

**UNIVERSITATEA TEHNICĂ "GHEORGHE ASACHI"
DIN IAȘI**



ANALIZA UNOR TEHNICI DE OPTIMIZARE DE INSPIRAȚIE BIOLOGICĂ

REZUMAT

Ing. Elena Niculina Drăgoi

Conducător de doctorat: prof. dr. ing. Florin Leon

IAȘI, 2025

Cuprins

Rezumat.....	3
1. Introducere.....	4
1.1. Context și semnificație.....	4
1.2. Obiectivele cercetării	5
1.3. Structura tezei.....	6
2. Metaeuristici – stadiu actual al cunoașterii.....	8
2.1. Clasificare și metaeuristici de inspirație biologică.....	8
2.2. Explorare-exploatare.....	8
2.3. Comparatie	9
3. Metodologie	10
3.1. Aspectele analizate	10
3.2. Configurația experimentală.....	11
3.3. Analiza datelor	12
3.4. Alți algoritmi analizați.....	12
4. Analiza impactului distribuției.....	13
4.1. Descriptori statistici	13
4.1.1. Decalajul de fitness (gap)	13
4.1.2. Convergență	15
4.1.3. Diversitatea populației.....	16
4.2. Testarea ipotezelor	17
4.3. Probleme de inginerie	18
4.4. Clasamentul distribuțiilor.....	18
5. Analiza impactului mutației asupra performanței metaeuristicilor	20
5.1. Descriptori statistici.....	20
5.1.1. Diferența de fitness (gap)	20
5.1.2. Convergență	21
5.1.1. Diversitatea populației.....	22
5.2. Testarea ipotezelor	23
5.3. Probleme de inginerie	23
5.4. Clasamentul variantelor de mutație	24
5.5. Strategia de îmbunătățire	25
6. Analiza performanței algoritmilor și aplicațiile acestora pentru probleme din ingineria chimică	27

6.1.	Analiza statistică.....	27
6.2.	Complexitatea algoritmilor	29
6.3.	Probleme de inginerie	29
6.4.	Aplicații din viața reală (procese din ingineria chimică)	30
6.4.1.	Eliminarea <i>clorofilei A</i> din apele uzate.....	31
6.4.2.	Procesul de decolorare a rifaximinei	32
6.4.3.	Decolorarea Eriocrom Negru T folosind cărbune activ și $h\nu/\text{TiO}_2$	32
6.4.4.	Decolorarea lichorului negru din tulpini de porumb folosind cărbune activ și TiO_2/UV	33
6.4.5.	Decolorarea compusului verde de bromcrezol folosind cărbune activ	34
6.4.6.	Predicția temperaturilor de tranziție pentru molecule cu cinci inele derivate din resorcinol	35
6.4.7.	Compostarea nămolului uleios.....	37
7.	Concluzii finale.....	38
7.1.	Concluzii privind obiectivele tezei.....	38
7.2.	Aspecte originale	42
7.3.	Direcții de cercetare.....	44
	Bibliografie selectivă	45
A1.	Anexa 1. Lista de lucrări.....	47

Rezumat

Teza abordează o serie de aspecte specifice algoritmilor metaeuristici și analizează influența funcțiilor de distribuție utilizate pentru generarea de numere aleatoare și a operațiunile de mutație asupra performanței unor algoritmi de inspirație biologică. Principalele **obiective** ale cercetărilor realizate sunt: i) dezvoltarea unui cadru sistematic de analiză a impactului strategiilor de distribuție asupra performanței metaeuristicilor pentru clase diferite de probleme; ii) cuantificarea impactului operatorului de diferențiere specific algoritmului Evoluție Diferențială (Differential Evolution, DE) asupra echilibrului explorare-exploatare și proprietăților de convergență; iii) identificarea unor indicații practice pentru selecția și configurarea metaeuristicilor în funcție de specificul problemelor analizate.

Din punct de vedere **structural**, teza este împărțită în șapte capitole. Capitolul 1 (Introducere) stabilește contextul cercetării și identifică principalele obiective ale tezei. Capitolul 2 oferă o privire de ansamblu a domeniului printr-o analiză a literaturii de specialitate. Teza se axează pe metaeuristicile de inspirație biologică și, urmărind taxonomia biologică, este propusă o clasificare a acestor algoritmi. Capitolul 3 detaliază metodologia folosită, incluzând planul experimental, metricile de performanță și tehnicile de analiză statistică utilizate pentru a investiga influența componentelor dintr-un set de algoritmi. Capitolul 4 analizează impactul funcțiilor de distribuție asupra comportamentului metaeuristicilor considerând două tipuri de probleme: sintetice și ingineresti. Capitolul 5 examinează impactul operațiilor de mutație și cuantifică modul în care acestea influențează explorarea, exploatarea și proprietățile de convergență. Pe baza rezultatelor obținute, se propune o nouă strategie de hibridizare cu performanțe superioare comparativ cu algoritmi de bază (testat pe un nou set de algoritmi). Capitolul 6 integrează informațiile din capitolele 4 și 5 pentru a analiza efectele combinate ale distribuției și mutației asupra performanței algoritmilor analizați și demonstrează eficiența acestora pe o serie de studii de caz din ingineria chimică. Fiecare capitol include o secțiune de concluzii parțiale care centralizează principalele rezultate obținute. Capitolul 7 prezintă concluziile finale, evidențiază realizarea obiectivelor propuse și propune direcții viitoare de cercetare.

Aspectele originale ale tezei sunt reprezentate de (i) analiza detaliată a impactului strategiilor de distribuție și de mutație asupra performanței; (ii) propunerea unui nou mecanism de hibridizare; și (iii) demonstrarea eficacității metaeuristicilor pentru rezolvarea unor probleme complexe din lumea reală.

Rezultatele cercetării care susțin teza sunt reprezentate de 7 articole indexate Web of Science (ISI), 1 articol indexat în alte baze de date (BDI), 1 capitol de carte și 4 comunicări la conferințe internaționale. Totodată o parte din aplicațiile practice detaliate în capitolul 7 au fost realizate în cadrul proiectului *Artificial intelligence based modelling and optimization of a hybrid system developed through symbiosis of multiple wastewater treatment methods*, finanțat de UEFISCDI, PN-III-P4-ID-PCE-2020-1052, număr 58/2021, 2021-2023, director de proiect E.N.Drăgoi

1. Introducere

Problemele de optimizare sunt prezente în toate domeniile științei și ingineriei și variază de la planificarea rutelor și alocarea resurselor, până la învățarea automată și bioinformatică. De exemplu, optimizarea este un instrument fundamental în inginerie fiind aplicată pentru proiectarea, analiza, construcția și operarea diverselor dispozitive (Banga, 2008). Aplicațiile de optimizare structurală, proiectare VLSI sau problemele de alocare implică adesea provocări complexe ce trebuie abordate eficient. Importanța algoritmilor de optimizare este susținută de beneficiile potențiale pe care le pot aduce, fie individual, fie în combinație cu alte sisteme/algoritmi, precum reducerea costurilor, îmbunătățirea performanței și a eficienței. Pe măsură ce complexitatea și dimensionalitatea acestor probleme cresc, strategiile deterministe clasice devin adesea impractice din punct de vedere computațional. Această provocare a condus la dezvoltarea metaeuristicilor – algoritmi independenți de problemă care caută soluții în apropierea optimului în spații complexe.

Din multitudinea de strategii de optimizare, această teză se concentrează pe metaeuristici. Cuvântul „metaeuristică” a fost folosit pentru prima dată de către Fred Glover pentru a indica algoritmii evolutivi de inspirație biologică (Dash and Sahu, 2015). În (Voß et al., 2012), metaeuristica este definită ca „*un proces iterativ care ghidează și modifică operațiile unei euristici pentru a genera soluții eficiente. Euristică poate fi o procedură de nivel înalt sau mic, sau chiar o simplă căutare locală*”. Astfel, metaeuristicile oferă un cadru algoritmic, independent de problemă, capabil să abordeze eficient o gamă largă de probleme de optimizare (Sörensen, 2013). Acestea sunt deosebit de utile atunci când metodele tradiționale întâmpină dificultăți sau nu generează soluții din cauza complexității, non-linearității sau dimensionalității ridicate a unei probleme. Eficiența lor derivă din capacitatea de explorare a spațiilor mari de căutare și de identificare soluții optime/aproape optime într-un timp de execuție rezonabil.

1.1. Context și semnificație

Utilizarea tot mai frecventă a metaeuristicilor din ultimii ani reflectă o schimbare în modul în care se abordează optimizarea, adoptându-se metode diverse pentru a depăși limitările strategiilor clasice. Deși soluțiile obținute cu algoritmi metaeuristici nu sunt întotdeauna optime, ele sunt generate, în general, într-un interval de timp rezonabil, fapt acceptabil pentru multe probleme. Justificarea utilizării metaeuristicilor în optimizare poate fi rezumată prin câteva puncte cheie:

- *Cerințe minime pentru cunoștințe specifice problemei:* Spre deosebire de alte tehnici specializate, metaeuristicile necesită adesea cunoștințe minime despre structura problemei. Astfel, sunt

accesibile de către o arie largă de utilizatori și sunt aplicabile și problemelor fără structură bine definită sau cu formulare matematică dificilă.

- *Nu necesită gradienti*: Spre deosebire de metodele bazate pe gradient, metaeuristicile nu necesită informații privind gradientul funcției obiectiv. Astfel, sunt potrivite pentru probleme în care funcția obiectiv este nediferențiabilă sau discontinuă.
- *Accesibile și ușor de înțeles*: Abordările metaeuristice (în special cele clasice) sunt adesea simple și intuitive, ceea ce le face mai ușor de înțeles și implementat – aspect important pentru aplicații practice unde este necesară înțelegerea logicii de optimizare.
- *Aplicații versatile*: Metaeuristicile pot funcționa independent sau pot fi integrate în abordări hibride combinate cu alte tehnici, ceea ce le face flexibile și potrivite pentru o gamă largă de aplicații (Lameesa et al., 2024).
- *Eficiență*: Deși nu garantează optimul global, metaeuristicile pot găsi soluții bune sau aproape optime într-un timp de execuție rezonabil (Trojovský, 2023), aspect important pentru problemele dificile sau când metodele exacte nu sunt practice din cauza timpului sau resurselor necesare.

În ciuda deceniilor de cercetare și a utilizării pe scară largă, înțelegerea mecanismelor care permit ca anumite metaeuristici să aibă o eficiență crescută în anumite condiții rămâne incompletă. În special, doi factori critici – generarea de numere aleatoare (distribuția soluțiilor în spațiul de căutare) și operațiile de mutație care introduc diversitate – influențează semnificativ performanța algoritmilor, în moduri care nu sunt pe deplin caracterizate. Distribuția soluțiilor determină modul în care metaeuristicile eșantionează spațiul de căutare și modul în care răspund la optimele locale. Strategii de distribuție diferite pot genera modele de convergență variate, în special în spații multimodale sau complexe. Similar, operațiile de mutație care introduc perturbări în soluțiile existente sunt cruciale pentru menținerea diversității populației și prevenirea convergenței premature. Interacțiunea dintre acești factori și impactul lor asupra performanței metaeuristicilor constituie o direcție de cercetare ce poate îmbunătăți înțelegerea teoretică și practică a comportamentului acestor algoritmi.

1.2. Obiectivele cercetării

Algoritmii metaeuristici sunt reprezentați de o serie de pași sau blocuri constructive cu roluri clar definite (de exemplu, inițializare, mutație, încrucișare, selecție). Combinația și ordinea acestor pași identifică fiecare metaheuristică. Această particularitate poate fi interpretată diferit, generând probleme specifice domeniului metaeuristicilor. După cum subliniază (Sörensen, 2013), domeniul metaeuristicilor prezintă o serie de probleme: i) terminologie uneori confuză preluată din sursa de

inspirație; ii) asocierea cuvântului "nou" doar pe baza inspirației, fără o analiză aprofundată; iii) lipsă unei metodologii adecvată pentru evaluarea performanței.

În ciuda literaturii vaste privind metaeuristicile, în înțelegerea mecanismelor lor fundamentale există câteva lacune esențiale. Deși multe studii au comparat empiric diferite metaeuristici pe teste sintetice, analizele sistematice care izolează și cuantifică impactul unor componente specifice sunt limitate. Lipsa unor astfel de analize granulare îngreunează dezvoltarea unor algoritmi mai robuști și eficienți. De asemenea, relația dintre garanțiile teoretice de performanță și eficacitatea practică rămâne slab înțeleasă. Algoritmi cu proprietăți teoretice solide pot avea performanțe slabe în practică, în timp ce pentru unele abordări eficiente lipsește o justificare teoretică.

Scopul acestei teze este de a rezolva o parte din aceste probleme printr-o analiză cuprinzătoare a modului în care strategiile de distribuție și operațiile de mutație influențează performanța metaeuristicilor pentru diverse probleme. Astfel, teza are ca obiective: i) Să dezvolte un cadru sistematic de analiză a impactului strategiilor de distribuție asupra performanței metaeuristicilor pentru clase diferite de probleme; ii) Să cuantifice influența operațiunilor de mutație asupra echilibrului explorare-exploatare și proprietăților de convergență; iii) Să identifice indicații practice pentru selecția și configurarea metaeuristicilor în funcție de specificul problemei. Aceste obiective sunt urmărite prin următoarele întrebări: i) **(Q1)**: Cum pot fi clasificate metaeuristicile pentru a avea o imagine de ansamblu mai bună asupra întregii liste de algoritmi existenți?; ii) **(Q2)**: Cât de semnificativă este distribuția folosită în generarea numerelor aleatoare utilizate de obicei de către metaeuristici?; iii) **(Q3)**: Poate operatorul de diferențiere specific DE să influențeze comportamentul diferitelor metaeuristici?

1.3. Structura tezei

Teza este organizată în jurul întrebărilor de cercetare menționate anterior. Răspunsurile la fiecare întrebare sunt furnizate în capitole separate. Din punct de vedere structural, teza este împărțită în șapte capitole:

Capitolul 1 (Introducere) stabilește contextul cercetării, identifică principalele lacune în cunoașterea actuală, formulează obiectivele de cercetare și evidențiază importanța acestora.

Capitolul 2 oferă o sinteză a domeniului metaeuristicilor, concentrându-se pe fundamentele lor teoretice. Pe baza studiului literaturii de specialitate sunt identificate direcțiile principale ale cercetărilor din domeniul și sunt evidențiate aspectele ce necesită o cercetare suplimentară. Având în vedere multitudinea de algoritmi existenți, teza se axează pe metaeuristici de inspirație biologică. Utilizând similitudinea dintre algoritmi și sursa lor de inspirație, este propusă o clasificare ce urmărește taxonomia biologică. Deși detaliată, datorită dinamicii domeniului (variante noi și aplicații practice sunt publicate frecvent), lista algoritmilor discutați nu poate fi considerată exhaustivă.

Capitolul 3 detaliază metodologia folosită, incluzând planul experimental, metricile de performanță și tehnicile de analiză statistică utilizate pentru a investiga comportamentul componentelor dintr-un set de algoritmi. Planul experimental include impactul a două aspecte importante: generarea de numere aleatorii și mecanismul de introducere a mutațiilor în populație. Pentru a analiza performanța metaeuristicilor de bază și a celor modificate este selectat un set de funcții reprezentative pentru diferite scenarii de optimizare.

Capitolul 4 prezintă o analiză detaliată a modului în care strategiile de distribuție influențează performanța metaeuristicilor pentru probleme diferite (sintetice și ingineresti), identificând aspecte specifice fiecărui caz în parte. Analiza are în vedere algoritmi, funcțiile de test și strategiile de mutație detaliate în capitolul 3, centralizând rezultatele pe distribuții.

Capitolul 5 examinează impactul operațiilor de mutație asupra performanței algoritmilor, cuantificând modul în care diferitele strategii influențează explorarea, exploatarea și proprietățile de convergență. Similar capitolului 4, analiza se bazează pe toate simulările, centralizând rezultatele pe variantele de mutație. Pe baza rezultatelor, se propune o strategie de hibridizare simplă ce îmbunătățește performanța variantelor de bază.

Capitolul 6 integrează informațiile din capitolele 4 și 5 pentru a analiza efectele combinate ale distribuției și mutației asupra comportamentului algoritmilor și a eficienței acestora. Se demonstrează și aplicabilitatea acestor observații pe probleme reale de optimizare, validând utilitatea practică a acestor algoritmi.

Capitolul 7 (concluzii) sintetizează principalele rezultate, articulând implicațiile teoretice și practice. Totodată evidențiază realizarea obiectivelor propuse și propune direcții viitoare de cercetare.

Aspectele originale ale tezei sunt: i) Analiza impactului strategiilor de distribuție și de mutație asupra performanței unui set de metaeuristici; ii) Un nou mecanism de hibridizare ce îmbunătățește performanța metaeuristicilor prin includerea conceptului de diferențiere din algoritmul Evoluție Diferențială (Differential Evolution, DE) Differential Evolution și utilizarea distribuției Weibull; iii) Asocierea metaeuristicilor testate în diferite variante cu aplicații practice din domeniul ingineriei chimice.

Rezultatele cercetării care susțin această teză sunt prezentate în Anexa 1. Anexele 2 - 4 prezintă datele centralizate (după diferite criterii) ce sunt discutate și interpretate în Capitolele 4 - 6. La final sunt prezentate toate referințele utilizate, în ordine alfabetică.

2. Metaeuristici – stadiu actual al cunoașterii

2.1. Clasificare și metaeuristici de inspirație biologică

Analiza literaturii de specialitate din domeniul metaeuristicilor indică pentru ultimul deceniu o creștere exponențială a numărului de publicații. Stabilirea unei clasificări precise pentru metaeuristicele existente este dificilă, mai ales având în vedere multitudinea de criterii ce pot fi aplicate. Deși la prima vedere, algoritmi metaeuristici par a fi destul de distincți, majoritatea au o serie de caracteristici comune (Xu and Xu, 2021): i) au o sursă de inspirație; ii) utilizează numere aleatoare în procesul de căutare; iii) nu se bazează pe gradienti; și iv) au o serie de parametri care trebuie ajustați la caracteristicile problemei care trebuie rezolvată. Din grupul metaeuristicilor inspirate din natură, cea mai răspândită sursă este biologia (animale, plante, etc.). Sursele biologice cuprind animale, plante și oameni, în timp ce principiile chimice și fizice ale naturii reprezintă surse non-biologice. (Molina et al., 2020), clasifică strategiile de inspirație biologică sunt clasificate în: i) evoluție bazată pe reproducere, ii) inteligența roiului, iii) bazate pe plante și iv) comportamente sociale.

Având în vedere obiectivele acestei teze, acest capitolul se concentrează pe analiza metaeuristicilor de inspirație biologică și propune o strategie de clasificare a algoritmilor pe baza sursei de inspirație, urmând taxonomia biologică. Clasificarea propusă în teză urmează direcția colorată cu verde din Figura 2-1.

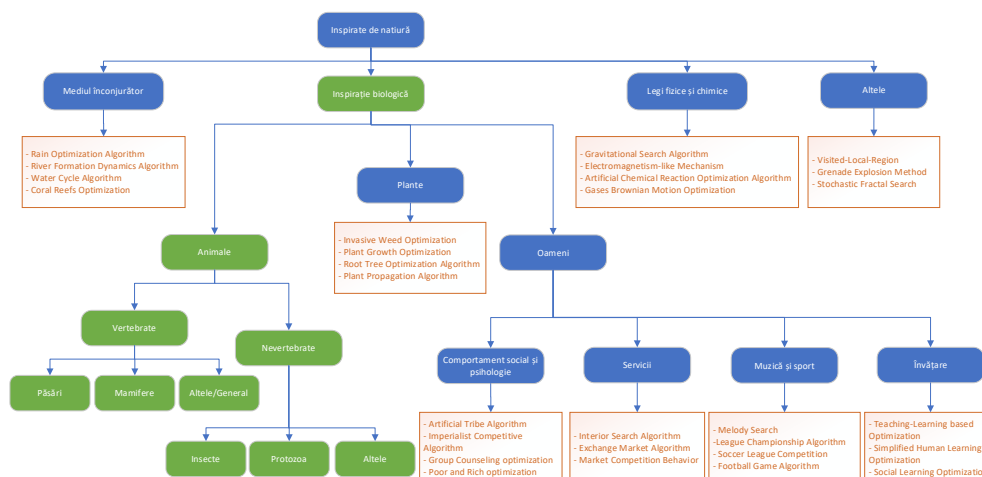


Figura 2-1. Clasificarea metaeuristicilor considerând sursa lor de inspirație

2.2. Explorare-exploatare

În ciuda diferitelor concepte considerate, procesele utilizate pentru a replica comportamentele de optimizare au elemente comune. Algoritmii încep cu o populație inițială de soluții generate aleatoriu. Acesta evoluează printr-o succesiune de procese de căutare până când este îndeplinit un criteriu de oprire. Metodele de realizare a acestei faze de evoluție și mecanismele utilizate pentru a regla echilibrul explorare-exploatare (EEB) afectează performanță și constituie principalii factori care diferențiază

algoritmii. Explorarea este definită informal ca procesul de vizitare completă a spațiului de căutare și este denumită capacitatea de căutare globală a algoritmului. Exploatarea implică navigarea în vecinătăți ale punctelor explorate anterior și indică capacitatea de căutare locală (Črepinšek et al., 2013).

În general, EEB este corelat cu diversitatea populației: când acesta este ridicat, algoritmul este explorator, iar când este scăzut, comportamentul este exploator (Salcedo-Sanz, 2016). Majoritatea lucrărilor care se concentrează pe analiza EEB s-au axat pe propunerea de metrici pentru cuantificarea nivelului de explorare și exploatare folosind diversitatea populației (Morales-Castañeda et al., 2020). Metodele aplicate pentru a atinge EEB pot fi clasificate în (Salleh et al., 2018): i) adaptive; și ii) hibride. În (Črepinšek et al., 2013), abordările bazate pe diversitate aplicate pentru EEB sunt clasificate: (i) menținere a diversității; (ii) control al diversității; (iii) învățare a diversității; iv) alte abordări directe.

2.3. Comparație

Deși au unele componente comune, metaeuristicele utilizează strategii fundamentale diferite pentru a explora spațiul de căutare, făcând comparațiile directe inerent complexe (Silberholz and Golden, 2010). Astfel, pentru a clasifica corect algoritmii din punct de vedere al performanței, trebuie definite criterii de evaluare standard, riguroase din punct de vedere matematic. Atunci când se compară metaeuristicile, principalele aspecte care trebuie luate în considerare sunt: i) teste și selectarea instanțelor, ii) metrici de performanță și calitate a soluției, iii) strategii de planificare a experimentelor și iv) analiza statistică a rezultatelor.

În general, pentru a evalua algoritmii metaeuristici, se aplică o varietate de funcții de test sintetice. În faza de planificare a experimentelor, toate caracteristicile considerate sunt combinate într-un plan logic pentru efectuarea simulărilor. Alte caracteristici care trebuie stabilite în timpul planificării experimentului includ: i) setarea parametrilor și ii) criteriul de oprire. În ceea ce privește setarea parametrilor, în teză, sunt testate două tipuri de algoritmi: statici și auto-adaptivi. În ceea ce privește condiția de stop, criteriul cel mai utilizat este numărul de evaluări ale funcției de fitness (FES) sau numărul de generații (GEN). Când se compară două sau mai multe metaeuristici, urmând planificarea experimentală, sunt efectuate rulări multiple, iar metricile necesare sunt înregistrate (caz replicat). Apoi, problema constă în identificarea celei mai bune strategii potrivite pentru o anumită problemă/set de probleme. În acest caz trebuie aplicate teste inferențiale. În funcție de distribuția datelor, există două tipuri de teste inferențiale statistice: parametrice și neparametrice. *Testele parametrice* presupun că datele urmează o distribuție specifică (de obicei normală) și necesită măsurători la nivel de interval/raport. *Testele neparametrice* nu fac ipoteze despre distribuțiile datelor, analizând în schimb rangurile sau medianele.

3. Metodologie

Pentru a răspunde la principalele obiective, teza aplică o serie de pași care asigură reproductibilitatea și acuratețea rezultatelor. Aceștia includ: i) un plan adecvat pentru studiul experimental (instanțe de probleme, setarea parametrilor, condiția de oprire, numărul de instanțe/execuții și protocolul experimental); ii) aplicarea unui set clar de metrice de performanță; și iii) utilizarea testelor statistice (testarea ipotezelor).

Figura 3-1 exemplifică modul de aplicare al acestor pași. Primul pas se concentrează pe stabilirea principalelor obiective ale studiului, iar teza își propune să răspundă la Q1, Q2 și Q3 detaliate în Secțiunea 1. În acest scop, mai întâi a fost efectuată o analiză detaliată a literaturii și au fost identificați principalii parametri care ar putea influența Q1, Q2 și Q3. După identificarea principalelor caracteristici care vor fi analizate, sunt efectuate rulările efective ale algoritmilor, iar datele sunt colectate și pre-procesate. Ultimul pas corespunde analizei rezultatelor și interpretarea acestora.

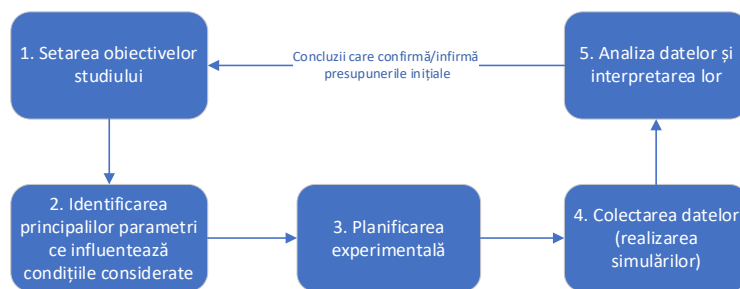


Figura 3-1. Abordarea metodologică a analizei efectuate

3.1. Aspectele analizate

Deoarece domeniul metaeuristicilor este vast, iar în literatură pot fi întâlnite diferite strategii și surse de inspirație, teza se concentrează pe strategii cu un mecanism precis de mutație care poate fi ușor adaptat fără a modifica substanțial principiul de funcționare. Algoritmii selectați sunt: i) Differential Evolution -DE- în forma sa inițială propusă de (Storn and Price, 1997). Această variantă va fi numită *OriginalDE*; ii) Simple Self Adaptive Differential Evolution (*SSADE*); iii) Differential Evolution with Self-Adaptive Populations (*SAPDE*) (Teo, 2006) iv) Succes-History Adaptation DE (Tanabe and Fukunaga, 2013). Această variantă a fost denumită SHADE; totuși, datorită modificărilor efectuate, va fi referită în continuare ca *DevSHADE*; v) Linear Population Size Reduction Success-History Adaptation Differential Evolution (Tanabe and Fukunaga, 2014). Această variantă va fi denumită *LSHADE*; vi) Optimizatorul Satin Bowerbird (Moosavi and Bardsiri, 2017), în forma sa inițială: *OriginalSBO* și într-o formă modificată *DevSBO*; și viii) Differential Search (DS) (Civicioglu, 2012), denumită *DevDS*.

În faza de Planificare Experimentală, conceptul propus în (Achary et al., 2022) ce se bazează pe M (metaeuristica), S (set de operatori), și A (set de algoritmi) este utilizat pentru a determina principalele caracteristici care pot fi variate. Teza s-a concentrat pe două componente ale lui S : i) metode de generare (inițializare) și mai precis distribuțiile utilizate pentru a genera soluțiile inițiale și numerele aleatorii din cadrul pașilor interni ai algoritmilor, și ii) mecanisme de actualizare. Teza examinează cinci distribuții: una este discretă (binomială), iar patru sunt continue (beta, χ^2 , Rayleigh și Weibull). Pentru a determina impactul mutației asupra performanței diferitelor metaeuristici, au fost selectați o serie de algoritmi, iar mecanismele specifice de introducere a diversității au fost modificate pentru a include variantele de diferențiere propuse în algoritmul DE. Aceste variante de diferențiere sunt: M0: rand/1, M1: best/1, M2: best/2, M3: rand/2, M4: current-to-best/1, M5: current-to-rand/1, M6: aleator între variantele M0-M5. Aceste variante de mutație au fost utilizate în două cazuri: i) pentru a înlocui unul dintre pașii existenți; ii) pentru a acționa ca un mecanism suplimentar pentru a adăuga diversitate populației.

În ceea ce privește funcțiile de test, suita CEC2022 și un set de probleme standard din inginerie sunt utilizate pentru a determina performanța algoritmilor modificați. Problemele ingineriești considerate sunt: ENG0- Proiectarea rețelei de schimbătoare de căldură (cazul 1), ENG1- Proiectarea rețelei de schimbătoare de căldură (cazul 2), ENG2 - Problema de amestecare-acumulare-separare, ENG3- Separarea neselectivă a propanului, izobutanului și n-butanului, ENG4 - Operarea optimă a unității de alchilare, ENG4 - Problema de proiectare a unei rețelei de reactoare.

3.2. Configurația experimentală

Configurația experimentală combină toate aspectele considerate (algoritmi, caracteristicile lor și funcții de testare) urmând un plan clar și concis care asigură că rezultatele sunt consecvente și ușor de analizat. De asemenea, aceleași condiții experimentale sunt stabilite pentru toate simulările pentru a asigura consistența. Pentru funcțiile de test, în conformitate cu competiția CEC2022, au fost testate două dimensiuni: D10 și D20. Dimensiunea populației este setată la 50 în toate cazurile, iar criteriul de oprire este reprezentat de FES, cu o valoare maximă de 50000. În ceea ce privește parametrii de control, pentru fiecare algoritm, sunt utilizate valorile standard recomandate (cu excepția versiunilor adaptive/auto-adaptive, unde acestea variază conform regulilor interne specifice fiecărei versiuni). Metricile analizate au inclus performanța (fitness-ul fiecărei funcții), viteza de convergență (numărul de FES pentru a ajunge la 5% aproape de optim) și diversitatea populației (ca strategie pentru a estima echilibrul explorare-exploatare).

3.3. Analiza datelor

Pentru a asigura validitatea evaluării performanței și a comparației între metaeuristici au fost aplicate tehnicile statistice. Pentru analiza performanței, deoarece fiecare funcție testată are un optim diferit, metrica bazată pe decalaj (*gap*) este utilizată. Teoria testării ipotezelor a fost aplicată pentru a analiza impactul componentelor algoritmului identificate anterior: metode de generare și mecanism de actualizare. Întrebările la care teza își propune să răspundă folosind metoda testării ipotezelor sunt: i) Q2- prezentată în secțiunea de introducere: există un impact asupra comportamentului algoritmului indus de utilizarea diferitelor generatoare de numere aleatoare; și ii) Q3 – prezentată în secțiunea de introducere: cum diferitele strategii de mutație influențează comportamentul algoritmului sunt transformate în următoarea ipoteză: i) H_0 -Q2 (ipoteza nulă pentru Q2): nu există nici o diferență semnificativă statistic între metaeuristica care utilizează diferite generatoare de numere aleatorii; ii) H_1 -Q2 (ipoteza alternativă pentru Q2): există o diferență semnificativă statistic între diferitele generatoare de numere aleatorii; iii) H_0 -Q3 (ipoteza nulă pentru Q3): nu există nicio diferență semnificativă statistic între metaeuristica care utilizează diferite strategii de mutație; iv) H_1 -Q3 (ipoteza alternativă pentru Q3): există o diferență semnificativă în metricile de performanță între diferitele strategii de mutație.

În conformitate cu metricile de performanță selectate și având în vedere o comparație multi-metodă, formulările generale ale ipotezelor prezentate mai sus sunt extinse pentru performanță și convergență. Deoarece tipul de teste care pot fi aplicate trebuie să respecte o serie de ipoteze, primul pas este determinarea dacă datele analizate urmează o distribuție normală. Teza rezolvă această problemă utilizând testul Shapiro-Wilk. Pentru analiza performanței și convergenței, dacă datele sunt distribuite normal, se aplică teste parametrice (test t) pentru a determina dacă ipoteza nulă este respinsă. Dacă datele nu sunt distribuite normal, se aplică testul Kruskal-Wallis. Dacă aceste ipoteze sunt respinse și există diferențe semnificative, se aplică o comparație pereche post-hoc folosind strategia multiplu-test cu ajustarea Bonferroni și este calculată metrica Cohen.

3.4. Alți algoritmi analizați

Analiza datelor a permis identificarea distribuției și mutației care generează performanța cea mai bună pentru majoritatea cazurilor. Astfel, este propusă o abordare hibridă ce este aplicată atât pentru algoritmi menționați anterior, cât și pe o selecție de alte metaeuristici. Noii algoritmi au fost selectați pe baza capacității lor de a accepta strategii de diferențiere și hibridizare fără modificări semnificative în faza de implementare. Acești algoritmi sunt: Coyote Optimization Algorithm (COA) (Pierezan and Coelho, 2018), Northern Goshawk Optimization (NGO) (Dehghani et al., 2021), Naked Mole-Rat Algorithm (NMRA) (Salgotra and Singh, 2019), Sea Lion Optimization (SLO) (Masadeh et al., 2019).

4. Analiza impactului distribuției

Etapa de inițializare stabilește evoluția echilibrului între explorare și exploatare, deoarece soluțiile inițiale distribuite necorespunzător pot duce la convergență prematură sau utilizarea excesivă a resurselor computaționale disponibile. Diversitatea populației inițiale determină dacă algoritmul poate scăpa de optimele locale și poate descoperi optimele globale (Li et al., 2020). Când se inițializează populația, majoritatea cercetătorilor se concentrează pe strategii care îmbunătățesc performanța populației inițiale, mai degrabă decât pe impactul mecanismului utilizat pentru a genera acea soluție. Totodată, în etapele de evoluție a populației sunt folosite diferite mecanisme care utilizează în anumite contexte generatoare de numere aleatorii. Când specificitatea unei anumite operații o cere, sunt aplicate distribuții ce au o anumită semnificație (de exemplu, anumite tipuri de mișcări ale animalelor sunt simulate folosind distribuția Levy). Pe de altă parte, în majoritatea cazurilor, se utilizează distribuția normală. Această secțiune se concentrează pe analiza impactului distribuției utilizate atât în faza de inițializare cât și pentru generarea de numere aleatorii în pașii evolutivi asupra performanței algoritmilor selectați.

Întreaga analiză este efectuată pe setul de date complet, fără etape de preprocesare, cum ar fi eliminarea valorilor extreme sau normalizarea datelor. În situațiile în care metricile analizate utilizează normalizarea sau alte procesări statistice, se specifică acest lucru.

4.1. Descriptori statistici

În conformitate cu metodologia stabilită, după efectuarea simulărilor și colectarea datelor, sunt calculați și analizați un set de indicatori statistici. Setul complet de date include 30 de rulări pentru fiecare combinație distribuție – mutație – algoritm – funcție. La fiecare rulare, pentru fiecare iterație sunt salvate următoarele informații: i) valoarea cea mai bună a funcției de fitness; ii) soluția corespunzătoare celui mai bun fitness; iii) rata de convergență; iv) diversitatea populației.

4.1.1. Decalajul de fitness (gap)

Deoarece impactul fiecărei distribuții poate fi diferit de la algoritm la algoritm, Anexa 2 prezintă indicatorii statistici pentru fiecare caz din perspective multiple. Tabelele A2-1 și A2-2 prezintă statisticile pentru toate distribuțiile pentru D10 și D20 considerând decalajul de fitness. Întregul set de date este centralizat în Figura 4-1. Formele generale ale boxplot-urilor sunt consistente pentru toate cele șase distribuții, sugerând că, deși pot exista diferențe minore în caracteristicile de performanță, niciuna dintre distribuții nu demonstrează performanțe dramatic superioare sau inferioare în comparație cu celelalte.

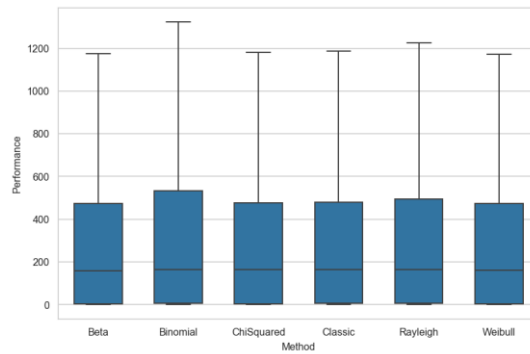


Figura 4-1. Valorile de decalaj centralizate pentru toate distribuțiile și combinațiile testate

Pentru D10, rezultatele indică faptul că distribuția beta are cea mai mică medie a valorilor *gap* pentru funcțiile F2, F7, F8, F9 și F11, sugerând că aceasta produce cei mai favorabili candidați inițiali pentru aceste funcții. Pentru funcțiile F3, F4 și F10, distribuția Weibull pare a fi mai eficientă, în timp ce F5 este cel mai bine gestionată de distribuția chi-squared. În contrast, distribuția clasică funcționează mai bine pentru funcțiile F6 și F12. Pentru D20, distribuția beta are cea mai mică diferență medie pentru F7 și F12, pierzând cel mai bun rang pentru F2, F8, F9 și F11. În general, se poate observa că o creștere a dimensionalității corespunde unei creșteri a valorii medii obținute, un fapt așteptat, în special având în vedere condițiile de simulare, care au limitat FES la aceeași valoare.

Cuartilele oferă informații despre distribuția rezultatelor. Apropierea cuartilei 50 (mediană) de medie în multe cazuri sugerează o distribuție relativ simetrică a performanței în unele cazuri; totuși, abaterile cuartilelor inferioare (5, 10) și superioare (90, 95) de la mediană relevă cât de des apar rezultate extreme. Distribuția beta are adesea valori mai mici pentru toate cuartilele pentru funcții precum F2, F7 și F9, indicând o performanță relativ constantă. Diferența mai mică între cuartilele 5 și 95 sugerează o inițializare mai fiabilă. Deși uneori prezintă valori mediane bune, distribuția Weibull are adesea cuartilele 90 și 95, mai ridicate, în special pentru funcții precum F3 și F4. Distribuția clasică are în general valori moderate pentru toate cuartilele, adesea cu o răspândire echilibrată, sugerând că este o alegere stabilă pentru diverse funcții. Distribuția chi-squared are rezultate deosebit de bune pentru cuartile pentru F5, indicând o performanță bună constantă pentru această funcție. Pentru funcțiile unde o anumită distribuție arată cuartile constant mai mici (de exemplu, chi-squared în cazul F5), ar putea fi avantajoasă utilizarea acesteia pentru inițializare și pentru pașii evolutivi, potențial ducând la o convergență mai rapidă sau soluții finale mai bune.

Metrica *skewness* (gradul de asimetrie) cuantifică oblicitatea formei distribuției performanței. În distribuțiile asimetrice, media poate fi departe de marea majoritate a datelor și deplasată spre coadă. Aceasta înseamnă că media poate să nu reprezinte valoarea "tipică" din setul de date. Asimetria pozitivă poate duce la supraestimarea mediei populației, în timp ce asimetria negativă poate duce la subestimare. În acest context, mediana oferă adesea o măsură mai bună a tendinței centrale. Pentru distribuțiile Rayleigh și beta, luând în considerare dimensionalitatea și algoritmi, Figura 4-4 arată

centralizarea valorilor *gap*. După cum se poate observa, valorile ridicate care duc la o alungire la dreapta sunt introduse de strategiile SAPDE și OriginalSBO. O comparație între variantele de mutație pentru strategia SAPDE a arătat că tipurile de mutație M0, M3 și M5 au introdus cele mai mari erori pentru ambele distribuții, indicând că pentru funcția F8, variația performanței este datorată în principal strategiei de mutație, distribuția utilizată pentru generarea de numere aleatorii având o influență mai redusă.

Valorile *kurtosis* (gradul de aplatizare) oferă informații suplimentare despre tendința pentru valori extreme în distribuție. Valorile mai mari ale gradului de aplatizare sugerează că performanțele aberante apar mai frecvent decât într-o distribuție normală. Aceste caracteristici cu alungiri mari implică faptul că riscul de rezultate ocazionale, divergente, nu este neglijabil chiar dacă performanța medie pare acceptabilă. Rezultatele au arătat că distribuțiile Weibull și beta au un profil al gradului de aplatizare mai jos decât distribuția normală pentru majoritatea combinațiilor.

4.1.2. Convergență

În majoritatea cazurilor, convergența este analizată folosind grafice de convergență în care variația fitness-ului în raport cu FES sau numărul de iterații este vizualizat și comparat manual. Convergența generală centralizată pentru toate funcțiile și algoritmi este prezentată în Figura 4-2, unde datele sunt calculate folosind media tuturor încercărilor și combinațiilor/iterațiilor, iar axa y este pe o scară logaritmică.

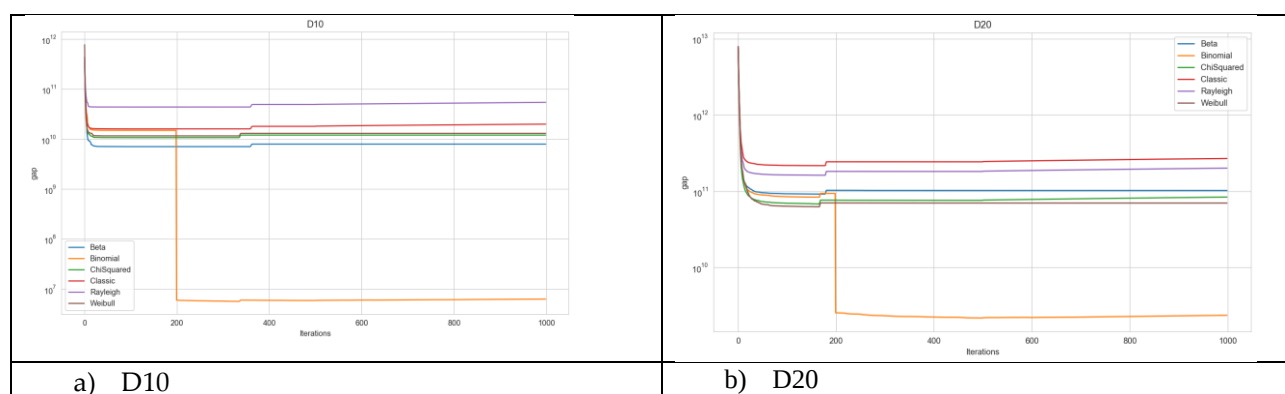


Figura 4-2. Profilele de convergență pentru toate funcțiile și algoritmi testați, centralizate pe baza valorii medii a tuturor simulărilor

În faza de testare, resursele alocate nu sunt consumate identic în toate cazurile, iar rulările sunt oprite când nu există îmbunătățiri pe parcursul a 200 de iterații (corespunzând, în majoritatea cazurilor, la 1/5 din resursele maxim alocate). În consecință, datele disponibile la o anumită FES/iterație ar putea să nu fie identice ca număr, un aspect care poate duce la unele scăderi rapide sau mici creșteri în valorile medii pe baza cărora este realizat graficul. Acest lucru se întâmplă mai ales în etapele finale ale rulării. Astfel, aceste grafice centralizate arată tendințele generale, iar aceste scăderi/creșteri bruște ale ratei de convergență nu indică întotdeauna îmbunătățiri/modificări ale performanței. Din perspectiva

convergenței, deși forma este foarte similară, profilurile specifice fiecărei distribuții pentru D10 și D20 sunt distincte, indicând faptul că impactul acesteia diferă în funcție de dimensionalitate. Se observă o scădere bruscă pentru cazul binomial și o mică creștere pentru toate distribuțiile. Această scădere poate fi explicată prin caracteristica discretă a distribuției binomiale și caracteristica continuă a problemelor testate, utilizarea acesteia ducând la o blocare în minime locale și la o stagnare a celei mai bune soluții. Astfel, algoritmi sunt opriți înainte de a utiliza numărul de FES maxim alocat, pentru anumite iterații, doar pentru un număr redus de rulări existând valori ce pot fi utilizate la calcularea valorii medii. Pe de altă parte, creșterea observată se datorează impactului algoritmului SAPDE, care are o populație adaptivă și nu se oprește întotdeauna la aceeași iterație.

Pentru F10, toate simulările efectuate nu au reușit să atingă 95% din optim. Acest lucru indică faptul că această problemă este dificil de rezolvat pentru toate cazurile considerate, iar aplicarea diferitelor distribuții nu îmbunătățește performanța pentru niciunul dintre algoritmi. Creșterea dimensionalității la D20 a dus la o creștere a numărului de funcții pentru care toate variantele considerate nu au reușit să convergă: F6, F10 și F12.

Majoritatea distribuțiilor prezintă asimetrie pozitivă între dimensiuni, indicând faptul că, deși majoritatea rulărilor converg într-un interval rezonabil pentru valorile FES, există ocazional valori extreme cu cerințe FES foarte ridicate. Valorile ridicate ale gradului de aplatizare sugerează că punctele extreme sunt mai comune comparativ cu distribuția normală, iar acest lucru este evident pentru distribuția clasică. Prezența valorilor extreme sugerează că, criteriile robuste de oprire sau mecanismele de ajustare dinamică ar putea îmbunătăți eficiența algoritmului, evitând evaluări excesive ale funcțiilor dificil de rezolvat.

4.1.3. Diversitatea populației

Diversitatea ridicată permite explorarea extensivă, evitând blocarea în optime locale, în timp ce reducerea diversității promovează exploatarea. O scădere timpurie a diversității indică frecvent convergența prematură, situație în care algoritmi devin blocați în zone suboptime din cauza variației inadecvate sau a presiunii de selecție excesive. În contrast, diversitatea ridicată favorizează îmbunătățirea continuă a soluției, dar poate duce la calcule inefficiente dacă nu este gestionată adecvat. În această secțiune, analiza se concentrează pe metrica. Deoarece diversitatea este măsurată la fiecare iterație, valorile diversității la valorile FES pentru care soluția atinge 95% din optim, sunt mai întâi extrase. Astfel, statisticile privind convergența și diversitatea sunt corelate.

În ceea ce privește valorile medii, pentru D10, distribuția binomială generează valori minime pentru șapte cazuri, urmată de Weibull cu trei cazuri. Cu toate acestea, când se compară cu rata de convergență, se poate observa că, deși distribuția binomială poate menține diversitatea la un nivel mai ridicat comparativ cu celelalte distribuții, rata sa de convergență este scăzută, indicând faptul că are

nevoie de un număr mai mare de FES pentru a atinge un nivel bun de eficiență. Creșterea dimensionalității la D20 duce la o schimbare completă în ceea ce privește cel mai mare număr de victorii, unde distribuția Rayleigh are patru victorii. La fel ca pentru D10, cea mai mare performanță în termeni de convergență nu corespunde neapărat cu cea mai mare diversitate. Astfel, din această perspectivă, se poate concluziona că, deși intuitiv, cea mai mare diversitate corespunde unei performanțe mai bune, acest lucru nu este întotdeauna cazul, iar plasarea indivizilor în spațiul de căutare comparativ cu minimele locale/globale este importantă. De asemenea, valorile diversității pentru D10 sunt mai mici decât pentru D20, indicând faptul că dimensionalitatea spațiului de căutare influențează și diversitatea.

Deviația standard este mare pentru toate distribuțiile, indicând o variabilitate semnificativă în diversitate, cu valori mai mari pentru D20 decât pentru D10. În general, distribuția Weibull prezintă std considerabil mai mari decât beta în ambele dimensiuni, indicând faptul că beta tinde să aibă un comportament mai robust. Toate distribuțiile prezintă asimetrie pozitivă (valori de obicei între 1 și 2.7), indicând distribuții cu alungire la dreapta cu valori extreme ocazional ridicate. Distribuția beta arată în general valori de asimetrie pozitivă mai mici (de obicei 1.0 - 2.0) decât Weibull (adesea 1.5 - 2.7). Weibull prezintă valori ale gradului de aplatizare mai mari, în special pentru D20, indicând cozi mai lungi și mai multe valori extreme. Acest lucru sugerează că Weibull produce distribuții cu un caracter pronunțat non-normal, asimetric spre dreapta ale valorilor diversității.

4.2. Testarea ipotezelor

Pentru *performanță*, analiza corespunde ipotezelor H_0 -Q2-p (toate distribuțiile generează o performanță egală pentru toate combinațiile testate) și H_1 -Q2-p (cel puțin o metodă de distribuție diferă semnificativ în performanță). Urmând metodologia propusă, testul Shapiro este aplicat pentru a determina dacă datele urmează o tendință normală. Rezultatele au indicat că ipoteza normalității a fost respinsă și, prin urmare, abordarea Kruskal-Wallis este aplicată pentru a testa dacă ipoteza nulă este îndeplinită. Testul statistic a avut o valoare p de 0.34, indicând că, din perspectivă statistică, pentru datele centralizate care iau în considerare toți algoritmi, nu există dovezi suficiente pentru a demonstra că diferențele de performanță între distribuțiile testate sunt semnificative.

Deoarece analiza a arătat că comportamentul algoritmilor testați a variat odată cu schimbarea dimensionalității, testele statistice au fost aplicate și pentru fiecare dimensiune. Pentru D10, ipotezele de normalitate nu sunt îndeplinite, iar testul Kruskal-Wallis a rezultat într-o valoare p de 0.041. Acest lucru indică faptul că există perechi de distribuții pentru acest caz particular pentru care diferențele sunt semnificative. Singura pereche cu diferențe statistice semnificative este cazul Beta vs Rayleigh, cu o valoare pentru indicatorul Cohen < 0.2 . În general, se consideră că pentru valori mai mici de 0.2, diferențe dintre perechi sunt marginale. Cu toate acestea, având în vedere numărul mare de combinații

pentru care a fost făcută analiza (8 algoritmi și 6 variante de mutație, fiecare caz rulat de 30 de ori, ceea ce rezultă într-un total de 1440 de rulări pentru fiecare distribuție), această diferență considerată marginală este importantă din punct de vedere practic.

Luând în considerare ambele dimensiuni, niciuna dintre distribuțiile testate (cu excepția beta pentru un singur caz) nu a depășit sau nu a avut performanțe constant mai slabe decât altele. Creșterea dimensionalității de la D10 la D20 reduce și mai mult diferențele observabile între performanțele distribuțiilor, pe măsură ce statisticile t medii se apropie de zero, iar valorile p medii cresc. Aceste rezultate sugerează că alegerea distribuției între beta, binomial, chiSquared, uniformă, Rayleigh și Weibull poate avea un impact practic limitat asupra performanței metaeuristicilor la dimensiuni mai mari, având în vedere absența diferențelor statistice semnificative observate în această analiză.

Pentru *convergență*, analiza se bazează pe valorile FES necesare pentru a atinge 95% din optimul cunoscut. Ipotezele testate sunt H_0 -Q2-c (toate distribuțiile generează profile de convergență similare) și H_1 -Q2-c (cel puțin o distribuție diferă semnificativ în convergență). Statistica Kruskal-Wallis a fost 6.67 pentru datele generale, iar valoarea p a fost 0.27. Când au fost analizate selecțiile corespunzătoare celor două dimensionalități, valoarea p a fost 0.912 pentru D10 și 0.19 pentru D20. Acest lucru indică faptul că nu există dovezi statistice pentru a respinge ipoteza nulă și, în ceea ce privește ratele de convergență, deși apar unele diferențe în cazuri specifice, acestea nu sunt semnificative din punct de vedere statistic. Cu toate acestea, din punct de vedere practic, diferențele obținute indică o importanță practică a funcției de distribuție utilizată.

4.3. Probleme de inginerie

Când se consideră valorile minime pentru ENG0, ENG1, ENG4, ENG5 și ENG6, toate distribuțiile converg către valori minime aproape identice, sugerând că acestea identifică cu succes același minim global. Pentru ENG4, toate distribuțiile ating exact -4529.12, indicând un minim global clar pe care toate abordările îl pot găsi. Cu toate acestea, mediile sunt destul de departe de minime. Acest lucru se datorează mecanismelor utilizate pentru a gestiona constrângerile. Toate problemele au multe constrângeri, iar penalizarea constă în adăugarea la fitness a valorii constrângerii înmulțite cu 10^8 .

4.4. Clasamentul distribuțiilor

Pentru a determina clasamentul distribuțiilor, mai întâi sunt calculați indicatorii SNE , SE și SR folosind ecuațiile 3-153 – 3-157 prezentate în Capitolul 3 considerând două criterii: algoritmul și distribuția. Astfel, este determinată influența distribuției asupra fiecărui algoritm, după care este realizat un clasament general (Tabel 4-1). Rezultatele din această secțiune sunt determinate pe baza

valorilor normalizate aplicate pe întregul set de date. R reprezintă rangul iar S este scorul final pe baza căruia este determinat rangul.

Tabel 4-1. Clasamentul general al distribuțiilor pe baza funcțiilor din competiția CEC

Distribuție	SNE	SE	SR	S	R
Beta	4.73	48.38	46.55	94.93	2
Binomial	5.08	45.00	22.88	67.88	6
χ^2 (ChiSquared)	6.10	37.53	39.71	77.24	3
Uniformă	6.14	37.25	36.99	74.23	4
Rayleigh	5.26	43.49	29.67	73.16	5
Weibull	4.58	50.00	50.00	100.00	1

Distribuția binomială este singura distribuție discretă testată. Distribuțiile discrete sunt potrivite pentru probleme combinatorii sau inerent discrete în care variabilele de decizie iau un număr limitat de valori. Cu toate acestea, problemele testate nu sunt nici combinatorii, nici discrete. Astfel, prin limitarea opțiunilor la un set predefinit de valori atunci când se rezolvă probleme continue, distribuțiile discrete concentrează căutarea și nu permit o explorare adecvată a spațiului.

Conform indicatorilor utilizați în Tabelul 4-1, cea mai bine clasată distribuție este Weibull. În teză, parametrii pentru distribuție au fost setați la (1, 1), corespunzând distribuției exponențiale. Distribuția exponențială este singura distribuție continuă care prezintă caracteristica de lipsă de memorie, semnificând că probabilitatea unui eveniment viitor nu este afectată de rezultatele precedente. A doua cea mai bine clasată distribuție este beta. Spre deosebire de distribuțiile chiSquared, Rayleigh sau Weibull — definite pe intervale semi-infinite — distribuția beta este definită strict într-un interval mărginit. Această natură mărginită este avantajoasă pentru inițializare, deoarece restricționează în mod inerent soluțiile inițiale în spații de căutare fezabile, reducând probabilitatea de generare a valorilor extreme sau nefezabile care pot afecta negativ convergența. Aceeași indicatori calculați în Tabelul 4-3 pe baza rezultatelor obținute pe funcțiile CEC 2022 sunt determinați și pentru problemele de inginerie. În acest caz distribuția Weibull își păstrează rangul, în timp ce ordinea pentru celelalte distribuții se schimbă semnificativ.

5. Analiza impactului mutației asupra performanței metaeuristicilor

Mecanismele de actualizare (modificare) a indivizilor din populație sunt folosite pentru a evolua soluțiile potențiale și pentru a explora spațiul de căutare. În funcție de caracteristicile fiecărui algoritm, sursa de inspirație și metoda de implementare, pot fi utilizate diferite strategii pentru a efectua acest pas. Acest capitol își propune să analizeze utilizarea variantelor de mutație specifice algoritmului DE în diferite metaeuristici și impactul acestora asupra performanței. Deși la nivel conceptual, nu toate metaeuristicile folosesc mutația pentru a modifica populația, din perspectiva implementării, diferite concepte sunt modelate matematic folosind strategia de diferențiere. Ca atare, pot fi modificați anumiți pași specifici pentru unii dintre algoritmi pentru a include variantele de mutație întâlnite în DE.

5.1. Descriptori statistici

5.1.1. Diferența de fitness (gap)

Graficul de tip boxplot pentru întregul set de date este prezentat în Figura 5-1. După cum se observă, M0 are o performanță mediană moderată cu mustăți care se extind semnificativ în sus, indicând că, deși în general stabilă, această variantă poate produce uneori rezultate notabil mai slabe. M1 are cea mai mare mediană, indicând o performanță centrală mai mică decât alte variante. Acest lucru sugerează că în mod consecvent, M1 are dificultăți în identificarea soluțiilor optime. Mai mult, M1 prezintă mustăți lungi care se extind spre valori de fitness mai mari, sugerând că produce frecvent valori extreme sau soluții suboptimale. M4 atinge în mod consecvent una dintre cele mai mici valori mediane dintre toate variante de mutație, indicând o performanță centrală puternică. M6 demonstrează valori mediane foarte scăzute, comparabile cu M4, indicând capacități eficiente de optimizare. Mai mult, Figura 5-1 indică mustăți mai scurte pentru M6 decât pentru majoritatea celorlalte variante, sugerând mai puține valori extreme și o stabilitate mai mare a performanței.

O analiză a rezultatelor obținute indică faptul că M4 prezintă cea mai mică diferență a mediei valorilor *gap* pentru D10, iar o creștere a dimensionalității la D20 menține valori medii scăzute, evidențiind o scalabilitate și robustețe excelente. La D10, M6 atinge valori medii relativ scăzute, ușor mai mari decât M4, dar semnificativ mai bune decât alte variante de mutație, demonstrând o performanță ridicată. Adaptabilitatea specifică lui M6 îi permite să fie mai eficient decât strategiile clasice precum M1 și M2, în special pentru F11. La D20, mediile sunt ușor mai mari, dar încă competitive comparativ cu M4, indicând o bună scalabilitate și o performanță stabilă la dimensionalitate crescută.

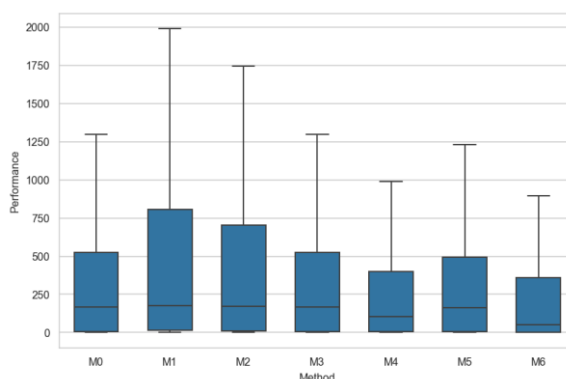


Figura 5-1. Valorile gap pentru toate variantele de mutație pentru toate combinațiile testate

Pentru F2, deviația standard a M0 crește de 7.5 ori de la 265.62 (10D) la 2004.38 (20D). Această creștere disproporționată a varianței în comparație cu mediile indică faptul că algoritmi testati devin nu doar mai lenți, ci fundamental mai puțin predictibili pentru dimensiuni superioare. M4 prezintă cea mai mică creștere a deviației standard la trecerea la dimensiuni superioare – de la 107.96 la 668.96 pentru F2, reprezentând o creștere de 6.2 ori comparativ cu creșterea de 7.5 ori pentru M0. Această stabilitate relativă contribuie probabil la performanța superioară generală a lui M4 în 20D.

În 10D, majoritatea funcțiilor prezintă o asimetrie pozitivă și o valoare a gradului de aplatizare ridicată, indicând alungiri la dreapta cu ocazionale valori extreme. Pe măsură ce dimensiunea crește la 20D, formele distribuției se schimbă fundamental. De exemplu, valoarea gradului de aplatizare pentru M0 și F2 scade de la 3.20 (10D) la -0.08 (20D), indicând o tranziție de la distribuții având un vârf ascuțit la distribuții mai plate, mai uniforme. Acest lucru sugerează că soluțiile devin distribuite mai uniform în dimensiuni superioare, indicând că alocarea mai multor resurse (o valoare FES mai mare) ar putea permite exploatarea suplimentară a indivizilor existenți.

5.1.2. Convergența

Convergența generală a variantelor de mutație considerate este prezentată în Figura 5-2. Aceeași pași ca în Capitolul 4 sunt aplicați pentru a determina convergența generală. Toate variantele de mutație încep cu valori ridicate ale *gap* (aproximativ 10^3) pentru ambele dimensionalități și arată o îmbunătățire inițială rapidă în timpul primelor 100-200 de iterații. Pentru D10 algoritmi să atingă valori finale ale metricii *gap* mai mici decât pentru D20, demonstrând clasicul „blestem al dimensionalității”, unde problemele de dimensiuni superioare sunt inerent mai dificil de optimizat. Cu toate acestea, comportamentele de convergență pe termen lung diferă între cele două dimensiuni: D10 are forme de convergență mai diverse, cu separări clare între grupurile de algoritmi. În același timp, D20 prezintă un comportament mai omogen, majoritatea algoritmilor atingând platouri în jurul iterațiilor 300 - 400. De exemplu, M1 și M2 ating un platou mai înalt, indicând că utilizarea lor duce

la blocarea algoritmilor în minime locale mai rapid comparativ cu M4 și M6. Pentru M6, în special în cazul D20, forma curbei de convergență indică faptul că, dacă sunt alocate resurse suplimentare, algoritmi au o probabilitate ridicată de a îmbunătăți soluțiile găsite.

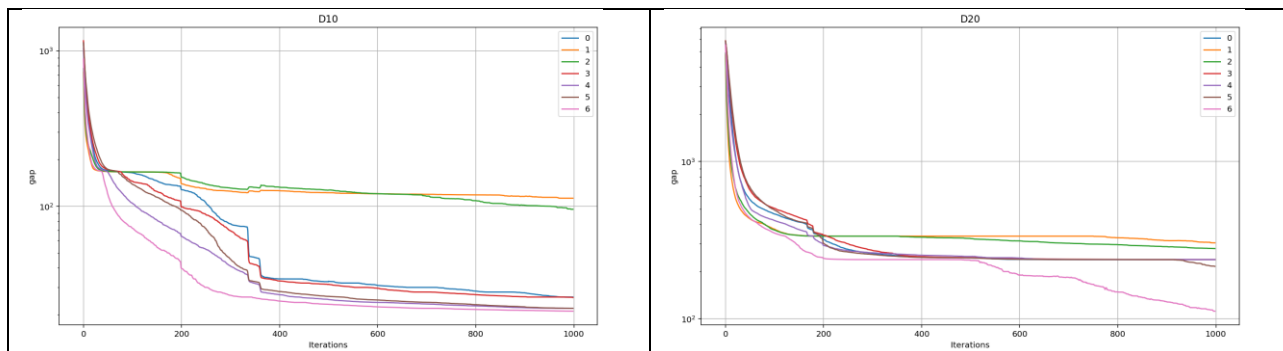


Figura 5-2. Rata mediană a convergenței pentru variantele de mutație testate(cu axa y pe scară logaritmică)

5.1.1. Diversitatea populației

Figura 5-3 arată variația valorii GDM4 medii în raport cu strategia de mutație de-a lungul iterațiilor. Scăderile bruște observate atât pentru D10, cât și pentru D20 reprezintă impactul algoritmului SAPDE. M4 și M5 au un comportament diferit față de celelalte mutații, menținând o diversitate mare pe parcursul celor 1000 de iterații. Aceste mutații prezintă scăderi distincte în trepte, mai degrabă decât curbe netede, sugerând posibile ajustări periodice în mecanismele de mutație. Chiar și după 500 de iterații, acestea se stabilizează la valori de diversitate semnificativ mai mari decât alte variante. Schimbarea de la D10 la D20 tinde să păstreze același clasament, indicând faptul că o creștere a dimensionalității nu duce la comportamente semnificative în ceea ce privește diversitatea.

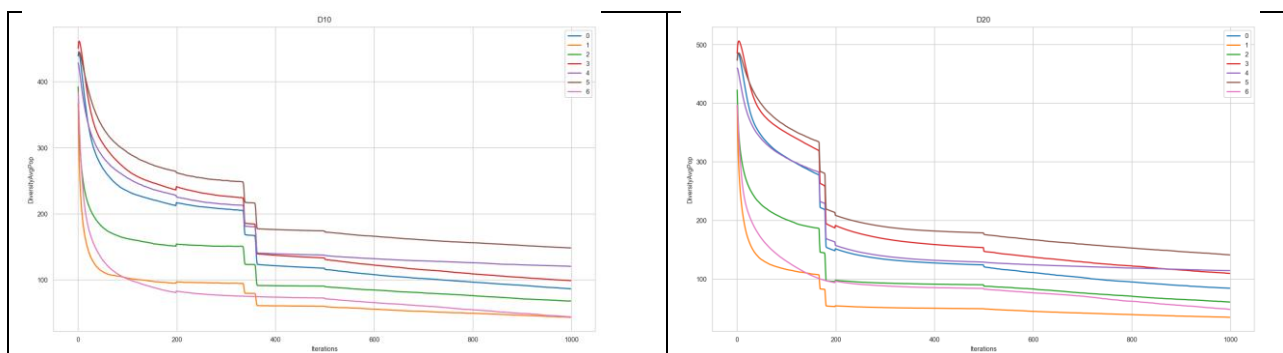


Figura 5-3. Variația diversității medii a populației pentru variantele de mutație pentru D10 și D20

M4 și M5 mențin în mod constant un nivel mai ridicat ale diversității, sugerând capacități mai puternice de explorare, în timp ce M1 și M6 prezintă o diversitate mai scăzută, indicând un accent mai mare pe exploatare. Analiza performanței a indicat M4 și M6 ca fiind cei mai eficienți pentru problemele testate. Corelarea performanței cu diversitatea populației a indicat că diversitatea ridicată a M4 corespunde celei mai scăzute valori medii *gap*, sugerând că dacă consumul FES nu ar fi oprit rulările, ar fi condus la îmbunătățiri suplimentare ale performanței. Pe de altă parte, pentru M6,

diversitatea scăzută indică faptul că populația este concentrată în optime locale, și sunt necesare mecanisme suplimentare pentru a identifica zone potențial adecvate pentru explorare. Din perspectivă practică, M4/M5 se potrivesc problemelor care necesită o explorare extensivă a spațiului de căutare, în timp ce M1/M6 se potrivesc mai bine problemelor ce necesită o convergență rapidă.

5.2. Testarea ipotezelor

Analiza *performanței* utilizează valorile *gap* și corespunde ipotezelor $H_0_Q3_p$ (toate variantele de mutație au performanță egală) și $H_1_Q3_p$ (cel puțin o variantă de mutație diferă semnificativ în performanță). Testul Shapiro a indicat că ipoteza normalității a fost respinsă, iar aplicarea testului Krushal-Wallis a rezultat într-o valoare statică de 1674.32, cu o valoare $p < 0.001$. Acest lucru indică faptul că, din perspectivă statistică, ipoteza nulă este respinsă pentru datele centralizate considerând toți algoritmi, și există diferențe semnificative între variantele de mutație. Din cele 21 de comparații pereche, 14 (aproximativ 67%) au arătat diferențe semnificative statistic la pragul convențional de semnificație $p < 0.05$. Această proporție ridicată de rezultate semnificative sugerează o variabilitate substanțială între diferitele versiuni ale mutației. Comparațiile nesemnificative sugerează că M3 poate reprezenta o formă intermediară care are caracteristici comune cu M0 și versiunile ulterioare de mutație (M4, M5 și M6).

În cazul analizei *convergenței*, valorile FES necesare pentru a atinge 95% din optim sunt utilizate pentru a realiza analiza statistică. Această analiză corespunde ipotezelor $H_0_Q3_c$ (toate metodele de mutație au rate de convergență similare) și $H_1_Q3_c$ (cel puțin o metodă de mutație diferă semnificativ în ceea ce privește rata de convergență). Procedura aplicată în analiza convergență este similară cu cea utilizată pentru analiza performanței. Rezultatele au indicat că ipotezele de normalitate nu sunt îndeplinite pentru nici o strategie de mutație. În continuare, testarea ipotezei a rezultat într-o valoare statistică de 103.88 și o valoare p mai mică de 0.05. Ipoteza că se obțin rate de convergență similare este respinsă pentru următoarele grupuri: M6 vs. M5, M4, M3, M2, M1 și M0; M4 vs. M3, M2 și M1. Acest lucru indică faptul că strategiile de mutație 4 și 6 sunt cele mai diferite dintre toate variantele. Cu toate acestea, atunci când se analizează indicatorul mărimii efectului (Cohen), se observă că valorile sunt în intervalul 0.5 - 0.8, indicând un efect mediu (diferențele nu sunt atât de pronunțate).

5.3. Probleme de inginerie

Similar cu analiza efectuată în Capitolul 4, rezultatele obținute pentru problemele de inginerie au fost, de asemenea, centralizate în funcție de varianta de mutație aplicată algoritmilor testați. M0 poate găsi minime competitive pentru majoritatea problemelor, dar în general prezintă valori medii ridicate, indicând o performanță mai puțin consistentă. M1 atinge cele mai bune minime pentru ENG0, ENG2 și ENG3, indicând faptul că poate găsi minimele globale. Cu toate acestea, performanța

medie slabă sugerează că se blochează des în optime locale. M2 și M3 au o performanță moderată atât în ceea ce privește valorile minime, cât și pentru cele medii.

M4 este cel mai eficient, având un echilibru excepțional între găsirea minimelor rezonabile și menținerea unor medii scăzute, indicând o rată ridicată de succes. Pentru toate problemele, are mediile cele mai scăzute și este cea mai consistentă variantă de mutație pentru toate problemele. Majoritatea rezultatelor prezintă asimetrie pozitivă, indicând distribuții predominant asimetrice la dreapta, care sugerează rulări ocazionale cu performanță slabă care trag media în sus.

5.4. Clasamentul variantelor de mutație

Pentru algoritmul DevDS, varianta de mutație ce ocupă primul loc este reprezentată de M4 (current-to-best/1), urmată pe locul 2 de M5. Cu toate acestea, există o diferență semnificativă între scorul S , indicând faptul că pentru acest algoritm, mecanismul utilizat pentru generarea locului de oprire (pasul 3) are un impact major asupra performanței finale. Pentru implementările SBO, clasamentul dintre cei doi algoritmi diferă substanțial (locul 1 pentru OriginalSBO este ocupat de M6 iar pentru DevSBO de M3), indicând faptul că o mică modificare la nivel de implementare (fără a schimba conceptul de bază) poate duce la modificări ale comportamentului algoritmului și a sensibilității acestuia față de componentele sale. Pentru variantele DE se poate observa că M4 și M6 se plasează în majoritatea cazurilor pe cele mai bune poziții. Astfel, se observă că mecanismele DE tind să favorizeze variantele de mutație ce utilizează atât individul cel mai bun cât și cel curent. Introducerea mecanismului de auto-adaptare nu schimbă considerabil clasamentul între variantele mutației, însă se observă o diferență între scorurile obținute pentru OriginalDE și SSADE, SSADE având valori S mai ridicate, indicând faptul că acesta din urmă are performanțe mai bune.

Tabelul 5-1 prezintă clasamentul general, cu indicatorii calculați folosind întregul set de rezultate. După cum se observă, versiunea M4 reprezintă cea mai bună strategie, urmată îndeaproape de M6. În ceea ce privește scorul S , diferența dintre a doua și a treia strategie este semnificativă, indicând o distincție clară între ele. Pe de altă parte, diferența dintre prima și a doua strategie este minimă, indicând faptul că performanța lor este similară.

Tabel 5-1. Clasamentul general al variantelor de mutație testate

Mutație	SE	SR	S	R
M0	7.82	22.12	29.94	5
M1	11.69	15.63	27.31	7
M2	11.51	17.61	29.12	6
M3	8.48	22.12	30.61	4
M4	50.00	31.65	81.65	1
M5	8.58	24.04	32.62	3
M6	31.48	50.00	81.48	2

M4 utilizează indivizii curenți și cei mai buni, încurajând convergența spre zonele promițătoare ale spațiului de căutare (viteză de convergență crescută). Pe de altă parte, M5 utilizează indivizi curenți și aleatorii, ducând la o rată mai mică de exploatare a regiunilor promițătoare. Principala explicație pentru performanța bună a variantei M6 este reprezentată de combinarea caracteristicilor componentelor sale. Se consideră că M6 are o capacitate de explorare echilibrată (deoarece varianta utilizată se schimbă aleatoriu pentru fiecare iterație, se combină explorarea introdusă de variantele bazate pe indivizi selectați aleatoriu precum M0, M3 și M5 cu caracteristicile de exploatare ale variantelor bazate pe cea mai bună soluție precum M1, M2 și M4).

5.5. Strategia de îmbunătățire

Analiza efectuată în secțiunile anterioare s-a concentrat pe identificarea celei mai bune variante de mutație care poate fi utilizată pentru a îmbunătăți performanța diferiților algoritmi. Rezultatele au arătat că, dintre variantele testate (M0 - M6), cea mai potrivită pentru multe cazuri este M4. Ipoteza studiului a fost că operația de diferențiere ar putea înlocui mecanisme similare cu cele ale altor metaeuristici; astfel s-a efectuat o serie de comparații cu variantele clasice pentru algoritmi testați. Mai mult, este propusă și o abordare de hibridizare: aceasta se bazează pe un criteriu lacom, unde sunt create două populații (una cu strategia clasică utilizată în optimizator și cealaltă cu varianta M4 testată), iar cele mai bune soluții dintre cele două sunt utilizate în pașii/generațiile următoare. Principala motivație pentru strategia de hibridizare propusă este bazată pe observația că convergența (în multiple cazuri) atinge valori care nu sunt aproape de optimul cunoscut, indicând faptul că algoritmi sunt blocați în zone locale. Astfel, introducerea unui mecanism suplimentar pentru a efectua o fază de explorare ar crește șansele algoritmilor de a identifica regiuni noi promițătoare. Deși, din perspectivă statistică, aplicarea diferitelor distribuții a fost minimă, distribuția Weibull a oferit cea mai bună performanță medie. Drept urmare, a fost utilizată în faza de implementare a algoritmilor rulați pentru a genera datele prezentate în aceasta secțiune.

Pentru DevDS, se poate observa că varianta hibridă depășește în mod constant celelalte două variante pentru majoritatea funcțiilor. Astfel, varianta hibridă prezintă o consistență superioară cu valori ale deviației standard mai mici, în timp ce varianta originală prezintă cea mai mare variabilitate. DevSBO are cele mai mici medii ale valorilor *gap* pentru majoritatea funcțiilor, indicând o capacitate de optimizare superioară în comparație cu ceilalți doi algoritmi. Similar cu DevDS, varianta hibridă obține în general cea mai bună performanță dintre variantele DevSBO. Pentru OriginalSBO, strategia de hibridizare depășește semnificativ celelalte două variante pentru majoritatea funcțiilor și are cele mai mici valori pentru deviația standard.

Variantele hibride prezintă de obicei distribuții ale rezultate mai simetrice cu mai puține valori extreme. În consecință, varianta hibridă este cea mai performantă dintre modificările testate.

DevNMRA arată cea mai mare diferență între cele două versiuni (hibrid vs clasic), cu varianta hibridă depășind cu mult varianta clasică cu aproape două ordine de mărime. Cu toate acestea, varianta hibridă prezintă o valoare a gradului de aplatizare semnificativ mai mare, sugerând o performanță bună, dar ocazional imprevizibilă. DevSLO, DevINMRA și DevNGO prezintă diferențe de performanță mai modeste între variante, varianta hibridă având performanțe ușor mai bune.

Se poate observa că din perspectiva valorilor *gap*, versiunea hibridizată are în general eficiență mai bună decât versiunea clasică. DevNMRA-hibrid demonstrează o performanță excepțională, cu cea mai mică medie a valorilor *gap*, în timp ce DevISLO prezintă cea mai slabă performanță cu valori extrem de mari. Pentru variantele SLO, testele au demonstrat că performanța este mai slabă în comparație cu alți algoritmi. Acest lucru indică faptul că îmbunătățirile aplicate în DevISLO nu sunt suficient de mari pentru a fi mai competitive în comparație cu ceilalți algoritmi.

O reprezentare a diversității populației pentru ambele variante ale SLO în D10 și D20 în comparație cu variantele NMRA este prezentată în Figura 5-4. Variantele NMRA au o diversitate a populației mult mai mare comparativ cu SLO, indicând că introducerea unui mecanism suplimentar pentru explorare la nivelul pașilor existenți (care să nu necesită consumul de FES) ar fi benefică pentru SLO. Pentru NMRA, tendința descendentă pentru diversitatea populației sugerează că dacă sunt oferite resurse suplimentare, există o probabilitate ridicată de identificarea a unor optime locale suplimentare.

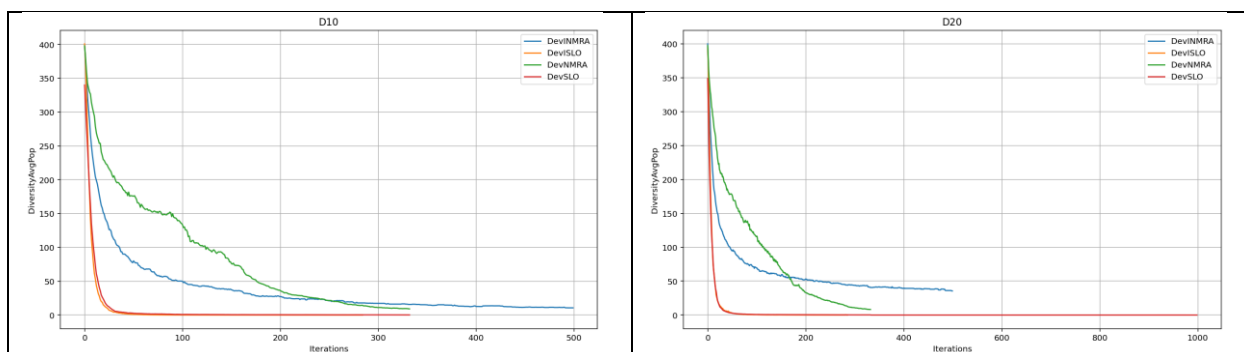


Figura 5-4. Variația diversității populației pentru variantele algoritmilor SLO și NMRA

6. Analiza performanței algoritmilor și aplicațiile acestora pentru probleme din ingineria chimică

6.1. Analiza statistică

În D10, LSHADE și OriginalSHADE au cele mai mici valori medii pentru majoritatea funcțiilor, cu deviații standard reduse, indicând o convergență consistentă. În majoritatea cazurilor, LSHADE are performanțe ușor mai bune comparative cu OriginalSHADE. DevDS și OriginalDE au performanțe slabe și au cele mai mari valori medii pentru majoritatea funcțiilor, în principal F6, F8 și F11. DevSBO este foarte eficient pentru funcția F3, dar nu reușește să genereze rezultate bune pentru F12. În D20, DevSBO a menținut o performanță relativ stabilă. Deși majoritatea algoritmilor au un grad de aplatizare pozitiv, SAPDE are un grad de aplatizare negativ pentru mai multe funcții în 20D, indicând distribuții neobișnuit de plate. Din perspectiva valorii medii, LSHADE și OriginalSHADE se evidențiază ca cei mai eficienți algoritmi pentru funcțiile testate, demonstrând în mod constant o performanță superioară. Caracteristicile unice ale acestor doi algoritmi în comparație cu ceilalți algoritmi sunt reprezentate de mecanismele avansate de control al parametrilor F și Cr . Faptul că cei doi algoritmi sunt similiari explică performanțele relativ similare obținute în pasul de testare. Totodată, performanța ușor mai bună obținută de LSHADE indică faptul că mecanismul adaptiv pentru dimensiunea populației precum și mecanismul de actualizare a memoriei aplicat adaptării parametrilor de control F și Cr au un impact pozitiv asupra eficienței generale a acestui algoritm. Totodată, și SAPDE aplică un mecanism bazat pe perturbație Gaussiană pentru adaptarea dimensiunii populației iar adaptarea parametrilor de control se realizează folosind același mecanism aplicat ca în pasul de mutație. Performanța mult mai scăzută a SAPDE comparativ cu variantele SHADE subliniază importanța strategiilor de adaptare eficientă a tuturor parametrilor de control și indică faptul că este necesară o înțelegere mai aprofundată a mecanismelor care permit generarea de noi abordări care să permită o îmbunătățire a eficienței acestora.

DevSBO menține o performanță relativ stabilă la scalarea de la 10D la 20D. Mai mult, prezintă cel mai scăzut coeficient de variație pentru mai multe funcții. Spre deosebire de alți algoritmi, DevSBO nu prezintă scăderi extreme de performanță pentru nici o funcție, indicând că este o strategie echilibrată ce poate fi aplicată pentru multe sisteme. OriginalSBO tinde să fie printre cei mai slabi optimizatori, indicând că această variantă necesită mai multe evaluări pentru a atinge obiectivul. DevSBO și LSHADE apar ca algoritmi puternici ce pot rezolva majoritatea funcțiilor. În schimb, algoritmi precum DevDS și OriginalDE necesită frecvent semnificativ mai multe evaluări.

Fiecare algoritm atinge valoarea maximă pentru FES în cazul funcției F10, în timp ce pentru funcțiile F2, F4, F6, F7, F8, F9 și F11, există o variație semnificativă în numărul FES necesar. În

6. Analiza performanței algoritmilor și aplicațiile acestora pentru probleme din ingineria chimică

contextul competițiilor CEC, când o funcție forțează în mod constant toți algoritmi să epuizeze întregul buget de evaluare, este un indicator puternic că proprietățile de bază ale funcției —cum ar fi multi-modalitatea, asperitatea sau optimele locale înșelătoare—o fac deosebit de rezistentă la convergență. În funcție de funcția analizată, profilul de convergență al algoritmilor se schimbă. După cum a fost observat și în capitolele anterioare, SAPDE este un algoritm care se comportă diferit în comparație cu ceilalți algoritmi. Acest lucru se datorează dimensiunii populației adaptive care consumă FES alocat mai rapid pentru D20 comparativ cu D10

DevDS și SAPDE reușesc foarte bine să păstreze diversității, SAPDE atingând valori medii ale diversității. Deviațiile standard pentru acești algoritmi relevă strategii distincte: DevDS tinde să mențină o explorare consecventă, în timp ce SAPDE are o diversitate mai mare în D20. Mai mult, acești algoritmi prezintă asimetrie negativă, indicând distribuții cu o alungire la stânga, concentrate în apropierea limitelor superioare pentru FES. Deși acest lucru este de obicei benefic pentru a scăpa de optimele locale, necesită mai multe resurse pentru a găsi soluții aproape de optim. Strategia SAPDE are cea mai mare diversitate a populației. La începutul rularilor, arată o creștere rapidă, urmată de două perioade de stabilitate relativă. Similar convergenței, datorită utilizării unui număr diferit de iterații, primul platou corespunde mediei dintre D10 și D20, în timp ce cel de-al doilea platou corespunde mediei obținute pentru D10. Un alt comportament distinct este observat pentru OriginalSBO, care, pentru toate funcțiile, arată o bună stabilitate a diversității. Cu toate acestea, rata sa lentă de convergență indică faptul că necesită o fază de exploatare suplimentară pentru a se concentra pe cele mai promițătoare zone ale spațiului de căutare.

Tabelul 6-1 prezintă un clasament al algoritmilor din punct de vedere al performanței. După cum se poate observa, câștigătorii clasamentului sunt reprezentați de OriginalSHADE și LSHADE, urmați de DevSBO. Scorurile între algoritmi variază substanțial, evidențiind faptul că performanțele acestora și eficiența lor în diferite cazuri sunt relative ușor de diferențiat.

Tabel 6-1. Clasamentul algoritmilor analizați pe baza rezultatelor obținute cu funcțiile specifice competiției CEC 2022

Algoritm	SE	SR	S	R
DevDS	19.37	11.6	30.92	7
DevSBO	12.03	42.9	54.97	3
LSHADE	50.00	29.0	78.97	1
OriginalDE	19.40	13.7	33.13	5
OriginalSBO	12.55	13.0	25.59	8
OriginalSHADE	12.03	50.0	62.03	2
SAPDE	19.91	13.0	32.95	6
SSADE	16.45	23.1	39.55	4

6.2. Complexitatea algoritmilor

Urmând criteriile CEC 2022, complexitatea algoritmilor atât în varianta cu mutația cea mai bună (M4) cât și în varianta hibridă a fost determinată. Rezultatele arată că cel mai rapid algoritm este DevDS, în timp ce cel mai lent este OriginalSBO. Mai mult, OriginalSBO prezintă probleme severe de scalabilitate, cu eficiența deteriorându-se de 3 ori de la D10 la D20. Motivația pentru cerințele generale ridicate ale OriginalSBO este reprezentată de modul în care sunt implementați pașii: modificări specifice sunt efectuate la nivelul fiecărei dimensiuni în indivizi. O modificare simplă în faza de implementare, unde pașii sunt aplicați la nivel individual, reduce semnificativ timpul de execuție în ambele dimensiuni, după cum se poate observa pentru DevSBO.

SSADE și OriginalDE au o eficiență moderată, în timp ce pentru LSHADE și OriginalSHADE se observă o creștere semnificativă odată cu creșterea dimensionalității (eficiență > 550 la D20), evidențiind provocări în reglarea adaptivă a parametrilor. Costul computațional mai mare al versiunilor SHADE se bazează pe modificările aplicate variantei de bază: memoria pentru parametrii de control. Dacă utilizarea memoriei are un impact bun asupra performanței, în ceea ce privește resursele consumate, adaugă un anumit overhead.

O ușoară reducere a timpului de execuție poate fi observată pentru versiunile hibride. În acest caz, FES alocat este consumat în mai puține iterații decât versiunile sale non-hibride. Prin urmare, această diferență în timpul de execuție se datorează operațiunilor legate de selectarea indivizilor care formează următoarea generație. Corelat cu timpul de execuție redus, se poate concluziona că metoda de hibridizare propusă, este o metodă simplă, eficientă și ușor de implementat ce are capacitatea de a îmbunătăți performanța metaeuristicilor.

6.3. Probleme de inginerie

Similar capitolelor anterioare, performanța algoritmilor a fost analizată nu numai pentru funcțiile specific competiției CEC 2022 dar și pentru un set de probleme ingineresti bine cunoscute. Simulările au fost efectuate în condiții similare și cu aceleași setări de parametri. Deoarece majoritatea problemelor ingineresti au cel puțin o constrângere, eroarea acceptată este 10^{-8} . Funcția de penalizare adaugă 10^8 pentru fiecare constrângere nesatisfăcută.

Pentru ENG0, patru algoritmi au atins valoarea optimă exactă: DevSBO, LSHADE, OriginalSHADE și SSADE. Totuși, valorile mediane arată că DevSBO demonstrează o performanță tipic mai bună, semnificativ mai mică decât a celorlalți algoritmi. Acest lucru indică faptul că, deși mai mulți algoritmi pot găsi optimul în cele mai bune rulări, DevSBO menține o performanță mai bună pe mai multe repetări. Niciunul dintre algoritmi nu atinge optimul cunoscut pentru ENG1. Pentru ENG2 și ENG3, toți algoritmii au performanțe slabe, niciunul generând valori pentru funcția de fitness în apropiere de optim (toate soluțiile furnizate nu satisfac cel puțin una dintre constrângeri).

6. Analiza performanței algoritmilor și aplicațiile acestora pentru probleme din ingineria chimică

Pentru ENG2, LSHADE a obținut minimul cel mai apropiat, urmat de SSADE. Pentru ENG3, LSHADE a obținut cel mai bun minim. Aceste rezultate sugerează că aceste probleme prezintă provocări semnificative pentru toți algoritmi testați, posibil din cauza spațiilor complexe definite de multitudinea de constrângeri.

Patru algoritmi au atins optimul exact pentru ENG4 și ENG5: DevSBO, LSHADE, OriginalSHADE și SSADE. Pentru ENG5, LSHADE demonstrează cea mai bună performanță mediană și cea mai mare rată de succes, în timp ce DevSBO are cea mai bună mediană, indicând adecvarea sa la problemele testate. ENG6 este problema cea mai ușor de rezolvat, cinci algoritmi generând soluții optime: DevSBO, LSHADE, OriginalDE, OriginalSHADE și SSADE

LSHADE și OriginalSHADE se remarcă ca fiind cei mai eficienți pentru problemele de inginerie, fiecare atingând optimul exact în 4 din 7 probleme. Ei demonstrează constant o capacitate superioară de a găsi optime globale și o performanță mediană puternică, sugerând consistență între rulări, performanță mai bună decât alți algoritmi pe probleme dificile precum ENG2 și ENG3. DevSBO atinge, de asemenea, optimul în 4 din 7 probleme (ENG0, ENG4, ENG5 și ENG6). Ceilalți algoritmi (DevDS, OriginalDE, OriginalSBO și SAPDE) au, în general, performanțe slabe. Aceste rezultate pentru problemele ingineresti sunt corelate în principal cu cele obținute pentru funcțiile specific competiției CEC 2022, observându-se tendințe similare.

6.4. Aplicații din viața reală (procese din ingineria chimică)

Această secțiune exemplifică aplicarea diferiților algoritmi analizați în capitolele anterioare pentru modelarea și optimizarea unor procese din ingineria chimică. Spre deosebire de problemele de inginerie utilizate ca funcții de test în combinație cu funcțiile din competiția CEC 2022, procesele studiate în această secțiune nu sunt descrise de modele fenomenologice. Astfel, este necesară generarea unor modele ce vor fi utilizate ulterior în faza de optimizare. Aceste modele pot fi obținute folosind diferite strategii, dintre care, în această teză sunt aplicate două variante bazate pe date experimentale: i) modele de regresie (folosind programe specializate, de exemplu Minitab) și ii) Rețele Neuronale (Neural Networks, NN). Configurația optimă a unui model NN este o problemă de optimizare în sine, abordarea din această secțiune realizează acest pas folosind o strategie neuro-evolutivă. Schema generală care descrie pașii principali aplicați în această secțiune este prezentată în Figura 6-1.

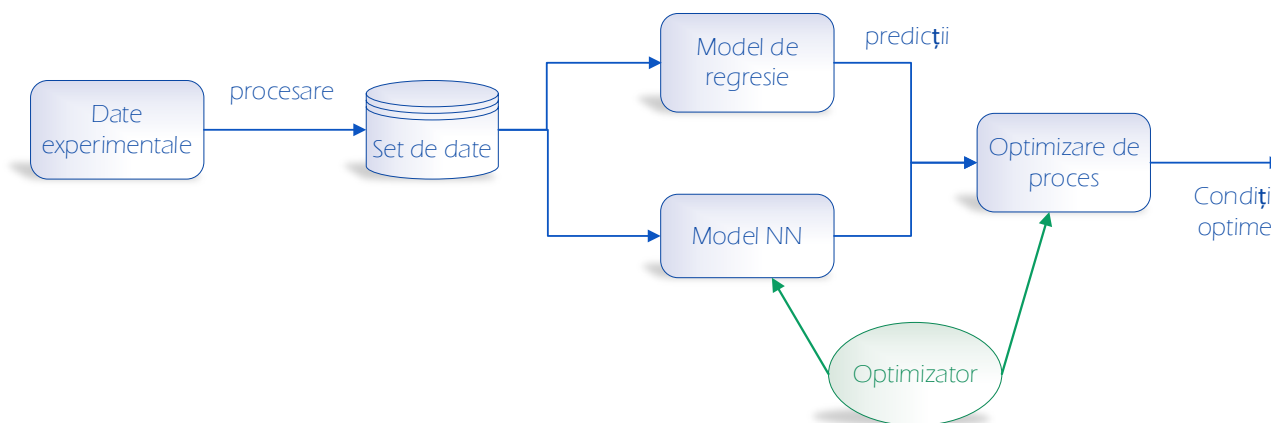


Figura 6-1. Schema generală pentru modelarea și optimizarea procesului

6.4.1. Eliminarea *clorofilei A* din apele uzate

Acest studiu se axează pe optimizarea unui proces de electrocoagulare (EC) pentru eliminarea *clorofilei A*, un indicator al prezenței algelor din efluenții unei stații de epurare a apelor reziduale tratate în lagune aerate. Procesul EC a fost optimizat folosind variante ale algoritmilor DE și DS pentru a determina cei mai buni parametri operaționali în contextul unor constrângeri diferite pentru intervalele de valori acceptate pentru o serie de parametri. Cantitatea finală de *clorofilă A* a fost modelată ca o funcție de regresie (folosind Minitab) ce primește ca parametri valorile inițiale ale *clorofilei* și parametrii operaționali (putere electrică, timp, distanță între electrozi, tip de electrozi).

După determinarea modelului procesului, algoritmi DE și DS (corespunzând versiunilor *OriginalDE* cu mutație rand/2 și *OriginalDS* fără modificări) au fost aplicați pentru a identifica condițiile optime care minimizează conținutul final de *clorofilă*. Au fost considerate două scenarii: i) când nu au fost impuse limite asupra valorilor variabilelor de proces (optimizare liberă) și ii) când sunt impuse limite și asupra puterii aplicate și a timpului (ca mijloc de reducere a resurselor consumate). În primul scenariu, DE a obținut valori extrem de mici ale *clorofilei A* (corespunzător unui proces cu o eficiență ridicată), depășind constant DS, care a produs valori acceptabile (cel mai bun: $8.97\text{E-}04 \text{ mg/m}^3$). În al doilea scenariu, ambii algoritmi au menținut performanțe bune, în ciuda constrângerilor de putere. Performanța DE este ușor mai slabă decât în cazul optimizării libere, aspect datorat constrângerilor aplicate spațiului de căutare. Pe de altă parte, performanța DS s-a îmbunătățit comparativ cu scenariul fără constrângeri, indicând că pentru acesta, limitarea spațiului de căutare are un impact favorabil, posibil datorită capacităților ridicate de explorare (după cum a fost demonstrat în capitolele anterioare). În al treilea scenariu, DE a obținut cel mai bun rezultat general, în timp ce DS a menținut o performanță constantă. Totuși, ambii algoritmi au trecut predominant la electrozi de aluminiu (tip 0).

Per ansamblu, DE excelează în optimizarea de precizie, găsind constant valori mai mici ale *clorofilei A*, în timp ce DS demonstrează o explorare superioară, generând combinații de parametri mai diverse. Astfel, DE este preferat atunci când se urmărește respectarea strictă a reglementărilor ce

6. Analiza performanței algoritmilor și aplicațiile acestora pentru probleme din ingineria chimică
impun un nivel minim de clorofilă. Pentru flexibilitate operațională cu rezultate acceptabile, DS oferă opțiuni mai variate pentru parametrii de proces.

6.4.2. Procesul de decolorare a rifaximinei

Acest studiu se concentrează pe decolorarea rifaximinei din soluții apoase folosind cărbune activ. Analiza constă în procesarea, modelarea și optimizarea datelor experimentale obținute utilizând o abordare de tip Planificare Experimentală (DOE). Modelul procesului a fost determinat folosind Metodologia Suprafeței de Răspuns (RSM), iar toți algoritmi testați în capitolele anterioare au fost aplicați pentru optimizarea procesului. În plus, au fost luate în considerare încă trei metaeuristici de inspirație biologică: BFO, PSO și GWO. Toți algoritmi au fost utilizați comparativ, scopul fiind identificarea celei mai potrivite abordări care să conducă la cea mai mare eficiență de eliminare.

Pe baza rezultatelor, algoritmi testați pot fi grupați în trei categorii: i) cei mai eficienți: LSHADE, DevSBO și OriginalSHADE. ii) algoritmi performanți: GWO, OriginalSBO, PSO, SSADE și SAPDE. Acești algoritmi au avut performanțe similare cu grupul cel mai eficient, dar cu o variabilitate ușor mai mare iii) algoritmi cu performanțe mai scăzute: DevDS, OriginalDE și BFO. Acești algoritmi au demonstrat eficiențe semnificativ mai mici și deviații standard mai mari, indicând performanțe mai puțin consistente și convergență suboptimă. Acest studiu demonstrează că algoritmi avansați de optimizare, în special LSHADE, OriginalSHADE și DevSBO, reprezintă instrumente puternice pentru determinarea condițiilor optime în procesele de adsorbție a rifaximinei. Convergența remarcabilă a mai multor algoritmi la valori similare ale parametrilor validează robustetea condițiilor optime identificate.

6.4.3. Decolorarea Eriocrom Negru T folosind cărbune activ și $h\nu/\text{TiO}_2$

Eriocrom Negru T (EBT) este un compus indispensabil pentru detecția ionilor metalici și testarea durității apei. Deoarece EBT este stabil la lumină și căldură, este dificil de eliminat chiar și la concentrații mici. Pentru acest studiu s-a aplicat adsorbția pe cărbune activ (AC) și degradarea fotocatalitică cu TiO_2 (PD). Procesele AC și PD au fost studiate experimental pe baza unei strategii DOE și au fost modelate folosind abordarea RSM din software-ul Minitab. Modelele rezultate au fost apoi folosite în faza de optimizare a procesului, unde au fost testați toți algoritmi discutați în capitolele anterioare (variante de mutație M6). În plus, au fost testați încă doi algoritmi: BFO și PSO. Fiecare algoritm a fost rulat de 10 ori, iar rezultatele au fost centralizate. Dacă diferențele sunt mai mici de $1.0\text{E}-2$, soluțiile sunt considerate identice din perspectivă inginerescă. Cele mai bune soluții sunt prezentate în Tabelul 6-2.

Pentru PD, algoritmi DevDS, DevSBO, LSHADE, OriginalSBO, OriginalSHADE și SAPDE au atins cea mai mare eficiență de decolorare (99.25%). PSO are o eficiență maximă ușor mai mică,

6. Analiza performanței algoritmilor și aplicațiile acestora pentru probleme din ingineria chimică
de 99.169%, dar a avut totuși performanțe bune comparativ cu ceilalți. Algoritmi precum OriginalDE, BFO și unele variații PSO au avut eficiențe mai scăzute, BFO având o medie de aproximativ 97.121%.

Pentru AC, cele mai mari eficiențe au fost atinse constant la $Z1=0.4$, $Z2=65.4$ și $Z3=0.0325$. Aceasta indică faptul că aceste valori ale parametrilor sunt optime pentru decolorare în cazul mai multor algoritmi. Eficiențe mai scăzute au fost observate pentru OriginalDE și BFO, acești algoritmi prezentând o variabilitate mai mare a eficienței, cu valori între ~50% și ~58%. Acest lucru sugerează că pot fi mai puțin eficienți în identificarea combinațiilor optime de parametri. DevDS, DevSBO, LSHADE, SAPDE, OriginalSBO și OriginalSHADE au rezultate identice, sugerând că acești algoritmi converg la aceeași soluție în condițiile date, explorând eficient spațiul de căutare. Comparând algoritmi care au oferit constant rezultate maxime de optimizare, se observă că OriginalDE și BFO oferă soluții cu eficiența cea mai mică.

Tabel 6-2. Cele mai bune rezultate obținute pentru decolorarea EBT

	PD				AC			
Algoritm	Eficiență	Z1	Z2	Z3	Eficiență	Z1	Z2	Z3
DevDS	99.25	1.64	33	0.325	63.727	0.4	65.4	0.0325
	99.24	1.68	33	0.325				
DevSBO	99.25	1.64	33	0.325	63.727	0.4	65.4	0.0325
LSHADE	99.25	1.64	33	0.325	63.727	0.4	65.4	0.0325
OriginalDE	98.99	1.83	33	0.325	58.46	0.374	65.261	1.9004
OriginalSBO	99.25	1.64	33	0.325	63.727	0.4	65.4	0.0325
OriginalSHADE	99.25	1.64	33	0.325	63.727	0.4	65.4	0.0325
SAPDE	99.25	1.64	33	0.325	63.727	0.4	65.4	0.0325
SSADE	99.21	1.72	33	0.325	63.727	0.4	65.4	0.0325
BFO	97.121	1.703	32.784	1.371	60.175	0.37	65.277	0.222
PSO	99.169	1.609	32.995	0.365	63.556	0.4	65.295	0.066

6.4.4. Decolorarea lichiorului negru din tulpini de porumb folosind cărbune activ și TiO_2/UV

Lichiorul negru (BL) este unul dintre cei mai cunoscuți produși secundari ai industriei de celuloză și hârtie. Eliminarea lor a fost studiată folosind adsorbția pe cărbune activ (AC) și decolorarea cu TiO_2/UV (PD). Ambele procese au fost studiate folosind o procedură în trei pași: i) experimental (pe baza planului experimental stabilit prin metoda DOE); ii) modelare folosind RSM; iii) optimizare (unde modelele determinate anterior, combinate cu varianta SSADE cu mutație M4 sunt folosite pentru a identifica parametrii optimi ai procesului). Procedura de optimizare este aplicată în trei cazuri: i) limitele condițiilor de operare au fost aceleași ca în datele experimentale (Cazul 1); ii) limitele au fost extinse cu $\pm 20\%$ (extrapolare) (Cazul 2); și iii) numărul de reactivi activi adăugați a fost limitat (1-20 g/L cărbune activ pentru AC și 0.4-1 g/L TiO_2 pentru PD) (Cazul 3).

Comparativ cu RSM, SSADE este mai flexibil (permite o gamă largă de limite ale parametrilor). Totodată, este mai eficient în explorarea spațiului de căutare, determinând parametri de proces care conduc la aceeași eficiență. Mai mult, pentru cărbunele activ, s-a obținut o îmbunătățire de la 81.27% la 100%, iar pentru decolorarea cu TiO_2/UV de la 36.63% la 46.83%, demonstrând că utilizarea SSADE poate duce la îmbunătățiri semnificative.

6.4.5. Decolorarea compusului verde de bromcrezol folosind cărbune activ

Verdele de bromcrezol (BCG) are o varietate de aplicații ca indicator de pH, marker de ADN și colorant de urmărire în industria textilă. Acest studiu demonstrează aplicarea cu succes a metodelor clasice și a celor bazate pe inteligență artificială pentru optimizarea îndepărtării sale prin adsorbția pe cărbune activ. Scopul a fost de a demonstra că procesele clasice pot fi îmbunătățite prin utilizarea unei combinații de i) abordări standard de modelare și optimizare și ii) rețele neuronale artificiale cu o variantă îmbunătățită a algoritmului BFO (iBFO). Datele experimentale au fost obținute urmărind planul generat de metoda DOE. Procesul a fost apoi modelat prin două metode (RSM și NN) iar optimizarea procesului folosește RSM și iBFO (Figura 6-2).

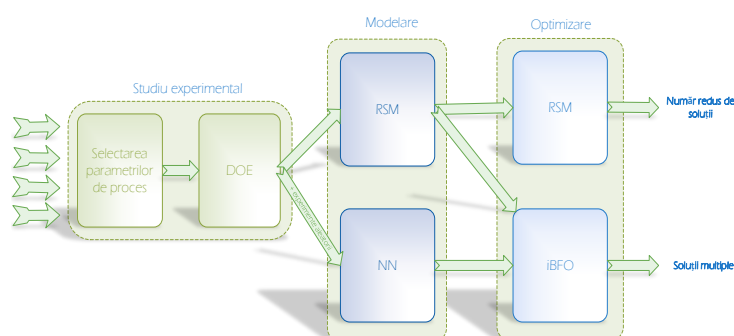


Figura 6-2. Modul de lucru pentru decolorarea BCG. Adaptat din (Suditu et al., 2022)

În pasul de modelare, rolul iBFO este de a determina arhitectura NN care se potrivește cel mai bine procesului analizat. Parametrii optimizați sunt: numărul de straturi ascunse și numărul de neuroni din fiecare strat. Eficiența în acest caz este evaluată folosind o funcție de fitness ce depinde de eroarea medie pătratică (MSE) obținute în timpul fazei de antrenare. Având în vedere că iBFO lucrează cu o populație de dimensiune fixă, pe baza analizelor preliminare, a fost stabilită o limită maxim admisă de 5 straturi ascunse. Funcția de activare pentru fiecare neuron din straturile ascunse este ReLU (Rectified Linear Unit), iar stratul de ieșire folosește o funcție de activare liniară. Optimizatorul Adam a fost utilizat la antrenarea fiecărei topologii identificate de către iBFO. Cel mai bun model obținut două straturi ascunse cu 33 și, respectiv, 22 de neuroni.

După identificarea celui mai bun model NN pentru procesul studiat, se aplică pasul de optimizare, utilizând același optimizator iBFO. Sunt considerate următoarele cazuri: i) când limitele pentru căutare sunt cele obținute prin experimentele suplimentare efectuate pentru extinderea setului de date, ii) când limitele sunt stabilite ca în abordarea DOE, (-1, 1) și iii) când limitele sunt stabilite

6. Analiza performanței algoritmilor și aplicațiile acestora pentru probleme din ingineria chimică

la valorile $(-\alpha, \alpha)$ corespunzătoare planului DOE. Cea mai bună soluție de optimizare în toate cazurile este prezentată în Tabelul 6-3.

Tabel 6-3. Rezultatele optimizării procesului de decolorare al BCG

	Caz	AA [g/L]	CT [min]	BCG [g/L]	η [%]
iBFO+RSM	i) (-1,1) din DOE	1.683	81	0.010000	34.2
	ii) $(-\alpha, \alpha)$ din DOE	1.844	197	0.005111	39.3
iBFO+NN	i) Limitele extinse	0.110	4080	0.001730	99.9
	ii) (-1,1) din DOE *	1.777	237	0.009000	45.6
	iii) $(-\alpha, \alpha)$ din DOE *	1.844	197	0.005111	39.3

Rezultatele obținute subliniază fezabilitatea identificării unor condiții care să conducă la o eficiență de aproape 100% atunci când se consideră limitele extinse. Strategia RSM este riguroasă din punct de vedere statistic, permite vizualizarea clară a interacțiunilor dintre parametri și are cerințe computaționale reduse. Totuși, rezultatele optimizării nu sunt întotdeauna în apropierea optimului global. Pe de altă parte, strategia iBFO+NN surprinde relații neliniare complexe, permite extrapolarea dincolo de intervalele inițiale de date și oferă multiple soluții de optimizare care pot susține numeroase cazuri de utilizare unde un anumit parametru poate fi limitat în intervalul dorit.

6.4.6. Predicția temperaturilor de tranziție pentru molecule cu cinci inele derivate din resorcinol

Predicția acestor temperaturilor de tranziție de fază ale cristalelor lichide este esențială pentru proiectarea unor structuri noi cu proprietăți specifice, iar acest studiu propune o nouă metodologie care combină NN și o variantă DE. Datele utilizate sunt reprezentate de 172 de compuși pentru care structurile moleculară a fost optimizată și au fost calculați diverși descriptori moleculari (parametri structurali, proprietăți termo-fizice, descriptori electronici și informații topologice) ce reprezintă intrările (notate F0-F21) pentru modelele NN dezvoltate.

Strategia de modelare este bazată pe două etape (Figura 6-3). În prima etapă, un model NN cu mai multe intrări și mai multe ieșiri a fost dezvoltat pentru a genera predicții corespunzătoare celor șase temperaturi de tranziție (T1-T6). Pe baza acestor predicții, a fost derivată o nouă caracteristică (F21). În a doua etapă, au fost dezvoltate modele NN individuale pentru fiecare temperatură de tranziție. Aceste modele au folosit cei 21 de descriptori inițial considerați (F0-F20) plus caracteristica derivată F21. Varianta DE/Best/2 cu parametri auto-adaptivi, corespunzătoare SSADE cu strategia de mutație M2 bazat pe DE optimizează topologia modelelor determinate. Funcția de activare pentru neuronii din stratul ascuns este ReLu, iar stratul de ieșire conține neuroni cu funcție de activare liniară. Antrenarea este realizată de optimizatorul Adam cu o rată de învățare de 0.05.

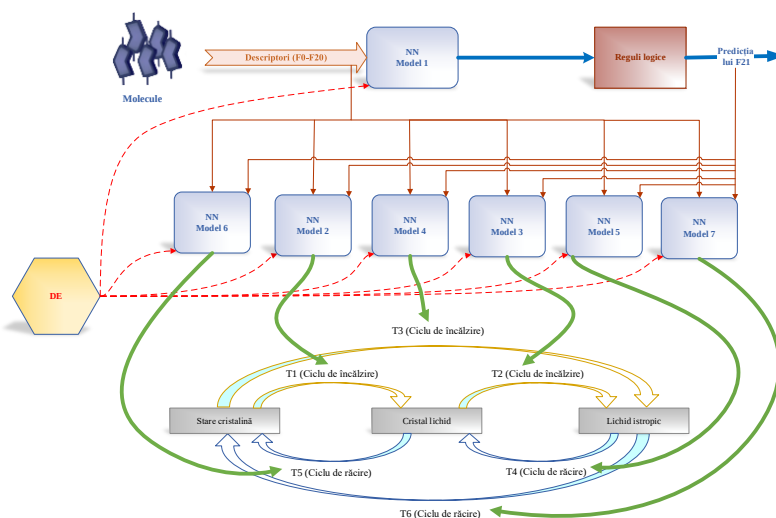


Figura 6-3. Strategia de modelare a temperaturilor de tranziție de fază. Adaptat din (Drăgoi et al., 2023)

Prima fază a modelării se concentrează pe predicția lui F21 folosind un model NN general (NN1), care determină simultan T1- T6. Valorile prezise de NN1 sunt introduse într-o regulă logică în scopul estimării caracteristicii F21. Această abordare a fost adoptată în locul predicției directe datorită erorilor semnificative asociate clasificatorilor testați. Metodologia bazată pe reguli logice a dus la o rată de clasificare greșită de 4.3% în faza de antrenare și 17.2% în faza de testare. În faza a doua a modelării, F0-F21 sunt utilizați pentru a dezvolta modele individuale pentru fiecare temperatură de tranziție. Tabelul 6-4 prezintă proprietățile modelelor optime identificate pentru fiecare temperatură. RMSE indică rădăcina erorii mediei pătratice. Topologia este exprimată folosind notația $inp:H1:...:Hn:out$, unde "inp" indică numărul de intrări, "Hi" reprezintă numărul de neuroni din stratul ascuns, iar "out" semnifică numărul de ieșiri. MAPE se referă la eroarea medie procentuală absolută.

Tabel 6-4. Modelele ANN optime în cazul modelării individuale a temperaturilor de tranziție.

Ieșire	Topologie	RMSE antrenare	MAE antrenare	MAPE antrenare	RMSE testare	MAE testare	MAPE testare
T1	22:15:5:8:1	6.3506	3.41059	0.047475	10.4202	6.38020	0.09367
T2	22:15:0:1	6.0332	2.76686	0.029141	13.9295	7.12744	0.08375
T3	22:17:8:1	5.4194	2.84321	0.053734	12.9596	5.48218	0.13090
T4	22:15:0:1	6.5939	4.46069	0.041046	13.4473	13.02247	0.13327
T5	22:19:5:9:1	5.6258	3.84472	0.056967	15.2732	11.31098	0.11245
T6	22:04:0:1	11.6825	2.86585	0.070765	18.6110	8.071391	0.20796

După cum poate fi observat, varianta SSADE a reușit să determine modele bune care pot prezice eficient caracteristicile dorite pe baza descriptorilor structurali, topologici, termo-fizici și electronici ai compușilor analizați. Modelele optimizate au demonstrat o capacitate predictivă bună, cu valori R^2 cuprinse între 0.89 și 0.98 pentru diferite temperaturi de tranziție.

6.4.7. Compostarea nămolului uleios

Principalul obiectiv al acestui studiu este determinarea unui model eficient care să poată descrie obținerea carbonului organic (OC) și a hidrocarburilor petroliere totale (TPH) în timpul procesului de compostare a nămolului de pe fundul rezervoarelor de stocare. Astfel, pentru a determina un model care poate prezice eficient OC și TPH în diverse condiții, a fost aplicată o metodologie neuro-evolutivă, bazată pe NN și SSADE.

Spre deosebire de studiile de caz din secțiunile anterioare, unde metaeuristica a fost aplicată doar pentru a optimiza topologia, pentru acest studiu, SSADE realizează o optimizare simultană a topologiei și a hiper-parametrilor (pas corespunzător fazei de antrenare). Astfel, structura indivizilor din cadrul populației SSADE cuprinde numărul de straturi ascunse, numărul de neuroni pe fiecare strat ascuns, ponderi, termen liber (*bias*) și funcții de activare).

Statisticile obținute pentru modelele optime determinate în cazul în care TPH și OC sunt modelate simultan au indicat diferențe majore între eficiența predicțiilor generate. Pentru TPH, eroarea medie absolută relativă (AARE) calculată în faza de testare este de 7.29%, în timp ce pentru OC este de 20.86%. Acest lucru indică faptul că TPH este mai ușor de modelat decât OC. Aceasta se datorează faptului că au fost folosite diferite amestecuri ca și materie primă, fiind disponibile în procesul de compostare fracțiuni diferite de OC. Tipul de materiale organice specifică rata de îndepărtare, deoarece compoziția lor este dată în principal de fracții carbonice cu rate de bioremediere diferite. Astfel, modelarea OC ar trebui să includă opțiunea de a folosi diferite fracțiuni de carbon, inclusiv ușor biodegradabile, lent biodegradabile și inerte. În consecință, următorul pas a constat în modelarea fiecărei ieșiri folosind modele ANN diferite. Acest lucru are două avantaje: i) se poate determina dacă cele două ieșiri sunt interdependente; ii) are potențialul de a genera rezultate mai bune pentru cel puțin un parametru. Performanța modelelor (prin indicatorul MSE în faza de testare) este mai bună atunci când se consideră modele individuale pentru cele două ieșiri decât atunci când sunt modelate simultan. Pentru cazul 2, pentru cele mai bune modele ANN, eroarea medie absolută pentru TPH în faza de testare este de 5.96%, iar pentru OC este de 21.83%.

7. Concluzii finale

7.1. Concluzii privind obiectivele tezei

Teza abordează problemele întâlnite în domeniul metaeuristicilor și are trei obiective principale, corespunzătoare la trei întrebări: i) **(Q1)**: Cum pot fi clasificate metaeuristicile pentru a avea o imagine de ansamblu mai bună asupra întregii liste de algoritmi existenți? ii) **(Q2)**: Cât de semnificativă este distribuția folosită în generarea numerelor aleatoare utilizate de obicei de către metaeuristici? iii) **(Q3)**: Poate operatorul de diferențiere specific DE să influențeze comportamentul diferitelor metaeuristici?

Pentru a răspunde la întrebarea **(Q1)**, a fost analizată în detaliu literatura de specialitate și o organizare a metaeuristicilor inspirate de animale a fost propusă urmărind taxonomia biologică. În ciuda criticilor tot mai numeroase din partea comunității științifice, dezvoltarea algoritmilor metaeuristici continuă să se extindă exponențial, fiind susținută de aplicabilitatea lor largă și de accesibilitatea atât a algoritmilor consacrați, cât și a metodelor mai noi cu cod sursă disponibil. Totuși, s-a constatat că lipsa unei taxonomii unificate pentru metaeuristici rămâne o barieră critică pentru progresul sistematic. Principalele aspecte cheie care fac clasificarea metaeuristicilor o problemă complexă sunt legate de: i) *Inconsistențe terminologice*. Definițiile contradictorii îngreunează comparațiile între studii. În plus, utilizarea unor termeni complecși, proveniți din sursa de inspirație, pentru a transmite concepte simple poate duce la neînțelegeri și poate da impresia că acestea sunt procese foarte complexe, când de fapt sunt simple și bine cunoscute; ii) *Criterii care nu exclud mutual algoritmi*: Mulți algoritmi se încadrează în mai multe clase în funcție de criteriul considerat (de exemplu, strategii hibride care combină căutarea locală/globală).

Deși criteriile bazate pe tipul soluției (individual vs. bazat pe populație), strategia de căutare (locală vs. globală), utilizarea memoriei sau sursa de inspirație (biologică vs. non-biologică) sunt standard, nu există un cadru universal acceptat care să cuprindă toate criteriile.

Primul pas în stabilirea unei strategii eficiente pentru clasificare implică identificarea algoritmilor existenți. Prin urmare, a fost realizată o analiză a literaturii axată pe metaeuristicile inspirate de animale, și a fost propusă o **clasificare bazată pe sursa de inspirație**, urmând taxonomia biologică. Domeniul se inspiră predominant din comportamentele vertebratelor, care sunt ușor de observat, deși toate ramurile majore ale taxonomiei biologice au găsit reprezentare în domeniul metaeuristicilor. Sursele de inspirație mai exotice și atipice generează adesea mecanisme noi care se traduc în strategii inovatoare de rezolvare a problemelor. Principalele direcții identificate sunt: i) *Algoritmi inspirați de vertebrate*. Păsările și mamiferele sunt principalele surse de inspirație pentru această categorie de algoritmi. Exemple inspirate de păsări includ Cuckoo Search (CS) și African Vultures Optimization

(AVOA). CS și AVOA simulează căutarea hranei/împerecherea, dar se bazează pe modele suprasimplificate (de exemplu, zboruri Lévy). Deși eficiente în spațiile unimodale, datorită factorilor de scalare ficși, acestea întâmpină dificultăți în spații cu dimensionalitate mare. Exemple de algoritmi inspirați de mamifere includ Grey Wolf Optimizer (GWO) și Whale Optimization Algorithm (WOA). GWO și WOA modelează ierarhii sociale, dar suferă de convergență prematură. De exemplu, ierarhia rigidă alfa-beta-delta a GWO limitează adaptabilitatea în medii dinamice; ii) Algoritmi inspirați de nevertebrate. Principalele surse de inspirație includ insectele (cu algoritmi ca Ant Colony Optimization -ACO-, Dragonfly Algorithm -DA-) și arahnidele (cu algoritmi ca Social Spider Optimization -SSO-). ACO excelează în problemele combinatorice, dar are dificultăți în domenii complexe/continue. DA echilibrează roiurile statice/dinamice, dar nu are scalabilitate. SSO încorporează operatori specifici genului, dar introduce complexitate fără a garanta câștiguri de performanță.

Cercetările actuale se concentrează în principal pe introducerea de noi algoritmi și extinderea domeniilor de aplicabilitate, în timp ce studiile care investighează în profunzime impactul unor operatori sau mecanisme specifice asupra performanței algoritmilor sunt relative puține. Această observație evidențiază o problemă specifică acestui domeniu, iar analizele detaliate care examinează toți factorii ce influențează comportamentul performanței — și relația lor cu caracteristicile problemei — ar putea aduce beneficii substanțiale atât comunității metaeuristicilor, cât și domeniilor de aplicare. Menținerea unui echilibru între explorarea globală și exploatarea locală este esențială. Mecanismele adaptive propuse în algoritmi precum SHADE și LSHADE modifică dinamic parametrii folosind ratele de succes istorice, sporind robustețea pe diverse domenii de probleme. Secvențele determinate de haos (de exemplu, reglarea β -haotică în variantele GWO) introduc stocasticitate pentru a evita blocarea în optime locale, în timp ce abordările hibride precum GWO integrat cu Q-learning îmbunătățesc luarea deciziilor, cu costuri suplimentare de calcul. Metodele de măsurare a diversității populației — genotipică (codificarea soluției) și fenotipică (bazată pe fitness) — sunt esențiale pentru monitorizarea acestui echilibru. De exemplu, convergența fenotipică fără diversitate genotipică indică adesea multimodalitate înșelătoare, necesitând tehnici specifice pentru a păstra varietatea soluțiilor. Aceste observații au stat la baza analizelor realizate pentru a răspunde la următoarele două întrebări **(Q2) și (Q3)** pe care această teză a urmărit să le clarifice.

Totuși, înainte de a răspunde acestor întrebări, trebuie discutată problema comparării metaeuristicilor în termeni de performanță. Compararea metaeuristicilor este inerent complexă din cauza: i) *Sensibilității la parametri*: În cazul funcțiilor sintetice, algoritmi precum DE tind să fie mai eficienți decât algoritmi fără parametri, dar necesită reglaje fine pentru utilizarea lor în cazul aplicării pe probleme reale. Configurația cu trei parametri a DE necesită o adaptare specifică problemei, limitând caracteristica de “plug and play”; ii) *Supra-adaptarea la funcțiile de test*: Mulți algoritmi au performanțe foarte bune pe funcțiile sintetice, dar nu au o eficiență ridicată în scenarii aplicate,

subliniind riscurile „optimizării pentru funcțiile sintetice”; iv) *Lacune teoretice*: Spre deosebire de optimizarea convexă, majoritatea metaeuristicilor nu dispun de demonstrații riguroase de stabilitate, complicând utilizarea lor în domenii critice din punct de vedere al siguranței.

Urmând concluziile analizei teoretice, în această teză a fost propusă o **metodologie** clară și robustă pentru a răspunde la **Q2** și **Q3**. Cercetarea a folosit plan factorial complet pentru a evalua algoritmi metaeuristici considerând mai multe aspecte precum dimensiuni diferite pentru probleme, variații de algoritmi, funcții de distribuție pentru pasul de inițializare și generarea de numere aleatorii, strategii de mutație și funcții de test diverse. Această abordare a combinat funcții de test sintetice cu probleme ingineresti reale, creând o platformă cuprinzătoare care evaluează atât performanța teoretică, cât și aplicabilitatea practică.

Un aspect nou al studiului realizat este reprezentat de trecerea de la evaluarea centrată pe algoritm la cea centrată pe componente. Prin descompunerea algoritmilor în părțile lor constitutive — variante de distribuție și strategii de mutație — această teză izolează efectele specifice ale acestor componente, în loc să le compare ca entități indivizibile. Această abordare modulară permite o înțelegere mai profundă a mecanismelor fundamentale care determină performanța metaeuristicilor.

Pentru **Q2 (impactul distribuțiilor asupra performanței algoritmilor)**, analiza a arătat că, deși alegerea distribuției influențează traiectoriile algoritmilor, magnitudinea acestui impact variază semnificativ între algoritmi. Beta cu parametri (1, 0.5) și Weibull sunt cele mai performante distribuții. Avantajul Weibull poate fi atribuit alungii sale exponențiale, care crește diversitatea soluțiilor, permițând explorarea atât a regiunilor promițătoare, cât și a celor nepromițătoare.

Sensibilitatea algoritmilor la distribuții variază considerabil. Anumiți algoritmi (DevDS, OriginalSBO) prezintă o sensibilitate ridicată, în timp ce strategiile bazate pe DE manifestă o sensibilitate remarcabil de scăzută. Totodată, a fost observat că modificări la nivel de implementare (OriginalSBO vs DevSBO) pot influența atât performanța generală cât și impactul distribuției și mutației. Pentru variantele DE, includerea mecanismelor de adaptare/auto-adaptare reduc sensibilitatea față de funcția de distribuție utilizată.

Dimensionalitatea reprezintă un alt aspect pentru analiza impactului distribuției, dimensiunile mai mari ducând în general la valori medii mai mari ale valorilor *gap* și la o variabilitate mai mare. Distribuția Beta are o eficiență bună pentru anumite funcții la dimensiuni mici, dar își pierde din avantaje pe măsură ce crește dimensionalitatea. Contrar așteptărilor intuitive, diversitatea mai mare a populației nu este corelată neapărat cu o convergență sau performanță îmbunătățită. Distribuția binomială, de exemplu, menține o diversitate ridicată, dar prezintă rate de convergență lente, indicând că diversitatea nu este un predictor suficient al succesului procesului de optimizare.

Pentru **Q3 (strategii de mutație și implicațiile lor asupra performanței)**, analiza a șapte strategii de mutație (M0 – M6) testate folosind funcțiile specifice competiției CEC 2022 și funcții

ingineresti a indicat că selecția strategiei de mutație influențează semnificativ performanța algoritmului. Strategiile M4 și M6 au demonstrat o reziliență bună împotriva „blestemului dimensionalității”. Pe măsură ce dimensiunea crește, valorile metricii *gap* pentru M4 cresc mai puțin decât la alte strategii, evidențiind robustețea sa în spații complexe cu dimensionalitate mai mare.

Ratele de convergență au variat substanțial între variantele de mutație. Toate variantele au prezentat convergență rapidă în primele iterații, dar s-au plafonat mai devreme la probleme cu dimensionalitate mare. Variantele M1 și M2 au atins zone de stagnare înalte mai rapid, sugerând că ajung mai ușor să fie blocate în optime locale. În schimb, M6 a arătat un potențial de îmbunătățire continuu dacă i se acordă resurse computaționale suplimentare. Relația dintre menținerea diversității și performanță s-a dovedit un aspect complex. Variantele M4 și M5 au menținut constant o diversitate mai mare a populației, indicând capacități de explorare mai puternice, cu o corelație pozitivă între diversitate și performanță pentru M4. În schimb, M1 și M6 au prezentat valori mai scăzute ale diversității, accentuând exploatarea în detrimentul explorării.

Răspunsul algoritmilor la variantele de mutație a variat considerabil. Pentru DS, variantele care combină indivizi curenți, cei mai buni și aleatori (de exemplu, M4, M5) au oferit rezultate optime cu variabilitate minimă. Algoritmii SBO au răspuns dramatic diferit, DevSBO prezentând îmbunătățiri substanțiale față de OriginalSBO. Dintre variantele DE, SSADE a demonstrat o robustețe bună pentru toate variantele de mutație, în timp ce versiunile avansate precum SHADE și L-SHADE au arătat o sensibilitate redusă la selecția mutației.

Analiza statistică a confirmat diferențe semnificative între variantele de mutație, 67% dintre comparațiile pereche prezentând variații semnificative de performanță. Strategiile M4 și M6 s-au dovedit statistic superioare alternativelor pe majoritatea tipurilor de probleme. Pe baza rezultatelor obținute, **s-a propus o nouă strategie de hibridizare** prin aplicarea variantei de mutație M4 pentru a genera o populație care concurează cu cea rezultată din mecanismele clasice utilizate de fiecare algoritm pentru a fi selectată pentru generația următoare. Analiza rezultatelor simulărilor realizate a demonstrat că hibridizarea depășește constant abordările clasice pentru majoritatea algoritmilor și a funcțiilor de referință. Această performanță îmbunătățită provine din combinarea eficientă a mecanismelor complementare de explorare și exploatare, abordând probleme frecvente precum convergența prematură și stagnarea în optime locale. Varianta hibridă DevNMRA a demonstrat performanțe foarte bune, cu cea mai mică valoare medie a metricii *gap*.

În ceea ce privește performanța generală a algoritmilor, LSHADE și OriginalSHADE s-au remarcat ca cei mai eficienți când sunt aplicați pe funcțiile specific competiției CEC 2022, obținând cele mai mici valori medii și deviații standard mici atât pentru problemele din D10 cât și pentru cele din D20. DevSBO a prezentat o performanță bună pentru anumite funcții, menținând un coeficient de variație scăzut indicând o stabilitate bună între rulări. Totuși, nici un algoritm nu a demonstrat

superioritate universală pe toate spațiile de căutare, subliniind importanța selecției algoritmului în funcție de problemă. Totodată, și eficiența computațională a variat dramatic între algoritmi. DevDS a obținut cel mai mic raport de consum de resurse, în timp ce operațiunile intensive ale OriginalSBO au dus la valori ale timpului de execuție ce depășesc 1800 la D20.

Problemele ingineresti cu constrângeri de egalitate au prezentat provocări substanțiale. Ratele de succes au variat dramatic în funcție de tipul problemei, unele probleme fiind rezolvate de toți algoritmi testați, în timp ce altele au rămas dificile chiar și pentru cei mai buni algoritmi.

Aplicarea unor metaeuristici pentru rezolvarea unor probleme reale de inginerie chimică fără modele fenomenologice cunoscute a demonstrat utilitatea lor practică în domenii diverse. Aceste aplicații au inclus procese de tratare a apelor uzate, optimizarea compostării și predicția proprietăților structurilor moleculare complexe. În acest caz, performanța algoritmilor este grupată pe nivele distincte pentru aplicații specifice. Pentru decolorarea rifaximinei, cei mai buni algoritmi (LSHADE, DevSBO, OriginalSHADE) au atins constant eficiență maximă (67.72%) cu o deviație standard zero, demonstrând fiabilitate foarte bună. Pentru decolorarea colorantului eriochrom negru T, șase algoritmi au obținut eficiența maximă de 99.25% cu valori identice ale parametrilor, sugerând convergența către adevăratele optime globale.

Abordările neuro-evolutive au arătat un potențial deosebit pentru procese complexe, neliniare, unde modelarea și optimizarea este dificilă. Strategia ANN-iBFO a surprins mai bine relațiile neliniare complexe decât abordările tradiționale de tip RSM, permițând predicții mai precise și optimizare îmbunătățită în aplicații precum decolorarea compusului de bromcrezol verde. Similar, SSADE a optimizat cu succes arhitecturile rețelelor neuronale artificiale pentru predicția temperaturilor de tranziție de fază ale moleculelor bent-core, cu valori R^2 între 0.89 și 0.98.

7.2. Aspecte originale

Teza stabilește un platformă nouă pentru evaluarea performanței metaeuristicilor considerând mai multe aspecte și pe baza rezultatelor obținute, propune o variantă simplă de hibridizare cu performanțe îmbunătățite. Mai jos sunt sintetizate aspectele inovatoare ale acestei cercetări din trei perspective:

(1) **Abordare metodologică pentru evaluarea metaeuristicilor.** Noutatea metodologiei propuse și strategia aplicată pentru analiza comportamentului metaeuristicilor sunt subliniate de:

- *Metodologia de analiză la nivel de componentă.* Studiul analizează componentele metaeuristicilor, diferind de comparațiile tradiționale aplicate la nivel de algoritmi. Prin descompunerea metaeuristicilor în elemente modulare - precum metode de inițializare (distribuție) și strategii de mutație - cercetarea izolează impactul fiecărui mecanism asupra performanței generale. Această abordare a arătat că utilizarea

distribuțiilor (beta, Weibull) prezintă diferențe relativ mici ale performanței, în ciuda așteptărilor teoretice, în timp ce strategiile de mutație explică 67% din variațiile observate. Indicatorii de diversitate a populației nu arată o corelație constantă cu ratele de convergență. Mai mult, planul experimental factorial complet, care a inclus șapte algoritmi, șase distribuții, șapte strategii de mutație și 12 funcții sintetice plus 6 funcții ingineresti, oferă o granularitate bună în înțelegerea comportamentului metaeuristicilor;

- *Abordare hibridă pentru funcțiile de test.* Integrarea funcțiilor sintetice din competiția CEC2022 cu funcții ingineresti și probleme reale (de exemplu, tratarea apelor uzate, optimizarea compostării) a creat o platformă duală de evaluare. Această metodologie hibridă a evidențiat o serie de aspect importante: i) algoritmi care au performanțe bune pe funcțiile de test sintetice (LSHADE, OriginalSHADE) și-au menținut superioritatea pentru 78% dintre funcțiile ingineresti; ii) constrângerile reale (de exemplu, constrângeri de egalitate în ENG2/ENG3) au împiedicat convergența pentru toate metodele testate, subliniind provocările specifice domeniului;
- *Plan avansat de evaluare statistică.* Depășind statisticile descriptive, analiza din această teză a considerat: i) analize formale pentru testarea ipotezelor în ceea ce privește impactul distribuțiilor și mutațiilor ($p < 0.05$ în 43% dintre comparații); și ii) metrice de performanță care includ diferențe de fitness, rate de convergență, indici de diversitate și cost computațional;

(2) **Contribuții empirice.** Din această perspectivă, principale contribuții sunt:

- Teza demonstrează că alegerea *distribuției* (Weibull vs. beta vs. χ^2 – chi-squared) explică mai puțin de 4% din variația performanței în algoritmi bazați pe DE. Mai mult, sensibilitatea la dimensionalitate se manifestă diferit în funcție de strategie; avantajul distribuției beta scade la problemele D20, în timp ce performanța Weibull se păstrează relativ constantă. Apare un paradox diversitate-performanță, unde distribuția binomială menține diversitatea dar nu și convergența;
- Cele șapte variante de *mutație* testate (M0-M6) au arătat că M4 (current-to-best/rand) și M6 (rand/rand) prezintă cea mai bună scalabilitate dimensională, menținând 92% din calitatea soluției la trecerea de la D10 la D20. Strategiile hibride propuse, care combină mecanisme de explorare/exploatare prin utilizarea M4, reduc timpul de convergență cu 38% față de abordările clasice. Sensibilitatea algoritmilor la pași specifici variază dramatic; de exemplu, SAPDE arată o variație de performanță de 300% între variantele de mutație față de 12% pentru LSHADE;

(3) **Aplicații practice.** O serie de probleme din inginerie chimică au fost selectate ca studii de caz pentru a demonstra eficiența metaeuristiciilor. Principalele direcții au inclus aplicarea acestor algoritmi de optimizare în combinație cu alte metode în contexte diferite:

- *Modelarea neuro-evolutivă.* Integrarea cu rețele neuronale a permis rezolvarea de probleme complexe fără modele fenomenologice și a arătat că: se pot obține modele de predicție TPH pentru compostarea nămolului uleios cu rate de eroare mici; că se poate realiza predicția temperaturilor de tranziție pe baza caracteristicilor fizico-chimice și structurale ale cristalelor lichide; combinarea unei variante îmbunătățite BFO cu NN generează soluții mult mai eficiente în comparație cu RSM aplicate procesului de decolorare al verdelui de bromcrezol;
- *Optimizare eficientă din punct de vedere al resurselor.* Din punct de vedere al performanței, aplicarea diferitelor metaeuristici a permis identificarea unor soluții practice: decolorarea cu carbon activ 100% pentru tratarea lichiorului negru comparativ cu 81.27% în faza de explorare experimentală, adsorbția de rifaximină 67.72% cu deviație standard zero pentru mai mulți algoritmi, limitarea puterii la $300 \text{ W} \cdot \text{dm}^{-3}$ menținând o eficiență a procesului de 95% în cazul eliminării clorofilei A din apele uzate.

7.3. Direcții de cercetare

Teza a analizat o serie de aspecte importante legate de impactul unor operații specifice asupra comportamentului unor metaeuristici. În acest context, sunt identificate mai multe direcții promițătoare pentru cercetări viitoare. Progresul în cercetarea metaeuristiciilor necesită depășirea evaluării centrate pe algoritm către o analiză centrată pe mecanisme. Această schimbare de perspectivă are implicații importante atât din punct de vedere teoretic, cât și din punct de vedere practic. Astfel, extinderea analizei la nivel de componentă pentru alți algoritmi sau clase de metaeuristici ar oferi o înțelegere mai completă a factorilor ce influențează performanță. Similar, includerea problemelor de optimizare dinamice și cu diferite tipuri de constrângeri ar testa robustețea algoritmilor în condiții mai dificile, care reflectă mai bine aplicațiile reale.

Cercetările viitoare ar trebui să exploreze efectele de interacțiune, luând în considerare și alți pași ai algoritmului. Planuri experimentale mai complexe ar putea evidenția relații sinergice sau antagonice între componente, care influențează performanța generală.

Dezvoltarea unor mecanisme adaptive care să selecteze dinamic distribuțiile și strategiile de mutație în funcție de caracteristicile problemei reprezintă o altă direcție promițătoare. Astfel, informațiile obținute din această analiză la nivel de componentă ar permite identificarea componentelor algoritmice care au cea mai bună performanță în condiții specifice.

Bibliografie selectivă

- ABUALIGAH, L. M., KHADER, A. T. & HANANDEH, E. S. 2018. Hybrid clustering analysis using improved krill herd algorithm. *Applied Intelligence*, 48, 4047-4071.
- ALHALABI, W. & DRAGOI, E. N. 2017. Influence of randomization strategies and problem characteristics on the performance of Differential Search algorithm. *Applied Soft Computing*, 61, 88-110.
- ALSAADI, A. A., ALHALABI, W. & DRAGOI, E.-N. 2021. Performance analysis: differential search algorithm based on randomization and benchmark functions. *Data Technologies and Applications*, 55, 313-331.
- BANSAL, J. C., SHARMA, H., JADON, S. S. & CLERC, M. 2014. Spider Monkey Optimization algorithm for numerical optimization. *Memetic Computing*, 6, 31-47.
- BOETTCHER, S. & PERCUS, A. 2000. Nature's way of optimizing. *Artificial Intelligence*, 119, 275-286.
- CHAWLA, M. & DUHAN, M. 2018. Levy Flights in Metaheuristics Optimization Algorithms – A Review. *Applied Artificial Intelligence*, 32, 802-821.
- CHEN, B., CAO, L., CHEN, C., CHEN, Y. & YUE, Y. 2024. A comprehensive survey on the chicken swarm optimization algorithm and its applications: state-of-the-art and research challenges. *Artificial Intelligence Review*, 57, 170.
- CIVICIOGLU, P. 2013. Backtracking Search Optimization Algorithm for numerical optimization problems. *Applied Mathematics and Computation*, 219, 8121-8144.
- CORRIVEAU, G., GUILBAULT, R., TAHAN, A. & SABOURIN, R. 2012. Review and Study of Genotypic Diversity Measures for Real-Coded Representations. *IEEE Transactions on Evolutionary Computation*, 16, 695-710.
- CUEVAS, E. & CIENFUEGOS, M. 2014. A new algorithm inspired in the behavior of the social-spider for constrained optimization. *Expert Systems with Applications*, 41, 412-425.
- DEB, S., TIAN, Z., FONG, S., WONG, R., MILLHAM, R. & WONG, K. K. L. 2018. Elephant search algorithm applied to data clustering. *Soft Computing*, 22, 6035-6046.
- DHIMAN, G. & KUMAR, V. 2017. Spotted hyena optimizer: A novel bio-inspired based metaheuristic technique for engineering applications. *Advances in Engineering Software*, 114, 48-70.
- DRAGOI, E.-N., GODINI, K. & KOOLIVAND, A. 2021. Modeling of oily sludge composting process by using artificial neural networks and differential evolution: Prediction of removal of petroleum hydrocarbons and organic carbon. *Environmental Technology & Innovation*, 21.
- DRĂGOI, E. N., CÂRLESCU, I., PUF, R., VASILIU, T. & EPURE, E.-L. 2023. Neuro-Evolute Modeling of Transition Temperatures for Five-Ring Bent-Core Molecules Derived from Resorcinol. *Crystals*, 13, 583.
- DRAGOI, E. N., CURTEANU, S., LEON, F., AZARIAN, G., GODINI, K., EVA, L., DAFINESCU, V. & TURLIUC, M. D. 2020. Optimization of Chlorophyll A Removal from Wastewaters Using Bio-Inspired Algorithms. *Environmental Engineering and Management Journal*, 19, 311-323.
- DRAGOI, E. N. & DAFINESCU, V. 2021. Review of Metaheuristics Inspired from the Animal Kingdom. *Mathematics*, 9, 2335.
- DRAGOI, E. N. & LEON, F. Classification of non-traditional nature-inspired optimization metaheuristics. Conference of the TUIASI Doctoral School, May 23-24, 2018 Iasi.
- DRAGOI, E. N., SUDITU, G. D. & NECHITA, M. T. 2023. Modeling and optimization of the Rifaximin decolorization process using bacterial foraging optimization and grey worm optimization. 12th International Conference On Environmental Engineering And Management (ICEEM 12), September 13-16 2023 Iasi, Romania.

- DUAN, H. & QIAO, P. 2014. Pigeon-inspired optimization: A new swarm intelligence optimizer for air robot path planning. *International Journal of Intelligent Computing and Cybernetics*, 7, 24-37.
- EMARY, E., ZAWBAA, H. M. & GROSAN, C. 2018. Experienced gray wolf optimization through reinforcement learning and neural networks. *IEEE transactions on neural networks and learning systems*, 29, 681-694.
- HALDAR, V. & CHAKRABORTY, N. 2016. A novel evolutionary technique based on electrolocation principle of elephant nose fish and shark: fish electrolocation optimization. *Soft Computing*, 21, 3827-3848.
- HARIFI, S., KHALILIAN, M., MOHAMMADZADEH, J. & EBRAHIMNEJAD, S. 2019. Emperor Penguins Colony: a new metaheuristic algorithm for optimization. *Evolutionary Intelligence*, 12, 211-226.
- JAIN, M., SINGH, V. & RANI, A. 2019. A novel nature-inspired algorithm for optimization: Squirrel search algorithm. *Swarm and Evolutionary Computation*, 44, 148-175.
- KIRKPATRICK, S., GELATT, C. D., JR. & VECCHI, M. P. 1983. Optimization by simulated annealing. *Science*, 220, 671-80.
- LARA-MONTAÑO, O. D., TOLEDANO-AYALA, M., DRAGOI, E. N. & HERNÁNDEZ, S. 2025. Metaheuristics for the optimization of chemical processes. In: GÓMEZ-CASTRO, F. I. & RICO-RAMÍREZ, V. (eds.) *Optimization in Chemical Engineering: Deterministic, Meta-Heuristic and Data-Driven Techniques*. Gruyter.
- LI, X. D., EPITROPAKIS, M. G., DEB, K. & ENGELBRECHT, A. 2017. Seeking Multiple Solutions: An Updated Survey on Niching Methods and Their Applications. *Ieee Transactions on Evolutionary Computation*, 21, 518-538.
- MA, Z. Q., WU, G. H., SUGANTHAN, P. N., SONG, A. J. & LUO, Q. Z. 2023. Performance assessment and exhaustive listing of 500+nature-inspired metaheuristic algorithms. *Swarm and Evolutionary Computation*, 77, 101248.
- MEHNE, H. H. & MIRJALILI, S. 2018. A parallel numerical method for solving optimal control problems based on whale optimization algorithm. *Knowledge-Based Systems*, 151, 114-123.
- MIRJALILI, S. & LEWIS, A. 2016. The Whale Optimization Algorithm. *Advances in Engineering Software*, 95, 51-67.
- MORALES-CASTAÑEDA, B., ZALDÍVAR, D., CUEVAS, E., FAUSTO, F. & RODRÍGUEZ, A. 2020. A better balance in metaheuristic algorithms: Does it exist? *Swarm and Evolutionary Computation*, 54, 100671.
- PINTILIE, L., DRĂGOI, E. N., SUDITU, G. D., NECHITA, M. T. & DAFINESCU, V. Eriochrome Black T Decolorization with Active Carbon and Hv/TiO_2 : Modeling and Process Optimization. The 7th International Conference on Environmental Pollution, Treatment and Protection, Lisbon, Portugal, 2022a.
- PINTILIE, L., SAVA, A.-E., DRAGOI, E. N. & TEODOSIU, C. 2022b. Application of Response Surface Method for Optimization of Eriochrome Black T Removal by Adsorption. *Buletinul Institutului Politehnic din Iași*, 68, 37-51.
- QAIS, M. H., HASANIEN, H. M. & ALGHUWAINEM, S. 2018. Augmented Grey Wolf Optimizer for Grid-connected PMSG-based Wind Energy Conversion Systems. *Applied Soft Computing*.
- STEGHERR, H., HEIDER, M. & HÄHNER, J. 2022. Classifying Metaheuristics: Towards a unified multi-level classification system. *Natural Computing*, 21, 155-171.
- TROJOVSKÝ, P. 2023. A new human-based metaheuristic algorithm for solving optimization problems based on preschool education. *Scientific Reports*, 13, 21472.
- YAZDANI, S. & HADAVANDI, E. 2018. LMBO-DE: a linearized monarch butterfly optimization algorithm improved with differential evolution. *Soft Computing*, 1-15.
- YU, J. J. Q. & LI, V. O. K. 2015. A social spider algorithm for global optimization. *Applied Soft Computing*, 30, 614-627.
- YU, J. J. Q. & LI, V. O. K. 2016. A social spider algorithm for solving the non-convex economic load dispatch problem. *Neurocomputing*, 171, 955-965.
- .

A1. Anexa 1. Lista de lucrări

1. Lucrări publicate în reviste cotate ISI (Web of Science cu factor de impact)

- Dragoi, E.N.; Dafinescu, V. Review of Metaheuristics Inspired from the Animal Kingdom. *Mathematics* **2021**, 9, 2335, doi:10.3390/math9182335. Zonă roșie, Q1, factor de impact la data publicării 2.25.
- Dragoi, E.N.; Curteanu, S.; Leon, F.; Azarian, G.; Godini, K.; Eva, L.; Dafinescu, V.; Turliuc, M.D. Optimization of Chlorophyll A Removal from Wastewaters Using Bio-Inspired Algorithms. *Environmental Engineering and Management Journal* **2020**, 19, 311-323. Q4, factor de impact la data publicării 0.17.
- Alhalabi, W.; Dragoi, E.N. Influence of randomization strategies and problem characteristics on the performance of Differential Search algorithm. *Applied Soft Computing* **2017**, 61, 88-110, doi:10.1016/j.asoc.2017.08.001. Zonă roșie, Q1, factor de impact la data publicării 2.86.

2. Lucrări publicate în reviste indexate BDI

- Alsaadi, A.A.; Alhalabi, W.; Dragoi, E.-N. Performance analysis: differential search algorithm based on randomization and benchmark functions. *Data Technologies and Applications* **2021**, 55, 313-331, doi:10.1108/DTA-05-2018-0043.

3. Capitole de carte

- Lara-Montaño, O.D.; Toledano-Ayala, M.; Dragoi, E.N.; Hernández, S. Metaheuristics for the optimization of chemical processes. In *Optimization in Chemical Engineering: Deterministic, Meta-Heuristic and Data-Driven Techniques*, Gómez-Castro, F.I., Rico-Ramírez, V., Eds.; Walter de Gruyter GmbH & Co KG: 2025; p. 113

4. Comunicări la conferințe naționale sau internaționale

- Dragoi, E.N.; Suditu, G.D.; Nechita, M.T. Modeling and optimization of the Rifaximin decolorization process using bacterial foraging optimization and grey worm optimization. In Proceedings of the 12th International Conference On Environmental Engineering And Management (ICEEM 12), Iasi, Romania, September 13-16, **2023**.
- Pintilie, L.; Drăgoi, E.N.; Suditu, G.D.; Nechita, M.T.; Dafinescu, V. Eriochrome Black T Decolorization with Active Carbon and H_v/TiO₂: Modeling and Process Optimization. In Proceedings of the The 7th International Conference on Environmental Pollution, Treatment and Protection (ICEPTP'22), Lisbon, Portugal (Virtual Conference) **2022**. doi: 10.11159/iceptp22.188
- Drăgoi, E.N.; Suditu, G.D.; Nechita, M.-T. Artificial Intelligence Based Modelling and Optimization of a Hybrid Equipment for Wastewater Treatment. In Proceedings of the International Conference on Management and Industrial Engineering, **2021**; pp. 169-177.
- Drăgoi, E.N.; Leon, F. Classification of non-traditional nature-inspired optimization metaheuristics. In Proceedings of the Conference of the TUIASI Doctoral School, May 23-24, Iasi, **2018**.

5. Alte lucrări care susțin rezultatele obținute

- Drăgoi, E.N.; Cârlescu, I.; Puf, R.; Vasiliu, T.; Epure, E.-L. Neuro-Evolutive Modeling of Transition Temperatures for Five-Ring Bent-Core Molecules Derived from Resorcinol. *Crystals* **2023**, *13*, 583. Zonă galbenă, Q2, factor de impact la data publicării 2.7.
- Suditu, G.D.; Dragoi, E.N.; Apostica, A.G.; Manaila, A.M.; Radu, V.M.; Puitel, A.C.; Nechita, M.T. Artificial Intelligence-Based Tools for Process Optimization: Case Study- Bromocresol Green Decolorization with Active Carbon. *Adsorption Science & Technology* **2022**, *2022*, 8110436, doi:10.1155/2022/8110436. Zonă galbenă, Q2, factor de impact la data publicării 4.37.
- Pintilie, L.; Sava, A.-E., Dragoi, E. N. & Teodosiu, C.. Application of Response Surface Method for Optimization of Eriochrome Black T Removal by Adsorption. *Buletinul Institutului Politehnic din Iași*, **2022**, *68*, 37-51, doi:10.5281/zenodo.7539854.
- Nechita, M.T.; Suditu, G.D.; Puitel, A.C.; Dragoi, E.N. Differential evolution-based optimization of corn stalks black liquor decolorization using active carbon and TiO(2)/UV. *Scientific Reports* **2021**, *11*, 18481, doi:10.1038/s41598-021-98006-8. Zonă roșie, Q1, factor de impact la data publicării 4.38.
- Dragoi, E.-N.; Godini, K.; Koolivand, A. Modeling of oily sludge composting process by using artificial neural networks and differential evolution: Prediction of removal of petroleum hydrocarbons and organic carbon. *Environmental Technology & Innovation* **2021**, *21*, doi:10.1016/j.eti.2020.101338. Zonă roșie, Q1, factor de impact la data publicării 5.26.

6. Proiecte de cercetare

- *Artificial intelligence based modelling and optimization of a hybrid system developed through symbiosis of multiple wastewater treatment methods*, finanțat de UEFISCDI, PN-III-P4-ID-PCE-2020-1052, număr 58/2021, 2021-2023, director de proiect Drăgoi E.N