wiedzę zgromadzoną przez algorytm, np. wartość σ , czy zawartość macierzy C . Warto zwrócić uwagę, że transformacja jako metoda dająca najszybszą zbieżność nie może być postrzegana jako najlepsza. Kiedy metoda jest szybsza niż optymalizacja bez ograniczeń, to znaczy, że wprowadza ukierunkowanie procesu przeszukiwania, które okazało się korzystne dla rozważanych funkcji jednomodalnych, natomiast może być niekorzystne dla innych praktycznie istotnych funkcji. Przy założeniu, że szybkość zbieżności nie powinna być modyfikowana przez metody uwzględniania ograniczeń, za najlepsze metody z punktu widzenia wszystkich trzech analizowanych funkcji należy uznać *Darwinian reflection* i *resampling*.

W drugiej serii badań zastosowałem funkcje wielomodalne tak sparametryzowane, aby optimum globalne znalazło się w rogu hiperkostki będącej wynikiem ograniczeń. Jest to bardzo trudny do optymalizacji scenariusz, ponieważ w pobliżu optimum globalnego znakomita większość generowanych punktów nie będzie spełniała ograniczeń. Użyłem 20 odpowiednio zmodyfikowanych funkcji z CEC 2017.

Na podstawie wyników badań metody uwzględniania ograniczeń można przydzielić do trzech grup: liderzy, dobre metody i słabe metody. Najlepszą metodą jest *resampling*, co ciekawe jej przewaga rośnie wraz ze wzrostem wymiarowości zadania. Drugą metodą jest *Darwinian reflection*, której jakość nie zależy od wymiarowości zadania. Kolejne miejsca zajmuje grupa metod opartych na ważonej karze (*weighted penalty*), a zawijanie darwinowskie (*Darwinian wrapping*) zamyka grupę liderów. Grupa dobrych metod zawiera: *substitution penalty*, *multiplicative penalty*, *Darwinian projection*. Metoda *weighted penalty with trimmed median* jest pomiędzy grupą dobrych i słabych metod. Uzyskane przez nią wyniki silnie pogarszają się wraz ze wzrostem wymiarowości zadania.

Do grupy słabych zalicza się *Lamarckian projection*, która to w praktyce jest najczęściej spotykaną metodą. Transformacja, która to jest metodą domyślną w zaawansowanej implementacji CMA-ES oraz *additive penalty*, które jest używane w prostszych implementacjach również znalazły się w grupie słabych. Grupa ta zawiera również *reinitialization*, która to w literaturze jest uznawana za dobre podejście ze względu na to, że nie wprowadza dodatkowego ukierunkowania przeszukiwania. Zaskakujący jest wynik rzutowania na ograniczenia (*projection*). Łatwo przyjąć, że metoda ta jest dobrym wyborem gdy optimum leży na ograniczeniach. W praktyce jest jednak mało prawdopodobne, że powstanie mutant ze wszystkimi wymiarami poza ograniczeniem, dzięki czemu zostałyby przesunięty do optimum. Nawet kilka współrzędnych w obszarze dopuszczalnym wystarcza, aby osobnik utknął w optimum lokalnym. Kolejnym problemem wynikającym z modyfikacji współrzędnych osobników poza mechanizmami przewidzianymi w CMA-ES jest to, że może to wprowadzać w błąd mechanizmy samoadaptacji używane w CMA-ES, co z kolei jest wykrywane przez rozbudowane kryteria stopu.

W trzeciej serii badań użyłem dwóch niezmodyfikowanych zestawów funkcji benchmarkowych: CEC 2017 i BBOB. Wspomniane zestawy charakteryzują się tym, że ich autorzy zadbali o to aby optimum globalne znajdowało się relatywnie daleko od ograniczeń. Zaobserwowałem, że pomimo tego, metody uwzględniania ograniczeń mają zauważalny wpływ na jakość osiągniętych wyników. Wpływ ten jest tym wyraźniejszy im większą wymiarowość rozważamy. Większy wpływ zaobserwować można na BBOB niż na CEC. W tej konfiguracji trudno odpowiedzialnie wskazać zwycięzców, można jednak wskazać metody najslabsze. Jedną z nich jest *conservative*. Jej zastosowanie prowadzi do populacji wypełnionej wieloma kopiami jednego osobnika. Taka sytuacja jest postrzegana przez CMA-ES jako skutek istnienia płaskiego obszaru w krajobrazie funkcji celu. Algorytm próbuje opuścić ten wyimaginowany obszar zwiększając σ , co dodatkowo zwiększa liczbę osobników poza ograniczeniami. Nadmierne zwiększenie σ jest również problemem metody *projection to base*. Innymi metodami, których stanowczo należy unikać są *Darwinian projection* i *Lamarckian wrapping*. Poniżej średniej znalazły się również metody: *projection to midpoint* i *additive penalty*.

Warto zauważyć, że *conservative*, która jest najgorszą metodą w połączeniu z CMA-ES była lepsza od reinicjacji w połączeniu z DE. W skojarzeniu z CMA-ES *Darwinian reflection* jest jedną z najlepszych metod, zaś w kontekście DE było jedną z najgorszych. Dlatego też jakości metod uwzględniania ograniczeń nie powinno się oceniać w oderwaniu od metody optymalizacji.

Używając standardowej dla CMA-ES metody *weighted penalty* jako metody odniesienia stwierdziłem, że lepsze od niej są *Darwinian reflection* i *resampling*.

W ramach prac zbadałem również jak procent wygenerowanych rozwiązań niedopuszczalnych zależy od zastosowanej metody. Dla problemów CEC 1-4, dla których znaleziono optima globalne, wśród dobrych metod, najmniej (16%) rozwiązań poza ograniczeniami wygenerował algorytm stosujący *multiplicative penalty*. Około 10 punktów procentowych więcej dało zastosowanie *weighted penalty* i kolejne 6 punktów procentowych więcej powstało przy użyciu: *resampling*, *weighted penalty with decreasing weights* i *weighted penalty Matlab implementation*. Najwięcej, bo około 46% rozwiązań niedopuszczalnych, powstało przy zastosowaniu *transformation*. Dla funkcji, dla których nie udało się odnaleźć optimum globalnego, zastosowanie najgorszej z metod doprowadziło do wygenerowania poniżej 4 procent punktów niedopuszczalnych.

Oprócz tego zbadałem jak zastosowanie rozważanych metod wpływa na czas pracy algorytmu (bez uwzględniania czasu potrzebnego na wyznaczenie wartości funkcji celu). W większości przypadków wpływ ten jest pomijalny. Są jednak dwie metody wyróżniające się negatywnie w tym zakresie. Są to *resampling* i *weighted penalty with decreasing weights*. Zaobserwowałem, że dla funkcji 1-4 re-

sampling w skojarzeniu z CMA-ES daje ponad pięciokrotnie wydłużenie czasu pracy w stosunku do czasu potrzebnego algorytmowi używającemu *Darwinian reflection*. Zastosowanie *weighted penalty with decreasing weights* powoduje, że algorytm wymaga w przybliżeniu dwukrotnie dłuższego czasu obliczeń. Pomimo tego, że dla funkcji 5-30 rozwiązania nie spełniające ograniczeń są generowane rzadziej, CMA-ES z *resampling* potrzebuje 3,7 raza więcej czasu.

Oprócz tego zbadałem, jak wnioski z badań przenoszą się do nowszej implementacji CMA-ES, oraz czy implementacja ta w połączeniu z najlepszą metodą będzie w stanie pokonać zwycięzców CEC 2017. Do badań wykorzystałem IPOP-CMA-ES, który to algorytm jest oparty na CMA-ES. Użyta implementacja domyślnie stosuje *transformation*. Porównałem ją z wersją, gdzie zaimplementowałem *Darwinian reflection*. Dodatkowo obie wersje porównałem z potomkami L-SHADE, tj. z SALSHADE i SALSHADE-cnEPSin. Algorytmy te były zwycięzcami konkursu CEC 2017.

Na podstawie wyników testu Wilcozona dla par można stwierdzić, że metoda zaproponowana (*Darwinian reflection*) była lepsza od domyślnej (*transformation*) w dziewięciu przypadkach, a gorsza w trzech. Pomimo tego, że IPOP nie był tworzony ani ustawiony pod kątem CEC 2017, zmodyfikowana wersja dała lepsze wyniki średnie niż potomkowie L-SHADE dla siedmiu funkcji. Zaproponowana metoda była najlepsza dla przesuniętych i obróconych wersji nieciągłych funkcji Rastrigina i Schwefela (numery 8 i 10). Dawała również dobre wyniki dla funkcji złożonych (21-30).

Wyniki tych badań przedstawiłem w artykule [A7].

Analiza rozbieżności pomiędzy specyfikacją algorytmu ewolucyjnego a jego praktycznymi implementacjami

Na podstawie zebranych wcześniej doświadczeń, a zwłaszcza tych związanych z pracami [A5, A7], zwróciłem uwagę na rozbieżności pomiędzy specyfikacją algorytmu ewolucyjnego a jego praktycznymi implementacjami. Zauważyłem, że specyfikacje algorytmów zawarte w artykułach naukowych nie są wystarczające do stworzenia kompletnej implementacji. Istniejące braki są różnie zapełniane przez różnych programistów, w wyniku czego otrzymujemy zbiór algorytmów o tej samej nazwie, a różnych własnościach. Wszyscy użytkownicy algorytmów ewolucyjnych powinni mieć świadomość istnienia tych różnic. Jeśli tymi użytkownikami są praktycy, to spotykając gorszą implementację, mogą niesłusznie zniechęcić się do wybranej metody lub do całej rodziny algorytmów. Gorzej sprawa wygląda, jeśli w oparciu o wybraną losowo implementację prowadzi się badania naukowe. Może to prowadzić do wyciągnięcia błędnych wniosków, które, jeśli zostaną opublikowane, mogą niesłusznie zatrzymać rozwój niektórych gałęzi drzewa nauki. Aby dobitniej pokazać, że różnice są znaczące i występują powszechnie skupiłem się na analizie najbardziej zaufanych i najlepiej opisanych implementacji. Wybrałem algorytm CMA-ES i implementacje w różnych językach programowania udostępnione przez Nikolasa Hansena, autora CMA-ES. Zaplanowałem i przeprowadziłem badania eksperymentalne wybranych implementacji. Użyłem benchmarku BBOB-2009. W celu porównania wyników implementacji skorzystałem z udostępnionych przez benchmark funkcji rysujących krzywe ECDF, tworzących zestawienia tabelaryczne, oraz porównujących algorytmy za pomocą odpowiednich testów statystycznych. Okazało się, że spośród 6 zbadanych wariantów algorytmu (5 implementacji) nie ma nawet jednej pary dającej różnice w wynikach, które dało by się wytłumaczyć tylko losową ich zmiennością. Przeanalizowałem kody źródłowe wybranych implementacji, znalazłem i opisałem występujące różnice. Większość różnic ukryta jest w kodzie uznawanym za pomocniczy, np. w procedurach uwzględniania ograniczeń kostkowych, w różnych warunkach stopu, a najczęściej w tych związanych z wykrywaniem niestabilności numerycznych, w różnych wartościach nieudokumentowanych parametrów. Rozbieżności w wynikach implementacji wynikają również z

tę, że implementacje te powstają w różnym czasie, podczas gdy sam algorytm się rozwija. Na podstawie wyników badań oraz analizy dostępnej literatury sformułowałem zestaw rekomendacji skierowanych zwłaszcza do naukowców zgłaszających propozycje nowych algorytmów oraz próbujących porównywać swoje algorytmy z obecnie najlepszymi:

1. Wydawcy powinni wymagać udostępnienia kodu źródłowego wszystkich nowych algorytmów metaheurystycznych.
2. Kod używany do przeprowadzenia eksperymentów powinien być również dostępny (co najmniej dla recenzentów).
3. Autorzy artykułów powinni używać starannie wybranej, zaufanej implementacji algorytmów używanych do porównań. Oprócz nazwy algorytmu powinni podawać również skąd został pobrany, a także wersję implementacji.
4. Autorzy powinni ujawniać jak wszystkie parametry używanych algorytmów zostały ustawione.
5. Autorzy implementacji powinni jasno określić na podstawie którego artykułu naukowego wykonali implementację, powinni również określić co wymagało uzupełnienia i jak to zostało zrobione.
6. Zarówno autorzy artykułów, jak i autorzy implementacji, powinni ujawniać, jakich metod uwzględniania ograniczeń kosztowymi użyli.

Rekomendacje te wraz z wynikami badań opublikowałem w pracy [A9].

4.3 Rozwój algorytmów ewolucyjnych

Pomimo dziesięcioleci rozwoju ciągle powstają nowe algorytmy ewolucyjne. Obecnie najczęściej nie są one całkowicie nowe, a proponowany algorytm jest rozszerzeniem lub połączeniem istniejących. Tęgo typu rozwój dotyczy szczególnie rodziny algorytmów ewolucji różnicowej, gdzie stopniowo dodawano komponenty, a nazwę algorytmu bazowego wzbogacano o stosowne przedrostki lub przyrostki.

RB-IPOP

W ramach badań związanych z ulepszaniem istniejących algorytmów zaproponowałem nowy wariant algorytmu IPOP-CMA-ES, który nazwałem RB-IPOP. Dwie z wprowadzonych modyfikacji czerpią z badań przedstawionych w pracy [A4]. CMA-ES jest jednym z ważniejszych i ciekawszych algorytmów ewolucyjnych. Aby jednak poprawić jego własności na problemach wielomodalnych, zaproponowano wprowadzenie mechanizmów restartu procesu optymalizacji. Jednym z algorytmów wykorzystujących restarty jest IPOP-CMA-ES, w którym, po wykryciu stagnacji, następuje restart procesu optymalizacji ze zwiększonym rozmiarem populacji. Algorytm ten zwyciężył podczas konkursu optymalizacyjnego CEC 2005. Z pracy [A4] wiadomo, że środek populacji jest lepszym aproksymatorem optimum niż najlepszy punkt z populacji. W CMA-ES środek jest wyznaczany przez algorytm w każdej iteracji, nie jest jednak dla niego obliczana wartość funkcji celu, a używany jest jedynie do wytworzenia populacji mutantów. W RB-IPOP zaproponowałem wykorzystanie środka do aktualizacji najlepszego dotąd znalezionego rozwiązania, jak również do zdefiniowania nowych

wyzwalaczy restartu. Na podstawie własnego doświadczenia, popartego wynikami ograniczonych w zakresie eksperymentów, znacząco zmieniłem również niektóre parametry: liczba osobników populacji potomnej (λ) została zwiększona czterokrotnie, a początkowa wartość siły mutacji (σ) została zwiększona siedmiokrotnie. Wartości te zostały dobrane z myślą o starcie w konkursie optymalizacyjnym CEC 2017. Po uruchomieniu, IPOP generuje losowy punkt startowy, a podczas restartu zaczyna od ostatniego środka populacji. Może to spowodować uruchomienie i restarty optymalizacji w bardzo słabym optimum lokalnym. Aby tego uniknąć, RB-IPOP rozpoczyna optymalizację od najlepszego ze 100 losowo wygenerowanych punktów, a restarty odbywają się z użyciem losowo wygenerowanego punktu dla każdego z restartów. Znacząco zmodyfikowałem wartości stałych występujących w istniejących strategiach wykrywania stagnacji algorytmu. Dodałem nową strategię, która co pewną liczbę iteracji dokonuje oceny jakości środka populacji. Jeżeli dwie kolejne oceny różnią się mniej niż 10^{-8} to wykonywany jest restart.

Na podstawie wyników badań eksperymentalnych stwierdziłem, że RB-IPOP nie był nigdy gorszy niż pozostałe przebadane algorytmy. Był lepszy od swojej wersji nie używającej punktu środkowego w 9 przypadkach. RB-IPOP był lepszy od IPOP na 29 funkcjach, niezależnie od wymiarowości zadania. Na podstawie pozostałych eksperymentów zauważyłem, że przewaga RB-IPOP rośnie wraz ze wzrostem wymiarowości zadania, a wprowadzone ulepszenia nie pogarszają istotnie wydajności algorytmu.

RB-IPOP został zaprezentowany w pracy [A5]. Zajął on siódme miejsce (na 12) w konkursie optymalizacyjnym związanym z konferencją międzynarodową (CEC 2017). Wiele późniejszych, zewnętrznych badań pokazuje, że algorytm ten dla pewnego podzbioru funkcji używanych w CEC nie ma sobie równych, bardzo dobrze wygląda również jego szybkość zbieżności. Niestety nie było to brane pod uwagę we wspomnianym konkursie.

NL-SHADE-RSP-MID

W ramach badań związanych z ulepszaniem istniejących algorytmów zaproponowałem szereg modyfikacji algorytmu NL-SHADE-RSP, który jest rozwinięciem algorytmu ewolucji różnicowej (DE). Wybrałem go jako algorytm bazowy ze względu na to, że wygrał on konkurs optymalizacyjny CEC 2021 dla funkcji, których wykresy są przesunięte lub obrócone względem środka układu współrzędnych. Pierwsza z wprowadzonych modyfikacji tego algorytmu polega na wykorzystaniu punktu środkowego, czyli jest związana z wynikami prac opisanymi w artykule [A4]. W każdej iteracji algorytmu wyznaczany oraz oceniany jest punkt środkowy populacji. Przekonanie, że punkt ten powinien wpływać na zawartość bieżącej populacji, wymusiło przeprowadzenie badań, w wyniku których okazało się, że najlepsze wyniki uzyskuje się zastępując punktem środkowym takiego osobnika, który znajduje się najbliżej w przestrzeni przeszukiwań, a jednocześnie nie jest od środka lepszy. Drugi z dodanych komponentów wprowadza mechanizm restartu. Do detekcji stagnacji algorytm wykorzystuje analizę zachowania środka populacji. W wyniku badań stwierdziłem, że jeśli odległość pomiędzy aktualnym środkiem, a środkiem populacji sprzed dziewięciu iteracji, jest poniżej wyspecyfikowanego progu, to algorytm powinien być restartowany. W wyniku restartu algorytm traci całą zdobytą wiedzę i zaczyna optymalizację od nowej populacji startowej. Drugi zaproponowany wyzwalacz restartu oparty jest na wiedzy, że w problemach CEC 2022 rozwiązanie nie leży na ograniczeniach przestrzeni. Jeśli zatem choć jedna współrzędna dowolnego osobnika jest równa wartości ograniczenia dłużej niż przez wyspecyfikowaną liczbę iteracji, to wyzwalany jest restart. W zaproponowanej wersji rozmiar populacji po restarcie zwiększany jest do 400 osobników. Rozmiar ten szybko maleje w funkcji liczby użytych iteracji, dążąc do 20 osobników. Do wyznaczania tego

rozmiaru zaproponowałem użycie odpowiednio sparametryzowanej funkcji homograficznej. Trzecią zmianą wprowadzoną do algorytmu jest zmiana sposobu uwzględniania ograniczeń kostkowych, co ma związek z wynikami badań [A6]. Zastosowałem *resampling*, który polega na powtórzeniu mutacji w razie wygenerowania rozwiązania poza ograniczeniami. Do maksymalnie 10 prób mutacja jest powtarzana bez zmiany współczynnika skalującego i prawdopodobieństwa krzyżowania, w kolejnych maksymalnie 90 próbach oba wymienione parametry są zmieniane. Gdyby zdarzyło się, że 100 prób nie wystarczyło do wygenerowania akceptowalnego rozwiązania to algorytm wraca do zastosowania metody domyślnej, czyli reinicjacji. Kolejna modyfikacja wiąże się z zastosowaniem algorytmu k-średnich. Populacja algorytmu ewolucji różnicowej może przez pewien czas przebywać w obszarach przyciągania różnych ekstremów funkcji celu. W takich przypadkach wyznaczenie punktu środkowego całej populacji nie doprowadzi do znalezienia lepszego aproksymatora żadnego z tych ekstremów. W związku z tym zaproponowałem użycie algorytmu k-średnich do wykonania podziału populacji na grupy osobników bliskich sobie w przestrzeni rozwiązań. Badania wykazały, że tworzenie więcej niż 2 grup nie powoduje poprawienia wyników, natomiast spowalnia algorytm. Aby niepotrzebnie nie zmniejszać dostępnego budżetu liczby ewaluacji funkcji celu, tylko grupowania powyżej określonej jakości są brane pod uwagę. Do oceny jakości grupowania wykorzystałem wskaźnik zwany szerokością sylwetki. Grupowanie nie jest przeprowadzane kiedy rozmiar populacji zmaleje poniżej określonego progu. Parametry wprowadzonych komponentów dobrałem na podstawie wiedzy eksperckiej oraz badań eksperymentalnych. Wykazałem empirycznie, że każdy z wprowadzonych komponentów zwiększał liczbę punktów rankingowych wyliczanych na podstawie procedury zdefiniowanej na potrzeby konkursu CEC 2022.

Zaproponowany algorytm został zaprezentowany w pracy [A10]. Algorytm ten zajął 3 miejsce (na 13) w międzynarodowym konkursie optymalizacyjnym CEC 2022. Stosując się do własnych zaleceń przedstawionych w artykule [A9], kod źródłowy NL-SHADE-RSP-MID udostępniłem online: <https://staff.elka.pw.edu.pl/%7Eerbiedrzy/publ/nl-shade-rsp-mid.zip>.

4.4 Praktyczne zastosowania algorytmów ewolucyjnych

Jak wspominałem wcześniej w moich badaniach analiza i rozwój algorytmów ewolucyjnych przeplata się z ich stosowaniem do rozwiązywania praktycznych problemów. Rozwiązywanie to należy zacząć od odpowiedniego zdefiniowania funkcji celu. Potrzebny jest też model optymalizowanego systemu charakteryzujący się niewielkim kosztem obliczeniowym. Należy również wybrać metodę optymalizacji, co jest trudne ze względu na istnienie ogromnej liczby publikacji, w których autorzy wykazują, że to właśnie ich algorytm jest najlepszy. Wybrany algorytm należy jeszcze skonfigurować. Oprócz ustawienia wartości parametrów, wybrania odpowiednich komponentów algorytmu (np. wariantów operatorów genetycznych oraz wariantów selekcji i sukcesji), należy jeszcze wybrać i zaimplementować odpowiednią do zadania i algorytmu metodę uwzględniania ograniczeń przestrzeni przeszukiwań. Dla wielu problemów rozpoczęcie optymalizacji od losowego punktu startowego nie daje akceptowalnych rozwiązań. Należy wtedy znaleźć rozwiązania startowe w oparciu o wiedzę z danej dziedziny zastosowań, uważając, aby nie doprowadziły one algorytmu do utknięcia w optimum lokalnym. Rozwiązywanie rzeczywistych problemów pozwala również na zbieranie materiałów będących podstawą kolejnych badań, a także na znajdowanie i promowanie praktycznie użytecznych algorytmów.

Optymalizacja struktury zwierciadła SDCM

Badania praktycznej stosowalności algorytmów ewolucyjnych rozpocząłem od rozwiązania problemu zaprojektowania zwierciadła SDCM (ang. Semiconductor Double Chirped Mirror). Zwierciadła tego typu stosowane są w ultraszybkich laserach, które są w stanie generować impulsy o czasie trwania rzędu piko- czy nawet femtosekund. Samo zwierciadło składa się z naprzemiennie ułożonej sekwencji warstw dwóch różnych materiałów optycznych. Na podstawie wiedzy eksperckiej przyjęto, że projektowane zwierciadło powinno składać się ze 120 warstw półprzewodników (GaAs/AlAs). Dodatkowo musi ono zostać pokryte warstwą antyrefleksyjną, która powinna składać się z 3 par warstw $\text{SiO}_2/\text{Si}_3\text{N}_4$. Zwierciadło musi zostać skonstruowane w taki sposób, aby dłuższe fale wchodziły głębiej w strukturę niż krótsze (w zadanym zakresie długości fal), a współczynnik odbicia R powinien być nie gorszy niż 0,999. Celem optymalizacji jest określenie grubości każdej ze 126 warstw.

Praca [A2] zawiera opis sposobu rozwiązania tego problemu, badania użytych metod optymalizacji oraz analizę zgodności wytworzonego zwierciadła z założeniami. W pracy tej zaproponowałem i przeanalizowałem szereg sformułowań funkcji celu. Jako obiecujące metody optymalizacji wybrałem dwa algorytmy z rodziny algorytmów ewolucyjnych: DE i CMA-ES. Jako punkt odniesienia wybrałem klasyczne metody optymalizacji: L-BFGS-B i metodę Neldera-Meada (NM). Pomimo licznych eksperymentów z różną parametryzacją algorytmów, żaden z nich nie znalazł rozwiązania spełniającego wymagania. Przyjąłem, że problemem jest rozpoczynanie przeszukiwania w sposób standardowy, czyli od rozwiązań wylosowanych przy użyciu rozkładu jednostajnego w obszarze dopuszczalnym. Po zapoznaniu się z wiedzą dziedzinową stworzyłem oprogramowanie generujące trzy rodzaje punktów startowych. W pierwszym z nich punkt startowy powstaje przy użyciu najbardziej zaawansowanych modeli, których typowo używa się do wstępnego projektowania tego typu zwierciadeł. Analiza własności tego punktu wykazała, że w strukturze zwierciadła występują niezaplanowane sprzężenia, co powoduje silne oscylacje w GDD. Drugim rodzajem punktu startowego było zwierciadło Bragga zaprojektowane tak, aby odbijało najlepiej środek wymaganego pasma. Trzecim rodzajem punktu startowego było zwierciadło Bragga z liniowym wzrostem grubości warstw (ang. single chirped mirror (SCM)).

Ponieważ DE wymaga różnorodnej populacji początkowej, w celu jej stworzenia dokonałem zaburzenia wybranego punktu startowego przy użyciu rozkładu normalnego. Pomimo zastosowania wstępnie zaprojektowanego zwierciadła jako punktu startowego nadal nie udało się znaleźć rozwiązania algorytmami z domyślnym ustawieniem ich parametrów. Dopiero po zmianie początkowej siły mutacji (σ) w przypadku CMA-ES, a współczynnika CR w przypadku DE, udało się znaleźć rozwiązanie spełniające wymagania. Najlepsze wyniki dał CMA-ES, natomiast DE dał lepsze wyniki niż metody BFGS i NM.

Zbadałem również wpływ sposobu generowania punktu startowego na jakość wyników. Okazało się, że użycie wstępnie zaprojektowanego zwierciadła jako punktu startowego dało wyniki gorsze niż rozpoczęcie optymalizacji od zwykłego zwierciadła Bragga. Najlepsze wyniki dało jednak rozpoczęcie z punktu startowego z liniowym wzrostem grubości warstw.

Podjąłem również próbę uproszczenia zwierciadła tak, aby dało się je łatwiej wytworzyć. Jako punkt startowy przyjąłem najlepsze ze znalezionych rozwiązań, po czym usunąłem z niego 4 warstwy antyrefleksyjne i 10 ostatnich warstw półprzewodnikowych. Używając takiego punktu startowego udało się uzyskać rozwiązanie spełniające ograniczenia. Tak zaprojektowane zwierciadło zostało wytworzone. Porównanie rzeczywistej charakterystyki części półprzewodnikowej zwierciadła z charakterystyką wyznaczoną na podstawie modelu zwierciadła dało dobrą zgodność. Uzyskane

rozwiązanie zostało uznane przez ekspertów z dziedziny za wartościowe z technologicznego punktu widzenia, ponieważ grubości poszczególnych warstw mają zbliżone wartości, co ułatwia wytwarzanie.

Wyznaczenie parametrów rozszerzonego modelu Jilesa-Athertona

Kolejnym praktycznym problemem, do którego rozwiązania zastosowałem odpowiednio dobrany i skonfigurowany algorytm ewolucyjny, było wyznaczenie parametrów rozszerzonego modelu histerezy magnetycznej (rozszerzonego modelu Jilesa-Athertona) na podstawie danych pomiarowych [A3]. Rozszerzony model, oprócz uwzględniania materiałów izotropowych, umożliwia modelowanie materiałów anizotropowych. Parametry tego modelu są jasno związane z fizycznymi własnościami materiałów magnetycznych, nie ma jednak analitycznego sposobu ich wyznaczenia na podstawie pomiarów własności danej próbki. Wyznaczenie takich parametrów modelu, że ich użycie pozwala na uzyskanie takiej pętli histerezy jak pomierzona umożliwia nieniszczącą analizę własności materiałów. W zadaniu postawionym w tej pracy należało wyznaczyć parametry modelu Jilesa-Athertona dla próbek stali nierdzewnej X30Cr13 w dwóch scenariuszach: 1) zakładając, że współczynnik k jest stały; 2) zakładając jego zmienność. Współczynnik ten określa średnią energię potrzebną do przesunięcia ściany domenowej.

We wcześniejszych pracach problem ten był rozwiązywany m.in. przy użyciu strategii ewolucyjnej $ES(\mu + \lambda)$. Wadą tego podejścia była niska wydajność. Jedno uruchomienie optymalizacji zajmowało 32 godziny pracy węzła należącego do klastra obliczeniowego Halo 2. Celem moich badań było znalezienie alternatywnej metody optymalizacji, dającej nie gorsze jakościowo wyniki w krótszym czasie. Jako potencjalnych kandydatów do przebadania wybrałem algorytm CMA-ES oraz algorytm DE w sześciu wariantach, różniących się zastosowaniem różnych metod mutacji.

W rozwiązaniu tego zadania lepiej sprawdził się algorytm DE. Wyniki CMA-ES bardzo silnie zależały od punktu startowego. Spośród zbadanych wersji DE najlepiej sprawdziła się wersja nieklasyczna, nazywana w literaturze local-to-best. Na drugim miejscu była rzadko spotykana wersja z jitter. Pozostałe metody są wyraźnie gorsze.

Zbadałem również wpływ wartości współczynnika krzyżowania CR na wynik. Najlepsza okazała się wartość CR równa jeden, co jest w zgodzie z wynikami opisanymi w artykule [A2]. Wartość ta w praktyce oznacza wyłączenie krzyżowania.

Zaproponowany algorytm jest znacznie szybszy od algorytmu pierwotnego. Obliczenia na procesorze Xeon 3,5 GHz trwały zaledwie 16 min przy stałym k i 26 minut przy zmiennym.

Minimalizacja kosztów pracy dużego magazynu chłodniczego

Kolejnym rzeczywistym zadaniem optymalizacji była minimalizacja kosztów pracy dużych magazynów chłodniczych. W planowaniu badań pomogły wcześniejsze doświadczenia, a zwłaszcza te z pracy [A3]. W jednym z zaproponowanych podejść został wykorzystany algorytm przebadany w [A5]. Nad rozwiązaniem tego zadania pracowałem w ramach projektu finansowanego przez NCBR pt: „Modelowanie predykcyjne i optymalizacja przepływu energii przemysłowych instalacji chłodniczych i mroźniczych”. Chłodnie i mroźnie, ze względu na dużą pojemność cieplną i dużą bezwładność, mogą być wykorzystywane do magazynowania energii w postaci chłodu w czasie, kiedy energia elektryczna jest tańsza. Dzięki temu agregaty można wyłączyć w czasie kiedy cena energii jest wysoka. Na potrzeby realizacji projektu zdefiniowałem funkcję celu oraz sposób uwzględniania ograniczeń kostkowych. Oprócz samego rozwiązania zadania optymalizacji pokazałem trzy

możliwości jego zakodowania oraz porównałem wyniki wielu algorytmów. W pierwszym z zaproponowanych przeze mnie podejść zastosowałem kodowanie dyskretne. Czas został zdyskretyzowany (odstępów półgodzinne). W podejściu tym optymalizator musi zdecydować o włączeniu kompresora w każdym z kwantów czasu. Przy założeniu, że optymalizacja dotyczy harmonogramu dobowego, uzyskuje się zakodowanie rozwiązania o długości 48 bitów. Do rozwiązania tak postawionego problemu zastosowałem szereg algorytmów, w tym PBIL (Probability-Based Incremental Learning) w wielu konfiguracjach oraz algorytmy genetyczne (GA) w różnych konfiguracjach (GA należą do rodziny algorytmów ewolucyjnych). Każdy z algorytmów wymagał ustawienia szeregu parametrów. W przypadku algorytmu genetycznego były to: rozmiar populacji, prawdopodobieństwo mutacji, prawdopodobieństwo krzyżowania. Najlepsze wyniki uzyskałem przy zastosowaniu GA w wersji z selekcją rankingową wykładniczą, krzyżowaniem jednorodnym i sukcesją elitarną. Konfiguracja taka nie występuje w literaturze i nie jest oczywista. W drugim przeze mnie zaproponowanym sposobie rozwiązania problemu zastosowałem algorytmy optymalizacji ciągłej. Do konwersji genotypów zawierających liczby rzeczywiste z zakresu 0-1 na wartości binarne sterujące kompresorem zastosowałem znane z literatury podejście, zwane standardową binaryzacją. W podejściu tym wartość ciągłą traktuje się jak prawdopodobieństwo logicznej prawdy. Zatem sekwencję zer i jedynek otrzymuje się przez próbkowanie zgodnie z tym prawdopodobieństwem. Taki sposób generowania ciągu binarnego występuje też w algorytmie PBIL, jednak w nim, wartości prawdopodobieństwa ulegają deterministycznym poprawkom w zależności od jakości generowanych rozwiązań. Do optymalizacji ciągłej zastosowałem następujące algorytmy: IPOP-CMA-ES (wariant algorytmu adaptacji macierzy kowariancji), DE (ewolucja różnicowa), jSO (potomek DE). Podejście to zawiodło przez zastosowany typ konwersji z przestrzeni ciągłej do przestrzeni dyskretnej — funkcja celu stała się funkcją zaszumioną. W trzecim z zaproponowanych podejść problem również zdefiniowałem jako zadanie optymalizacji ciągłej. Nadal czas jest zdyskretyzowany, natomiast tym razem optymalizacji podlega wartość temperatury zadanej w danym kwancie czasu. Za próbę utrzymania zadanej temperatury odpowiada termostat będący na wyposażeniu chłodni. Mamy zatem zadanie w 48 wymiarowej przestrzeni rzeczywistoliczbowej. Do jego rozwiązania zastosowałem te same algorytmy co w podejściu poprzednim. Wśród nich, najlepsze wyniki dał odpowiednio skonfigurowany algorytm ewolucji różnicowej. Jest to ciekawe, ponieważ znacznie gorsze wyniki dał znany jego potomek (jSO), który posiada wbudowane mechanizmy autoadaptacji większości potrzebnych mu parametrów. Na drugim miejscu uplasował się IPOP-CMA-ES. W DE zastosowałem nieklasyczną wersję mutacji zwaną „local-to-best”. Oczywiście wszystkie algorytmy optymalizacji ciągłej również wymagały ustawienia parametrów. Wyniki wybranych metod porównałem również na trudniejszym problemie powstałym przez dwukrotne skrócenie kroku stosowanego przy dyskretyzacji czasu. Jako punkt odniesienia zaproponowałem heurystykę, która w zdroworozsądkowy sposób wyznacza harmonogram pracy kompresora. Wszystkie zaproponowane przeze mnie podejścia były lepsze niż to znane z literatury. Większość dała wyniki lepsze niż rozwiązanie heurystyczne. Najlepsze wyniki dało dyskretne zakodowanie problemu wraz z odpowiednio skonfigurowanym algorytmem genetycznym.

Wyniki tych badań zostały opublikowane w pracy [A8].

4.5 Pozostałe prace zgodne z tematyką cyklu powiązanych tematycznie artykułów naukowych

Oprócz wcześniej omówionych istnieje jeszcze 11 prac zgodnych z tematyką cyklu powiązanych tematycznie artykułów naukowych. Prace te albo są uzupełnieniem lub rozszerzeniem omówionych wcześniej, albo uznałem je za mniej ważne ze względu na mniejszy wkład w ich powstanie lub mniejszy wpływ na rozwój dziedziny.

W pracach [B1, B3] zastosowałem algorytm ewolucyjny do wyznaczania parametrów rozszerzonego modelu histerezy magnetycznej. Z punktu widzenia tematyki osiągnięcia prace te niewiele wnoszą w stosunku do omówionej wcześniej [A3].

W pracy [B5] odnotowałem istnienie różnic w wynikach uzyskiwanych przez pięć implementacji CMA-ES stworzonych przez autora tego algorytmu. Pracę tę należy traktować jako zapowiedź artykułu [A9].

W pracy [B2] zastosowałem algorytm ewolucyjny do rozwiązania rzeczywistego problemu. Zadanie polegało na zaprojektowaniu optymalnego zestawu kalibracyjnego dla analizatora VNA (ang. Vector-Network-Analyzer). W ramach badań kolejny raz okazało się, że najtrudniej zaprojektować odpowiednią funkcję oceny rozwiązania-kandydata. Wyniki przeprowadzonej optymalizacji ewolucyjnej porównano z wynikami metody wykorzystującej wiedzę o specyfice problemu oraz z oszacowaniem położenia optimum globalnego. Okazało się, że algorytm ewolucyjny zwrócił dobre przybliżenie optimum globalnego uzyskując wyniki lepsze od algorytmu wykorzystującego wiedzę dziedzinową. W pracy tej odpowiadałem za część badań i tekstu związaną z algorytmami ewolucyjnymi.

W pracy [B4] porównałem wyniki sześciu implementacji CMA-ES oraz dwóch implementacji DE dostępnych w języku R. Implementacje te nie zostały wytworzone przez pomysłodawców algorytmów (CMA-ES i DE). Pokazałem, że wyniki optymalizacji uzyskane za pomocą różnych implementacji tego samego algorytmu znacząco się różnią.

Praca [B6] ma charakter przeglądowy. Wskazuję w niej algorytmy z rodziny ewolucyjnej, które mają szansę na uzyskanie dobrych wyników w zastosowaniu do optymalizacji głębokich sieci neuronowych. Określam cechy, jakie powinien mieć dobry algorytm, omawiam konkursy optymalizacyjne skupiające się na tych cechach, ponieważ wśród zwycięzców tych konkursów należy szukać algorytmów nadających się do omawianego zastosowania.

W pracy [B7] porównano wyniki dwóch wariantów algorytmu ewolucyjnego z wynikami solwera MIP w zastosowaniu do projektowania sieci optycznych DWDM. Stwierdzono, że dla dużych (czyli dla praktycznie istotnych) rozmiarów problemów, akceptowalne jakościowo rozwiązania, w akceptowalnym czasie, otrzymuje się jako wynik pracy algorytmów ewolucyjnych. W pracy tej odpowiadałem za badania i opis jednego z zastosowanych algorytmów ewolucyjnych.

W pracy [B8] zastosowano algorytm ewolucyjny do wyliczenia optymalnego położenia pojawiających się dynamicznie kontrolek z podpowiedziami w GUI podczas pracy z aplikacją. W pracy tej odpowiadałem za opracowanie sposobu prezentacji wyników oraz prace nad tekstem, a zwłaszcza opisem algorytmu ewolucyjnego.

W pracy [B9] zaproponowano algorytm hybrydowy będący połączeniem algorytmów DE i CMA-ES. Hybryda ta daje lepsze wyniki niż jej składowe. Miałem niewielki wpływ na kształt tej pracy (około 5% wkładu).

Z podstawowym wariantem algorytmu ewolucyjnego spotkałem się również przed doktoratem. W pracy [B10] zastosowałem algorytm ewolucyjny do minimalizacji liczby przecięć krawędzi w grafie. Wyniki porównałem z wynikami metody deterministycznej.

W pracy [B11] zdefiniowałem przestrzeń drzew decyzyjnych. Do przeszukiwania tej przestrzeni

zastosowałem kilka metod, w tym algorytm ewolucyjny.

Bibliografia (po doktoracie):

- [B1] Rafał Biedrzycki, Roman Szewczyk, Peter Švec Sr., Wojciech Winiarski. *Determination of Jiles-Atherton Model Parameters Using Differential Evolution*. Mechatronics – Ideas for Industrial Application / Awrejcewicz Jan [et al.], Advances in Intelligent Systems and Computing, pp. 11-18, vol. 317, 2015, Springer.
- [B2] Arkadiusz Lewandowski, Wojciech Wiatr, Leszek J. Opalski, Rafał Biedrzycki. *Accuracy and Bandwidth Optimization of the Over-Determined Offset-Short Reflectometer Calibration*. IEEE Transactions on Microwave Theory and Techniques, vol. 63, no. 3, pp. 1076-1089, 2015. IF: 2,943.
- [B3] Dorota Jackiewicz, Roman Szewczyk, Maciej Kachniarz, Rafał Biedrzycki. *Modelling the Influence of Stresses on Magnetic Characteristics of the Elements of the Truss Using Extended Jiles-Atherton Model*. Acta Physica Polonica, A., 2017, Vol. 131, no. 4, pp. 1189-1191.
- [B4] Rafał Biedrzycki. *Differences that make a difference: comparing implementations of selected optimization algorithms in R language*. Proceedings Volume 10808, Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2018; 108081S, 2018.
- [B5] Rafał Biedrzycki. *On equivalence of algorithm's implementations: the CMA-ES algorithm and its five implementations*. GECCO '19 Proceedings of the Genetic and Evolutionary Computation Conference Companion. 2019, pp. 247-248.
- [B6] Rafał Biedrzycki, Paweł Zawistowski, Bartłomiej Twardowski. *Deep Learning Optimization Tasks and Metaheuristic Methods*. Fundamenta Informaticae, vol. 168, no. 2-4, pp. 185-218, 2019.
- [B7] Stanisław Kozdrowski, Rafał Biedrzycki. *Application of evolutionary algorithms to DWDM optical networks design*. Proceedings Volume 11176, Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2019; 111760B, 2019.
- [B8] Wiktor B. Daszczuk, Rafał Biedrzycki, Piotr Wilkin. *Using Evolutionary Algorithm in On-line Deployment*. New Advances in Dependability of Networks and Systems. DepCoS-RELCOMEX 2022. Lecture Notes in Networks and Systems, vol 484. Springer, Cham, pp. 11–21, 2022.
- [B9] Eryk Warchulski, Jarosław Arabas, Rafał Biedrzycki. *Improving the differential evolution strategy by coupling it with CMA-ES*. GECCO '22: Proceedings of the Genetic and Evolutionary Computation Conference Companion, 2022.

Przed doktoratem:

- [B10] Rafał Biedrzycki. *Evolutionary algorithm for minimizing the number of edge crosses in the graph*. Konferencja Algorytmy Ewolucyjne i Optymalizacja Globalna (KAEIOG'05) 2005.
- [B11] Rafał Biedrzycki, Jarosław Arabas. *Evolutionary and greedy exploration of the space of decision trees*. Prace Naukowe Politechniki Warszawskiej, seria Elektronika, zeszyt 156, strony 61-68. Warszawa 2006.

4.6 Opis pozostałej działalności naukowo-badawczej

Oprócz algorytmów ewolucyjnych, zajmowałem się wybranymi metodami maszynowego uczenia się. W pracach [C1, C2] zajmowałem się zastosowaniem klasyfikatorów do fuzji danych. Prowadzone badania były wykonywane w związku z realizacją projektu międzynarodowego pt. „Distributed and Adaptive multisensor Fusion Engine”.

Podczas analizy powszechnie używanych algorytmów klasyfikacji zauważyłem, że mają one w sobie ukryty komponent optymalizacyjny. W pracy [C3] zaproponowałem aby wyeksponować ten komponent, dzięki czemu można by łatwo zmieniać użyte algorytmy optymalizacji. Można by również zmieniać strukturę sąsiedztwa, czy funkcję celu, co dawałoby możliwość prostego ulepszania istniejących metod. Pomysł ten zastosowałem do indukcji drzew decyzyjnych. Kontynuacją tych prac było [C4], gdzie tak indukowany klasyfikator wykorzystywany był do klasyfikacji sekwencji DNA. W [C5] dodatkowo zająłem się konstruktywną indukcją czyli tworzeniem odpowiednich dla klasyfikatora atrybutów. Opracowany sposób postępowania został rozwinięty i przebadany w zastosowaniu do rzeczywistego problemu adnotacji sekwencji DNA [C6].

Bibliografia (po doktoracie):

[C1] Robert Nowak, Jacek Misiurewicz, Rafał Biedrzycki. *Automatic adaptation in classification algorithms fusing data from heterogeneous sensors*. 14th Conference on Information Fusion (FUSION), pp. 1993 - 1999. IEEE, 2011.

[C2] Robert Nowak, Rafał Biedrzycki, Jacek Misiurewicz. *Machine learning methods in data fusion systems*. In 13th International radar symposium (IRS 2012), pp. 400 – 405. IEEE, 2012.

Przed doktoratem, lub pochodne doktoratu:

[C3] Rafał Biedrzycki, Jarosław Arabas. *A Search Based View of Decision Tree Induction*. Artificial Intelligence Studies, pp. 119-127, vol. 3(26)/2006.

[C4] Rafał Biedrzycki. *DNA sequence analysis*. Prace Naukowe Politechniki Warszawskiej, seria Elektronika, zeszyt 160, strony 35-42. Warszawa 2007.

[C5] Rafał Biedrzycki, Jarosław Arabas. *Constructive Induction in Bio-sequences*. In proc. of XVI International Conference on Systems Science, Wrocław, Poland, pp. 449-456, 2007.

[C6] Rafał Biedrzycki, Jarosław Arabas. *KIS: An automated attribute induction method for classification of DNA sequences*. International Journal of Applied Mathematics and Computer Science, vol. 22, no. 3, pp. 711-721, 2012. IF 1,008.

5 Informacja o wykazywaniu się istotną aktywnością naukową albo artystyczną realizowaną w więcej niż jednej uczelni, instytucji naukowej lub instytucji kultury, w szczególności zagranicznej.

- W ramach prac dla Przemysłowego Instytutu Automatyki i Pomiarów (PIAP) badałem stosowalność algorytmów ewolucyjnych do wyznaczania parametrów rozszerzonego modelu histerezy magnetycznej (modelu Jilesa-Athertona). Badania te zakończyły się powstaniem publikacji [A3, B1, B3].

- W ramach prac dla Instytutu Technologii Elektronowej (ITE) zajmowałem się optymalizacją struktury zwierciadła półprzewodnikowego (SDCM). Prace te prowadzone były w latach 2013-2015. Zaprojektowane zwierciadło zostało wytworzone. Badania te zakończyły się powstaniem publikacji [A2] oraz rozdziału pt. „Designing Mirrors for Ultrafast Lasers” w książce „Advances in Evolutionary Algorithms Research” pod redakcją Gregora Papy. Nova Science Publishers Inc, 2015. ISBN: 978-1-63463-849-4.
- Od 2018 roku jestem członkiem komitetu programowego: ENUM — Evolutionary Numerical Optimization Track Programme Committee of the Genetic and Evolutionary Computation Conference (GECCO).

Dalsza zawartość tego rozdziału jest w zgodzie z odpowiedziami udzielonymi przez podsekretarza stanu w Ministerstwie Nauki i Szkolnictwa Wyższego na interpelację nr 10041 z 2020 roku i zawiera informacje o uczestnictwie w pracach zespołów badawczych realizujących projekty finansowane w drodze konkursów krajowych lub zagranicznych, informacje o członkostwie w zespołach badawczych, informacje o działalności recenzenckiej, listę moich wystąpień konferencyjnych i wykaz recenzowanych prac naukowych.

- Uczestniczyłem w pracach zespołów badawczych realizujących projekty finansowane w drodze konkursów krajowych lub zagranicznych:
 - Kierownik w PW w projekcie: „Metody symulacji i analizy sieci logistycznych operatorów pocztowych”. NCBR. 01-12-2019 – 30.08.2022. Skład konsorcjum: SHG, PW, Poczta Polska S.A.
 - Wykonawca w projekcie: „Modelowanie predykcyjne i optymalizacja przepływu energii przemysłowych instalacji chłodniczych i mroźniczych”. 24-01-2019 – 31-10-2019.
 - Wykonawca w projekcie: „Badanie obrazów termicznych skórnej odpowiedzi alergicznej za pomocą algorytmów uczenia maszynowego”. 24-09-2018 – 19-12-2018.
 - Wykonawca w projekcie: „Wstępne badanie obrazów termicznych skórnej odpowiedzi alergicznej za pomocą algorytmów uczenia maszynowego”. 15-06-2018 – 30-07-2018.
 - Wykonawca w projekcie European Defence Agency Projects (EDA): „Distributed and Adaptive multisensor Fusion Engine”, DAFNE, 2009-2011. Skład konsorcjum: IDS - Ingegneria dei Sistemi, Pisa, Włochy (koordynator projektu) UNIUD – Università degli Studi di Udine, Udine, Włochy FGAN-FOM (obecnie Fraunhofer-FOM), Forschungsinstitut für Optronik und Mustererkennung, Ettlingen, Niemcy TNO (Nederlandse Organisatie voor Toegepast Natuurwetenschappelijk Onderzoek) Den Haag, Holandia VOP-026 (Vojenský opravárenský závod) Sternberk, Republika Czeska, oddział w Brnie (VTUO Brno).
 - Byłem kierownikiem i wykonawcą zadania T4.2 w projekcie „Logistyka i technologie monitoringu oraz sposoby ochrony środowiska przed rozpoczęciem prac, w trakcie wiercenia, w procesach szczelinowania hydraulicznego oraz na etapie eksploatacji, w tym monitoring wód podziemnych, powietrza, hałasu, gleby, emisji gazów i innych”, 2013-2016. Skład konsorcjum: Akademia Górniczo-Hutnicza im. Stanisława Staszica w Krakowie (AGH), Instytut Nafty i Gazu, Polskie Górnictwo Naftowe i Gazownictwo S.A. (PGNiG), Politechnika Warszawska (PW).

- Wykonawca w projekcie: „Uogólniona metodyka kalibracji wektorowego analizatora obwodów oparta na statystycznym wieloczęstotliwościowym ujęciu problemu kalibracyjnego”, 2015-2015, NCN.
- Jestem członkiem zespołów badawczych:
 - Zespół Metaheurystycznych Metod Optymalizacji i ich Zastosowań.
 - Zespół Analizy Danych Metodami Sztucznej Inteligencji.
- Jestem regularnie recenzentem dla prestiżowych konferencji naukowych:
 - IEEE Congress on Evolutionary Computation (CEC),
 - The Genetic and Evolutionary Computation Conference (GECCO).
- Jestem regularnie recenzentem dla czasopism:
 - Swarm and Evolutionary Computation,
 - Applied Soft Computing Journal.
- Recenzowałem również dla:
 - TRIZ Future 2022,
 - Conference on Methods & Models in Automation & Robotics,
 - Journal of Computational Science,
 - Journal of Engineering Optimization,
 - The Computer Journal.
- Zespół, w skład którego wchodzi naukowcy z różnych instytutów PW (w tym ja) oraz naukowcy spoza PW, otrzymał nagrodę zespołową I stopnia JM Rektora PW za osiągnięcia naukowe w latach 2017-2018. Skład zespołu Roman Szewczyk (WM/IMIB), Adam Bieńkowski (WM/IMIB), Jacek Salach (WM/IMIB), Maciej Kachniarz (WM/IMIB), Paweł Tomasz Nowak (WM/IMIB), Dorota Jackiewicz (WM/IMIB), Piotr Gazda (WM/IMIB), Andrzej Grzegorz Juś (WM/IMIB), Tomasz Charubin (WM/IMIB), Oleg Petruk (Sieć Badawcza Łukasiewicz, PIAP), Włodzimierz Strupiński (WF/ZBS), Tymoteusz Ciuk (Sieć Badawcza Łukasiewicz), Kinga Kołakowska (WM), Michał Urbański (WM/IMIB), Peter Råback (CSC - IT Center for Science), Juha Ruokolainen (CSC - IT Center for Science), Krzysztof Gromada (WM/IAR), Michał Nowicki (WM/IMIB), Peng Cheng (Mid Sweden University, Department of Electronics Design), Paweł Rozum (WM/IMIB), Rafał Biedrzycki (WEiTI/II), Jakub Szalatkiewicz (PIAP), Mateusz Kalinowski (PIAP), Juhani Kataja (CSC - IT Center for Science).
- Występowałem na krajowych i międzynarodowych konferencjach naukowych:
 - IEEE Congress on Evolutionary Computation 2022. Padwa, Włochy, 18-23 lipca 2022. Tytuły wystąpienia: „A Version of NL-SHADE-RSP Algorithm with Midpoint for CEC 2022 Single Objective Bound Constrained Problems”.

- IEEE Congress on Evolutionary Computation 2021. Kraków, Poland, 28 VI-1 VI 2021. Tytuły wystąpień: „Comparison with State-of-the-Art: Traps and Pitfalls”, oraz „Compressor Schedule Optimization for a Refrigerated Warehouse Using Metaheuristic Algorithms”.
 - GECCO 2019. The Genetic and Evolutionary Computation Conference. Prague, Czech Republic, 13-17 VII 2019. Tytuł wystąpienia: „On equivalence of algorithm’s implementations: the CMA-ES algorithm and its five implementations”.
 - Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2018, 2018, Wilga, Poland. Tytuł wystąpienia: „Differences that make a difference: comparing implementations of selected optimization algorithms in R language”.
 - IEEE Congress on Evolutionary Computation 2017. Donostia - San Sebastián, Spain, 5-8 VI 2017. Tytuł wystąpienia: „A Version of IPOP-CMA-ES Algorithm with Midpoint for CEC 2017 Single Objective Bound Constrained Problems”.
 - Konferencja „Metody Heurystyczne i Algorytmy Ewolucyjne” (MHAE). Sulejów, 16-18 IX 2015. Tytuł wystąpienia: „Praktyczne zastosowania algorytmów ewolucyjnych”.
 - 13th International Conference on Parallel Problem Solving from Nature. Ljubljana, Slovenia, 13-17 IX 2014. Tytuł wystąpienia: „Application of Evolutionary Methods to Semiconductor Double-Chirped Mirrors Design”.
 - XVI International Conference on Systems Science, Wrocław, Poland, September 4-6, 2007. Tytuł wystąpienia: „Constructive Induction in Bio-sequences”.
 - X Krajowa Konferencja „Algorytmy Ewolucyjne i Optymalizacja Globalna”. Będlewo, 11-13 VI 2007. Tytuł wystąpienia: „DNA sequence analysis”.
 - International Conference on Artificial Intelligence AI-21’2006. Siedlce, 21-22 IX 2006. Tytuł wystąpienia: „A Search Based View of Decision Tree Induction”.
 - IX Krajowa Konferencja „Algorytmy Ewolucyjne i Optymalizacja Globalna”. Murzasichle, 31 V-2 VI 2006. Tytuł wystąpienia: „Evolutionary and greedy exploration of the space of decision trees”.
 - VIII Krajowa Konferencja „Algorytmy Ewolucyjne i Optymalizacja Globalna”. Korbielew, 30 V-1 VI 2005. Tytuł wystąpienia: „Evolutionary algorithm for minimizing the number of edge crosses in the graph”.
 - VI Krajowa Konferencja „Algorytmy Ewolucyjne i Optymalizacja Globalna”. Łągów Lubuski, 26-29 V 2003. Tytuł wystąpienia: „Evolutionary algorithm for minimizing the number of edge crosses in the graph”.
 - IV krajowa konferencja „Metody i systemy komputerowe w badaniach naukowych i projektowaniu inżynierskim”. Kraków, 26-28 XI 2003. Tytuł wystąpienia: „Wykorzystanie narzędzia CASE do modelowania projektu RAD”.
- Jestem autorem recenzowanych prac naukowych (w tym 7 prac realizowanych w zespołach z różnych jednostek naukowych):
 - Rafał Biedrzycki, Jarosław Arabas, Eryk Warchulski: A Version of NL-SHADE-RSP Algorithm with Midpoint for CEC 2022 Single Objective Bound Constrained Problems, CEC 2022, 18-23 July 2022 in Padua, Italy.

- Eryk Warchulski, Jarosław Arabas, Rafał Biedrzycki: Improving the differential evolution strategy by coupling it with CMA-ES, GECCO '22: Proceedings of the Genetic and Evolutionary Computation Conference Companion. July 2022. Pages 407–410. DOI:10.1145/3520304.3529023.
- Wiktor B. Daszczuk, Rafał Biedrzycki, Piotr Wilkin: Using Evolutionary Algorithm in On-line Deployment. In: New Advances in Dependability of Networks and Systems. DepCoS-RELCOMEX 2022. Lecture Notes in Networks and Systems, vol 484. Springer, Cham, pp. 11–21. 27 May 2022.
- Rafał Biedrzycki: Comparison with State-of-the-Art: Traps and Pitfalls. 2021 IEEE Congress on Evolutionary Computation (CEC). Kraków, Poland, 28 VI-1 VII 2021.
- Rafał Biedrzycki, Kamil Kwiatkowski, Paweł Cichosz: Compressor Schedule Optimization for a Refrigerated Warehouse Using Metaheuristic Algorithms. 2021 IEEE Congress on Evolutionary Computation (CEC). Kraków, Poland, 28 VI-1 VII 2021.
- Rafał Biedrzycki: Handling bound constraints in CMA-ES: An experimental study, Swarm and Evolutionary Computation, vol. 52, no. 100627, 2020, ISSN:2210-6502, DOI: 10.1016/j.swevo.2019.100627.
- Stanisław Kozdrowski, Rafał Biedrzycki: Application of evolutionary algorithms to DWDM optical networks design. Proceedings Volume 11176, Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2019, Wilga, Poland. DOI: 10.1117/12.2536666.
- Rafał Biedrzycki, Paweł Zawistowski, Bartłomiej Twardowski: Deep Learning Optimization Tasks and Metaheuristic Methods, Fundamenta Informaticae, vol. 168, no. 2-4, pp. 185-218, 2019, DOI: 10.3233/FI-2019-1828.
- Rafał Biedrzycki: On equivalence of algorithm's implementations: the CMA-ES algorithm and its five implementations, GECCO '19 Proceedings of the Genetic and Evolutionary Computation Conference Companion, 2019, pp. 247-248, DOI: 10.1145/3319619.3322011.
- Rafał Biedrzycki, Jarosław Arabas, Dariusz Jagodziński: Bound constraints handling in Differential Evolution: An experimental study, Swarm and Evolutionary Computation, vol. 50, no. 100453, 2019, ISSN:2210-6502, DOI: 10.1016/j.swevo.2018.10.004. IF:6.33.
- Rafał Biedrzycki: Differences that make a difference: comparing implementations of selected optimization algorithms in R language. Proceedings Volume 10808, Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2018; 108081S (2018). DOI: 10.1117/12.2501381 Event: Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2018, 2018, Wilga, Poland.
- Rafał Biedrzycki: A Version of IPOP-CMA-ES Algorithm with Midpoint for CEC 2017 Single Objective Bound Constrained Problems. 2017 IEEE Congress on Evolutionary Computation (CEC). Donostia - San Sebastián, Spain, 5-8 VI 2017.
- Dorota Jackiewicz, Roman Szewczyk, Maciej Kachniarz, Rafał Biedrzycki: Modelling the Influence of Stresses on Magnetic Characteristics of the Elements of the Truss Using Extended Jiles-Atherton Model. Acta Physica Polonica, A., 2017, Vol. 131, no. 4. pp. 1189-1191. DOI: 10.12693/AphysPolA.131.1189.

- Jarosław Arabas, Rafał Biedrzycki: Improving evolutionary algorithms in a continuous domain by monitoring the population midpoint, in *IEEE Transactions on Evolutionary Computation*, vol. 21, no. 5, pp. 807-812, Oct. 2017, DOI:10.1109/TEVC.2017.2673962.
- Rafał Biedrzycki, Jarosław Arabas, Michał Szymański: Designing Mirrors for Ultrafast Lasers, in: *Advances in Evolutionary Algorithms Research* (ed: Gregor Papa), Nova Science Publishers Inc, ch. 7, 2015. ISBN: 978-1-63463-849-4.
- Arkadiusz Lewandowski, Wojciech Wiatr, Leszek J. Opalski, Rafał Biedrzycki: Accuracy and Bandwidth Optimization of the Over-Determined Offset-Short Reflectometer Calibration, in: *IEEE Transactions on Microwave Theory and Techniques*, vol. 63, no. 3, pp. 1076-1089, 2015, ISSN 0018-9480, DOI:10.1109/TMTT.2015.2396496. IF: 2,943.
- Rafał Biedrzycki, Roman Szewczyk, Peter Švec Sr., Wojciech Winiarski: Determination of Jiles-Atherton Model Parameters Using Differential Evolution, in: *Mechatronics - Ideas for Industrial Application / Awrejcewicz Jan [et al.]*, *Advances in Intelligent Systems and Computing*, pp. 11-18, vol. 317, 2015, Springer, ISBN 978-3-319-10989-3, DOI:10.1007/978-3-319-10990-9_2.
- Rafał Biedrzycki, Dorota Jackiewicz, Roman Szewczyk. Reliability and Efficiency of Differential Evolution Based Method of Determination of Jiles-Atherton Model Parameters for X30Cr13 Corrosion Resisting Martensitic Steel. *Journal of Automation, Mobile Robotics & Intelligent Systems*, 2014, Vol. 8, no. 4, pp. 63-68. DOI: 10.14313/JAMRIS_4-2014/39.
- Rafał Biedrzycki, Jarosław Arabas, Agata Jasik, Michał Szymański, Paweł Wnuk, Piotr Wasylczyk, Anna Wójcik-Jedlińska. Application of Evolutionary Methods to Semiconductor Double-Chirped Mirrors Design. In *Parallel Problem Solving from Nature – PPSN XIII*, 2014, Vol. 8672, pp. 761-770. ISBN 978-3-319-10761-5, DOI:10.1007/978-3-319-10762-2_75.
- Jarosław Arabas, Rafał Biedrzycki. Quasi-Stability of Real Coded Finite Populations. In *Parallel Problem Solving from Nature – PPSN XIII*, 2014, Vol. 8672, pp. 872-881. Print ISBN 978-3-319-10761-5, DOI:10.1007/978-3-319-10762-2_86.
- Rafał Biedrzycki, Jarosław Arabas. KIS: An automated attribute induction method for classification of DNA sequences. *International Journal of Applied Mathematics and Computer Science*, 2012, Vol. 22, No. 3, pp. 711-721. ISSN 1641-876X, DOI:10.2478/v10006-012-0053-2. IF: 1,008.
- Robert Nowak, Rafał Biedrzycki, Jacek Misiurewicz. Machine learning methods in data fusion systems. In *13th International radar symposium (IRS 2012)*, Warsaw, Poland, 23 – 25 May 2012, pp. 400 – 405. IEEE, 2012. ISBN 978-1-4577-1835-9.
- Robert Nowak, Jacek Misiurewicz, Rafał Biedrzycki. Automatic adaptation in classification algorithms fusing data from heterogeneous sensors. In *14th Conference on Information Fusion (FUSION)*, Chicago, USA, 5 - 8 July 2011, pp. 1993 - 1999. IEEE, 2011. ISBN 978-0-9824438-3-5.
- Rafał Biedrzycki, Jarosław Arabas. Constructive Induction in Bio-sequences. In *proc. of XVI International Conference on Systems Science*, Wrocław, Poland, September 4-6 2007, v. 3, pp. 449-456.

- Rafał Biedrzycki. DNA sequence analysis. Prace Naukowe Politechniki Warszawskiej, seria Elektronika, zeszyt 160, strony 35-42. Warszawa 2007.
- Rafał Biedrzycki, Jarosław Arabas. A Search Based View of Decision Tree Induction. In proc. of Artificial Intelligence Studies, Vol. 3,(26)/2006, pp. 119-127.
- Rafał Biedrzycki, Jarosław Arabas. Evolutionary and greedy exploration of the space of decision trees. Prace Naukowe Politechniki Warszawskiej, seria Elektronika, zeszyt 156, strony 61-68. Warszawa 2006.
- Rafał Biedrzycki. Evolutionary algorithm for minimizing the number of edge crosses in the graph. In Proc. of KAEIOG'05, Korbiewów, 30 V - 1 VI 2005, ss. 15-22.
- Rafał Biedrzycki. Wykorzystanie narzędzia CASE do modelowania projektu RAD. Materiały IV Krajowej Konferencji MSK'03, Kraków 26-28 listopada 2003, ss. 793-798.

6 Informacja o osiągnięciach dydaktycznych, organizacyjnych oraz popularyzujących naukę lub sztukę.

6.1 Osiągnięcia dydaktyczne

- Nagroda zespołowa II stopnia JM Rektora PW za osiągnięcia dydaktyczne w latach 2019-2020.
- Byłem członkiem komisji powołanej do ustalenia listy i zakresu przedmiotów z programowania na nowym programie nauczania dla Informatyki.
- Jestem współautorem sylabusów do nowych przedmiotów: Podstawy informatyki i programowania (PIPR), Przeszukiwanie i optymalizacja (POP).
- Opracowałem wykład i laboratorium z przedmiotów „Podstawy programowania” oraz „Programowanie obiektowe”. Na wykładzie posługiwałem się współczesnymi wersjami języków C i C++, wskazując zarówno dobre wzorce jak i typowe błędy.
- Opracowałem 1/3 wykładu i projekt z przedmiotu „Metody ewolucyjne i uczenie się maszyn”. Omawianie nowych zagadnień zaczynałem zawsze od pokazania rzeczywistej potrzeby uzasadniającej ich wprowadzenie.
- Jestem współautorem rozdziału pt. „Przeszukiwanie i optymalizacja” w książce „Sztuczna inteligencja dla inżynierów. Metody ogólne” pod redakcją M. Muraszkiewicza i R. Nowaka. Książka została pomyślana jako pomoc dla studentów studiów inżynierskich i jest zalecaną literaturą uzupełniającą do przedmiotu „Wprowadzenie do sztucznej inteligencji”.
- Pełnię rolę promotora prac magisterskich i inżynierskich.
- Jak dotąd prowadziłem wykłady, ćwiczenia lub laboratorium na kilkunastu przedmiotach prowadzonych na Wydziale Elektroniki i Technik Informatycznych PW, w tym:
 - wykład i laboratorium z przedmiotu „Podstawy programowania” w dwóch wersjach — dla elektroników oraz dla informatyków,

- wykład i laboratorium z przedmiotu „Programowanie obiektowe” w dwóch wersjach — dla elektroników oraz dla informatyków,
 - wykład z przedmiotu „Podstawy sztucznej inteligencji”,
 - 1/3 wykładu i projekt z przedmiotu „Metody ewolucyjne i uczenie się maszyn”,
 - projekt z przedmiotu „Metody bioinformatyki”,
 - projekt z przedmiotu „Metody odkrywania wiedzy”,
 - projekt z przedmiotu „Uczenie się maszyn”,
 - projekt z przedmiotu „Wstęp do algorytmów ewolucyjnych”,
 - projekty lub laboratoria na studiach anglojęzycznych z przedmiotów: „Advanced Artificial Intelligence”, „Artificial Intelligence Fundamentals”, „Object Oriented Programming”.
- Obecnie na Wydziale Elektroniki i Technik Informatycznych PW prowadzę:
 - wykład i ćwiczenia z przedmiotu „Wprowadzenie do sztucznej inteligencji”,
 - projekty z przedmiotu „Zaawansowane programowanie w C++”,
 - projekty z przedmiotu „Uczenie maszynowe”.

6.2 Osiągnięcia organizacyjne

- Od 2018 roku jestem członkiem komitetu programowego: ENUM - Evolutionary Numerical Optimization Track Programme Committee of the Genetic and Evolutionary Computation Conference (GECCO). Jest to konferencja międzynarodowa z grupy A (według CORE).
- Byłem członkiem komitetu naukowego konferencji: TRIZ Future 2022.
- Byłem sekretarzem platformy „Robotyka + AI”. W ramach prac udało nam się zebrać i spisać kompetencje zespołów z całej PW, w zakresie przydatnym do stworzenia robota kompana.
- Byłem sekretarzem komisji powołanych do przeprowadzenia publicznych obron rozpraw doktorskich: Marka Adamka, Grzegorza Jaśkiewicza, Szymona Jabłońskiego, Adama Buraczewskiego, Łukasza Bartnika.
- Byłem kierownikiem w PW w projekcie: „Metody symulacji i analizy sieci logistycznych operatorów pocztowych”. NCBR. 01-12-2019 – 30.08.2022. Skład konsorcjum: SHG, PW, Poczta Polska S.A.
- Byłem kierownikiem odpowiedzialnym za wykonanie zadania T4.2 w projekcie „Logistyka i technologie monitoringu oraz sposoby ochrony środowiska przed rozpoczęciem prac, w trakcie wiercenia, w procesach szczelinowania hydraulicznego oraz na etapie eksploatacji, w tym monitoring wód podziemnych, powietrza, hałasu, gleby, emisji gazów i innych”, 2013-2016. Skład konsorcjum: Akademia Górniczo-Hutnicza im. Stanisława Staszica w Krakowie (AGH), Instytut Nafty i Gazu, Polskie Górnictwo Naftowe i Gazownictwo S.A. (PGNiG), Politechnika Warszawska (PW).
- Jestem regularnie recenzentem dla prestiżowych konferencji naukowych:

- IEEE Congress on Evolutionary Computation (CEC),
 - The Genetic and Evolutionary Computation Conference (GECCO).
- Jestem regularnie recenzentem dla czasopism:
 - Swarm and Evolutionary Computation,
 - Applied Soft Computing Journal.
- Recenzowałem również dla:
 - TRIZ Future 2022,
 - Conference on Methods & Models in Automation & Robotics,
 - Journal of Computational Science,
 - Journal of Engineering Optimization,
 - The Computer Journal.
- Byłem przewodniczącym sesji „Special Session Associated with Competition on Real Parameter Single Objective Bound Const II” podczas konferencji międzynarodowej 2022 IEEE Congress on Evolutionary Computation (CEC).