

IOSUD – UNIVERSITATEA „DUNĂREA DE JOS” DIN GALAȚI

Școala doctorală de Științe Fundamentale și Inginerești



TEZĂ DE DOCTORAT

Rezumatul tezei de doctorat

CONDUCEREA AUTOMATĂ AVANSATĂ A PROCESELOR COMPLEXE

Doctorand,

Ing. Liliana – Maria GHINEA

Conducător științific,

Prof. Dr. Ing. Marian BARBU

Conducător științific în cotutelă,

Prof. Dr. Ion NECOARĂ

Seria: I8: Ingineria Sistemelor Nr. 11

GALAȚI

2024



TEZĂ DE DOCTORAT

Rezumatul tezei de doctorat

CONDUCEREA AUTOMATĂ AVANSATĂ A PROCESELOR COMPLEXE

Doctorand

Ing. Liliana – Maria GHINEA

Președinte

Prof. dr. ing. Eugen-Victor-Cristian RUSU

Membru corespondent al Academiei Române

Universitatea „Dunărea de Jos” din Galați

Conducător științific,

Prof univ.dr.ing. Marian BARBU

Universitatea „Dunărea de Jos” din Galați

Conducător științific în cotutelă

Prof univ.dr. Ion NECOARĂ

Universitatea Națională de Știință și Tehnologie

Politehnica din București

Referenți științifici

Prof. dr. ing. Ramon VILANOVA ARBOS

Universitat Autònoma de Barcelona, Spania

Prof. dr. ing. Dan SELIȘTEANU

Universitatea din Craiova

Prof. dr. ing. Constantin Florin CĂRUNTU

Universitatea Tehnică „Gheorghe Asachi” din Iași

Seria: I8: Ingineria Sistemelor Nr. 11

GALAȚI

2024

Seriile tezelor de doctorat susținute public în UDJG începând cu 1 octombrie 2013 sunt:

Domeniul fundamental ȘTIINȚE INGINEREȘTI

- Seria I 1: **Biotehnologii**
- Seria I 2: **Calculatoare și tehnologia informației**
- Seria I 3: **Inginerie electrică**
- Seria I 4: **Inginerie industrială**
- Seria I 5: **Ingineria materialelor**
- Seria I 6: **Inginerie mecanică**
- Seria I 7: **Ingineria produselor alimentare**
- Seria I 8: **Ingineria sistemelor**
- Seria I 9: **Inginerie și management în agricultură și dezvoltare rurală**

Domeniul fundamental ȘTIINȚE SOCIALE

- Seria E 1: **Economie**
- Seria E 2: **Management**
- Seria E 3: **Marketing**
- Seria SSEF: **Știința sportului și educației fizice**
- Seria SJ: **Drept**

Domeniul fundamental ȘTIINȚE UMANISTE

- Seria U 1: **Filologie- Engleză**
- Seria U 2: **Filologie- Română**
- Seria U 3: **Istorie**
- Seria U 4: **Filologie - Franceză**

Domeniul fundamental MATEMATICĂ ȘI ȘTIINȚE ALE NATURII

- Seria C: **Chimie**

Domeniul fundamental ȘTIINȚE BIOMEDICALE

- Seria M: **Medicină**
- Seria F: **Farmacie**

Cuprinsul tezei de doctorat

Introducere	XIV
Abrevieri	XVIII
Lista figurilor	XIX
Lista tabelelor	XXI
1. Stadiul actual	1
1.1 Generalități despre stațiile de tratare a apelor uzate	1
1.2 Aplicații privind metodele de control optimal utilizate în stațiile de epurare	2
1.3 Aplicații privind metodele de optimizare utilizate în controlul stațiilor de tratare a apelor uzate	5
1.4 Aplicații privind metodele de Deep Learning utilizate în recunoașterea și detectarea defectelor din stațiile de epurare	6
2. Modelarea proceselor complexe de tratare a apelor uzate	9
2.1 Sistemul de tratare a apelor uzate cu nămol activ	9
2.2 Benchmark Simulation Model no. 1 (BSM1)	12
2.3 Benchmark Simulation Model no. 2 (BSM2)	15
3. Control optimal	17
3.1 Generalități	17
3.2 Control optimal neliniar	19
3.2.1 ACADO Toolkit	20
3.2.2 Studiu de caz - ACADO Toolkit	21
3.2.3 Discretizare	24
3.2.4 Problema NLP	25
3.2.5 Studiu de caz - NLP	25
3.3 Control optimal liniar	28
3.3.1 Liniarizare	29
3.3.2 Forma QP	30
3.3.3 Studiu de caz - model liniarizat	33
3.3.4 Comparatie - ACADO vs. model liniarizat	35
3.4 Concluzii	37
4. Algoritmi de optimizare pentru MPC liniar	40
4.1 Generalități	40
4.2 Notatii și definiții	41
4.3 Metoda Coordinate Descent	42
4.4 Metoda Augmented Lagrangian și strategia MPC	45
4.5 Algoritmi pentru problema interioară	50
4.5.1 Metoda CD	51

4.5.2 Metoda APCG	52
4.5.3 Metoda R-APPROX	55
4.6 Studiu de caz	57
4.6.1 Sisteme liniare și neliniare	58
4.6.2 Rezultatele simulărilor	69
4.7 Concluzii	73
5. Metode de Deep Learning pentru recunoașterea și detectarea defectelor	75
5.1 Generalități	75
5.2 Defecte injectate pe senzorul de oxigen dizolvat	82
5.2.1 Defectul de pompă de aerare	83
5.2.2 Defectul drift	84
5.2.3 Defectul bias	84
5.2.4 Defectul stuck	85
5.2.5 Defectul spike	85
5.2.6 Defectul precision degradation	86
5.3 Recunoaștere de defecte	86
5.4 Studiu de caz	89
5.5 Detectare de defecte	96
5.6 Metode semi-supervizate	98
5.6.1 Metode clasice semi-supervizate	98
5.6.2 Metode de Deep learning semi-supervizate	103
5.7 Studiu de caz	107
5.8 Concluzii	120
6. Concluzii finale	123
6.1 Contribuții originale	125
6.2 Diseminarea rezultatelor	126
6.3 Direcții viitoare de cercetare	128
Bibliografie	131
Anexa 1	144
Anexa 2	148

Cuprinsul rezumatului

Introducere	IX
1. Stadiul actual	1
1.1 Generalități privind sistemele de tratare a apelor uzate	1
1.2 Aplicații privind metodele de control optimal utilizate în stațiile de epurare	1
1.3 Aplicații privind metodele de optimizare utilizate în controlul stațiilor de tratare a apelor uzate	1
1.4 Aplicații privind metodele de Deep Learning utilizate în recunoașterea și detectarea defectelor din stațiile de epurare	2
2. Modelarea proceselor complexe de tratare a apelor uzate	3
2.1 Sistemul de tratare a apelor uzate cu nămol activ	3
2.2 Benchmark Simulation Model no. 1 (BSM1)	3
2.3 Benchmark Simulation Model no. 2 (BSM2)	4
3. Control optimal	5
3.1 Generalități	5
3.2 Control optimal neliniar	5
3.2.1 ACADO Toolkit	5
3.2.2 Studiu de caz - ACADO Toolkit	5
3.2.3 Discretizare	6
3.2.4 Problema NLP	7
3.2.5 Studiu de caz - NLP	7
3.3 Control optimal liniar	8
3.3.1 Liniarizare	8
3.3.2 Forma QP	9
3.3.3 Case study - Linearized model	9
3.3.4 Comparație - ACADO vs. modelul liniarizat	10
3.4 Concluzii	10
4. Algoritmi de optimizare pentru MPC liniar	11
4.1 Generalități	11
4.2 Notatii și definiții	11
4.3 Metoda Coordinate Descent	11
4.4 Metoda Augmented Lagrangian și strategia MPC	11
4.5 Algoritmi pentru problema interioară	15
4.5.1 Metoda CD	15
4.5.2 Metoda APCG	16
4.5.3 Metoda R-APPROX	17
4.6 Studiu de caz	18

4.6.1 Sisteme liniare și neliniare	19
4.6.2 Rezultatele simulărilor	19
4.7 Concluzii	20
5. Metode de Deep Learning pentru recunoașterea și detectarea defectelor	21
5.1 Generalități	21
5.2 Defecte injectate pe senzorul de oxigen dizolvat	21
5.2.1 Defectul de pompă de aerare	22
5.2.2 Defectul drift	22
5.2.3 Defectul bias	22
5.2.4 Defectul stuck	23
5.2.5 Defectul spike	23
5.2.6 Defectul precision degradation	23
5.3 Recunoaștere de defecte	23
5.4 Studiu de caz	23
5.5 Detectare de defecte	26
5.6 Metode semi-supervizate	26
5.6.1 Metode clasice semi-supervizate	26
5.6.2 Metode de Deep Learning semi-supervizate	27
5.7 Studiu de caz	27
5.8 Concluzii	31
6. Concluzii finale	32
6.1 Contribuții originale	32
6.2 Diseminarea rezultatelor	33
6.3 Direcții viitoare de cercetare	34

Introducere

În această eră caracterizată prin progrese tehnologice rapide și o interacțiune din ce în ce mai complexă a variabilelor în cadrul diferitelor sisteme, metodele eficiente de optimizare și control sunt acum mai importante ca niciodată. Împletirea strategiilor de optimizare, control optimal și metodelor de Deep Learning a devenit o tehnică inovativă pentru rezolvarea diverselor provocări complexe apărute în lumea reală. Examinând sinergia și potențialul pe care aceste strategii îl au în numeroase industrii, lucrarea de față are ca scop depășirea limitelor acestora și abordarea diverselor aplicații care pot evidenția importanța lor.

Obiectivul principal al aplicațiilor este rezolvarea sistemelor care reprezintă stații de tratare a apelor uzate (WWTPs). Acestea sunt esențiale pentru menținerea sănătății publice, protejarea mediului înconjurător și garantarea sustenabilității resurselor de apă. Apa uzată provine din diverse surse, cum ar fi proprietățile private, fabricile și entitățile comerciale. Înainte de a fi deversată în mediul înconjurător sau folosită în scopuri nepotabile, apa uzată este tratată și filtrată în aceste stații de epurare. Trebuie astfel să asigurăm faptul că apa care ajunge în apele naturale nu este dăunătoare mediului și se supune regulilor și legislațiilor în vigoare. Scopurile principale ale stațiilor de tratare sunt atât îndepărtarea poluanților și prevenția poluării apelor, cât și protejarea sănătății populației și a mediului înconjurător. Deoarece apa uzată conține diverși contaminanți, cum ar fi materia organică, nutrienți (nitrogen și fosfor), patogeni, chimicale și solide în suspensie, stațiile de epurare trebuie să aplice diverse tipuri de tratament asupra apei uzate, pentru a se asigura că apa deversată în râuri, lacuri etc. nu este nocivă.

Complexitatea design-ului stațiilor de epurare variază în funcție de volumul și natura apei uzate, de specificul mediului înconjurător și de tehnologia valabilă. Stațiile de epurare moderne folosesc frecvent tehnici de control optimal, algoritmi de optimizare și metode de Deep Learning, atât pentru a îmbunătăți productivitatea și calitatea apei tratate, cât și pentru a identifica și preveni posibile anomalii care împiedică eficiența instalației de tratare. Astfel, stațiile de epurare reprezintă infrastructuri foarte importante care înlesnesc accesul la apă curată, ecosisteme protejate și siguranța sănătății oamenilor.

Controlul optimal, un subdomeniu în teoria controlului, joacă un rol important în rezolvarea problemelor derivate din sisteme complexe și oferă abordări sistematice pentru identificarea unui set ideal de date de intrare care ghidează sistemul către obiectivele dorite. Controlul optimal a avansat mult față de nivelul de unde a pornit datorită progreselor din domeniul matematicii, a puterii de calcul și a complexității sistemelor. Metodele de control optimal au demonstrat că sunt versatile și eficiente în numeroase domenii ingineresti, cum ar fi vehiculele aerospațiale, sisteme robotizate, procese chimice, aplicații biomedicale, economie și industria tratării apelor uzate. Tehnicile de control optimal sunt utilizate pentru reducerea costurilor, maximizarea eficienței și controlul sistemelor caracterizate prin dinamici și setări incerte. Aceste aspecte sunt realizate printr-un reglaj atent al datelor de intrare de-a lungul unui orizont de timp specificat. Tehnicile de control optimal reprezintă un domeniu promițător în optimizarea proceselor de tratare a apelor uzate, ceea ce va conduce la eficiență sporită, impact negativ scăzut asupra mediului și conservarea resurselor valoroase.

Algoritmii de optimizare reprezintă fundația aplicațiilor care presupun metode moderne de rezolvare și de luare a deciziilor. Acești algoritmi influențează modul în care sunt abordate aceste probleme, alocăm resurse și maximizăm rezultatele obținute în urma rezolvării diferitelor probleme apărute în cadrul a diverse discipline. Domeniul algoritmilor de optimizare este în continuă trans-

formare, dovedindu-și importanța într-un context complicat, în care există constrângeri asupra resurselor. Metodele de optimizare reprezintă o arie importantă de studiu pentru cercetători, având ca țintă obținerea eficienței alocării resurselor. Acești algoritmi sunt în sine potriviți pentru aplicații matematice, însă în combinație cu tehnicile de control optimal, ei devin de asemenea corespunzători pentru procesele de tratare a apelor uzate. Prin complexitatea lor, stațiile de epurare inovează astfel domeniul algoritmilor de optimizare, solicitând o abordare inedită, în combinație cu tehnicile de control optimal.

Metodele de Deep Learning stau la baza unui nou domeniu, revoluționând avansul tehnologic din cadrul ingineriei. În afară de îmbunătățirea acurateții analizei predictive, modelele de Deep Learning aduc un plus prin capacitatea lor de a identifica tipare complicate și de a extrage informații importante din seturi de date de mari dimensiuni. Inovația adusă de modelele de Deep Learning este vizibilă în domenii precum programe cu experiență personalizată a utilizatorului, diagnostic medicală și stații de tratare a apelor uzate. Este deja cunoscut faptul că stațiile de tratare joacă un rol foarte important în protejarea mediului înconjurător și a sănătății publice. Acest fapt, și de asemenea nivelul de complexitate al acestor stații de tratare, cauzează adesea comportamente defectuase. Metodele tradiționale de identificare și detectare a defectelor întâmpină de obicei complicații în cadrul acestor sisteme. Metodele de Deep Learning, însă, intervin în combaterea acestor provocări, prin capacitatea lor de a manipula seturi de date de dimensiuni mari, relații complicate între variabile și tipare în continuă evoluție.

În cadrul lucrării de față, partea de cercetare se bazează pe intersecția dintre tehnici ingineresti aplicate asupra mediului și tehnologii de ultimă generație. Principala zonă de interes este reprezentată de combinația dintre algoritmi de optimizare, strategii de control optimal și metode de Deep Learning, prin care sunt abordate o mare varietate de aplicații care au legătură cu sistemele de tratare a apelor uzate. Administrarea eficientă a apelor uzate duce la buna întreținere a ecosistemului și la protejarea sănătății oamenilor, în condițiile în care populația lumii este în continuă creștere și urbanizare. De aceea, străduința este spre ajustarea cât mai precisă a parametrilor de operare ai stațiilor de epurare. Astfel, incluzând mecanisme de control optimal, sistemele de tratare se păstrează durabile și stabile în ciuda condițiilor de intrare schimbătoare, iar prin utilizarea algoritmilor de optimizare, se îmbunătățește eficiența acestor sisteme, și se reduce consumul de resurse. Mai mult, explorând domeniul metodelor de Deep Learning, se utilizează rețelele neuronale pentru recunoașterea și identificarea posibilelor anomalii care împiedică funcționarea normală a stațiilor de tratare. Acest fapt rezultă în stații de epurare mai inteligente și mai sustenabile. Cercetarea din această teză de doctorat reprezintă un pas important către abordarea cerințelor din ce în ce mai complicate ale acestor sisteme de tratare a apei uzate, în același timp ținându-se cont de reducerea efectelor negative care pot apărea în urma acestor procese.

Primul obiectiv al lucrării de față este aplicarea atât a tehnicilor de control optimal liniar, cât și de neliniar pentru a stabili un sistem de tratare a apelor uzate cu patru stări. Acest sistem reprezintă o aplicație practică utilă, deoarece sistemele de tratare a apei uzate au frecvent dinamici complexe care necesită un control precis care să le asigure stabilitate și eficiență. Se dorește îmbunătățirea performanței acestui sistem și garantarea robusteții chiar și în condiții schimbătoare ale influentului, și acest lucru va fi realizat prin cercetarea și aplicarea noțiunilor de control optimal liniar și neliniar. Acest obiectiv este un prim pas esențial în misiunea de a îmbunătăți rezistența stațiilor de epurare, pentru a obține în final tehnici de tratare sustenabile și prietenoase cu mediul înconjurător.

A doua direcție a lucrării se referă la domeniul algoritmilor de optimizare. Metoda Coordinate

Descent și strategia Augmented Lagrangian sunt două exemple de algoritmi de optimizare care sunt cercetați și aplicați în lucrarea de față. Scopul este de a aplica aceste metode moderne pentru a rezolva probleme de programare pătratică (QP) care derivă din diverse sisteme liniare și neliniare, printre care se află și sistemul de tratare a apelor uzate cu patru stări. Acești algoritmi, care sunt o parte importantă a studiului, asigură un proces de optimizare eficientă și o ajustare precisă a parametrilor complecși din cadrul acestor sisteme. Prin rezolvarea cu succes a problemelor de tip QP, se pot găsi cele mai bune soluții pentru obiectivele stabilite, anume alocarea eficientă a resurselor și controlul proceselor, în același timp respectându-se și regulile stricte ale mediului. Acest studiu reprezintă un pas important către dezvoltarea stațiilor de tratare mai eficiente și mai sustenabile, având ca scop final protejarea mediului și a sănătății publice.

Al treilea obiectiv al lucrării noastre este reprezentat de examinarea în amănunt a anomaliilor care pot apărea în stațiile de tratare a apelor uzate, împiedicându-le operarea normală. Pentru a rezolva această problemă, sunt folosite tehnicile de Deep Learning, atât pentru recunoașterea, cât și pentru identificarea numeroaselor tipuri de defecte care pot împiedica funcționarea normală a senzorilor importanți aflați în stațiile de epurare. Focusul este pe Benchmark Simulation Model No. 2 (BSM2), un sistem de referință foarte cunoscut și des utilizat în simulări. Pentru a menține eficiența și durabilitatea stațiilor de tratare a apelor uzate, trebuie construite sisteme solide de recunoaștere și identificare a defectelor. Aceste sisteme se folosesc de tehnicile de Deep Learning, și se aplică asupra sistemului BSM2. Pe termen lung, studiul asigură o funcționare consistentă și o performanță ridicată a stațiilor de tratare a apelor uzate, având astfel implicații pozitive asupra protejării mediului și a sănătății publice. De asemenea, studiul rezultă în dezvoltarea sistemelor inteligente și dinamice care pot identifica rapid anomaliile și pot rezolva eficient problemele cauzate de acestea.

Structura tezei de doctorat include o introducere, cinci capitole și capitolul de concluzii finale.

În Introducere sunt prezentate direcțiile de cercetare ale lucrării de doctorat, și de asemenea importanța atât a stațiilor de tratare a apelor uzate în sine, cât și a metodelor de optimizare, a strategiilor de control optimal și a tehnicilor de Deep Learning aplicate asupra acestora.

Capitolul 1 oferă o prezentare generală a direcțiilor de cercetare alese în lucrarea de față, diverse studii realizate de-a lungul anilor asupra sistemelor de tratare a apelor uzate, și utilizarea strategiilor de control optimal, a algoritmilor de optimizare și a tehnicilor de Deep Learning asupra acestora. Este abordată importanța stabilizării sistemelor de tratare și a recunoașterii și identificării defectelor care pot apărea în stațiile de epurare, devenind rapid obstacole în acestea și având consecințe devastatoare asupra mediului.

Capitolul 2 explorează diverse sisteme de tratare a apelor uzate și oferă descrieri ale diferitelor componente ale acestora. Se începe astfel cu sistemul simplu de tratare a apelor uzate cu patru stări, prezentând schema acestuia și modelul matematic care îl descrie. De asemenea, sunt introduse două sisteme de referință consacrate în domeniu, și anume Benchmark Simulation Model No. 1 (BSM1) și Benchmark Simulation Model No. 2 (BSM2). Aceste modele sunt esențiale pentru modelarea și evaluarea eficacității procedurilor de tratare a apelor uzate, și ajută la optimizarea acestora. Acest capitol oferă o bază utilă pentru înțelegerea complexității acestor stații de epurare.

În Capitolul 3 sunt prezentate tehnici de control optimal aplicate asupra sistemului de tratare cu patru stări. Se începe capitolul cu generalități din teoria controlului optimal liniar, noțiuni despre procesele de liniarizare și discretizare a sistemelor, strategia Model Predictive Control, și de asemenea cu descrierea programului ACADO Toolkit care conține o interfață MATLAB utilizată pentru sistemele neliniare. Mai mult, sunt prezentate generalități din teoria controlului optimal

neliniar și din domeniul problemelor de programare neliniară (NLP). În secțiunile destinate studiului de caz, scopul este de a stabili sistemul de tratare cu patru stări, atât folosind ACADO Toolkit, cât și propriul nostru algoritm scris în MATLAB, și de a compara cele două abordări. De asemenea, este abordată și rezolvarea unei probleme scrise în formă NLP, problemă bazată pe același sistem de tratare a apelor uzate.

În Capitolul 4 este avansată cercetarea prin combinarea algoritmilor de optimizare, și anume metoda Coordinate Descent și abordarea Augmented Lagrangian cu strategia Model Predictive Control. Scopul este de a crea mai mulți algoritmi în MATLAB care rezolvă problema de optimizare pătratică descrisă în detaliu în acest capitol. În secțiunea destinată studiului de caz, sunt prezentate șapte sisteme (cinci liniare și două neliniare, printre care se află și sistemul de tratare cu patru stări descris în Capitolul 2) și performanța algoritmilor aplicați asupra lor. Acest capitol încheie partea de cercetare realizată asupra sistemelor simple de tratare a apelor uzate.

Capitolul 5 îmbunătățește cercetarea, ducând-o în domeniul metodelor de Deep Learning. În acest capitol este studiat sistemul BSM2, sunt injectate diverse defecte în senzorul de oxigen dizolvat și în pompa de aerare, apoi sunt evaluate diferite metode de recunoaștere și identificare a acestor anomalii. Astfel, sunt comparate performanțele a două rețele neuronale artificiale, și anume Feedforward Neural Network (FFNN) și Convolutional Neural Network (CNN), pentru recunoașterea defectelor injectate, și sunt prezentate valorile metricilor de performanță ale fiecăreia. În ceea ce privește detectarea defectelor în stațiile de tratare a apelor uzate, sunt studiate cinci strategii, trei metode clasice semi-supervizate, și anume Isolation Forest (IF), Local Outlier Factor (LOF), One-Class Support Vector Machine (OCSVM), precum și două tehnici autoencoder, mai exact Multilayer Perceptron Autoencoder (MLP-AE) și Convolutional Autoencoder (Conv-AE). Scopul este crearea unui sistem de detecție a defectelor care folosește cea mai bună metodă semi-supervizată.

În Capitolul 6, care cuprinde secțiunea de Concluzii Finale ale acestei lucrări de doctorat, sunt prezentate contribuțiile originale din această teză, direcțiile viitoare de cercetare și diseminarea rezultatelor.

Capitolul 1. Stadiul actual

1.1 Generalități privind sistemele de tratare a apelor uzate

Sistemele de tratare a apelor uzate constituie o rețea vitală din cadrul infrastructurii urbane complexe unde ritmul vieții cotidiene curge fără întreruperi. Aceste sisteme lucrează neîncetat pentru a transforma apa uzată (apa care a fost contaminată prin uz casnic, industrial, comercial sau prin activități de agricultură) într-o resursă care poate coexista cu natura, ajutând astfel la protejarea mediului înconjurător [2].

1.2 Aplicații privind metodele de control optimal utilizate în stațiile de epurare

În ceea ce privește aplicarea tehnicilor de Model Predictive Control pe sisteme de tratare a apelor uzate, un articol care stă la baza cercetării din lucrarea noastră este [3]. Autorii acesteia prezintă un instrument foarte important, ACADO Toolkit, care implementează unelte de control automat și optimizare dinamică. Articolul prezintă diferite clase de probleme pe care ACADO Toolkit le poate rezolva cu succes, printre ele aflându-se și aplicarea strategiei de MPC asupra unui sistem de tratare a apelor uzate. ACADO Toolkit poate fi utilizat pentru găsirea soluției unei probleme de control optimal, și un astfel de exemplu este întâlnit în lucrarea [4]. În aceasta, ACADO Toolkit este folosit pentru găsirea soluției numerice pentru o problemă de control optimal cu timp minim. O altă lucrare importantă este [5], unde autorii prezintă un controller robust bazat pe tehnici din teoria QFT aplicate asupra proceselor biotehnologice. La fel de importante sunt lucrările [6] și [7], care prezintă controller de tip QFT construit pentru a asigura proprietăți bune pentru trei condiții ale stațiilor de tratare a apelor uzate, și anume condiții normale, de ploaie și de secetă. Cele două modele de tratare sunt cel din [8], și respectiv ASM1 [9].

În ceea ce privește aplicarea controlului optimal neliniar pe sisteme de tratare a apelor uzate este important să menționăm lucrarea [10], care propune un algoritm de control predictiv neliniar hibrid, care minimizează energia folosită și în același timp ia în calcul necesarul de oxigen al stației de epurare, care este variabil, precum și constrângerilor proceselor care au loc. Foarte importantă pentru cercetarea noastră este lucrarea [11], care abordează importanța creșterii eficienței unei stații de tratare a apelor uzate prin reducerea concentrației de poluanți din efluent, și rezolvă această problemă transformând-o într-una de programare neliniară.

În ceea ce privește aplicarea tehnicii de Model Predictive Control neliniar (NMPC) pe sisteme de tratare a apelor uzate și alte domenii, o lucrare importantă este [12], care prezintă proiectarea unei scheme de NMPC bazată pe modelul neliniar al bioprocesului de producție de lipază care are loc în interiorul unui bioreactor Fed-batch. Alt articol este [13], unde autorii se ocupă de proiectarea unei legi de control predictiv pentru procesul de cultură a microalgelor, pentru a regla concentrația de biomasă la un punct de referință ales.

1.3 Aplicații privind metodele de optimizare utilizate în controlul stațiilor de tratare a apelor uzate

În ceea ce privește studiul metodelor de optimizare, există mai multe lucrări importante. Una dintre ele este [14], unde autorii iau în considerare probleme de optimizare de mari dimensiuni și propun două strategii de rezolvare bazate pe Coordinate Descent aleatoriu. Strict în contextul aplicării algoritmilor de optimizare pentru controlul stațiilor de tratare a apelor uzate, există mai

multe articole relevante. Unul dintre ele este [15], care prezintă o metodă fără derivare, cu scopul de a rezolva o problemă de optimizare care rezultă din modelul matematic al unui sistem de tratare a apelor uzate cu nămol activ, și anume Activated Sludge Model no. 1 (ASM1). Obiectivul este minimizarea investiției și a costurilor operaționale, iar metoda este bazată pe strategia Augmented Lagrangian de rezolvare a subproblemelor constrânse. Similare sunt articolele [16] și [17], unde autorii propun metode fără derivare de rezolvare a problemelor de optimizare constrânse, care apar din modelarea stațiilor de epurare. Algoritmii prezentați în aceste lucrări se bazează pe funcții Augmented Lagrangian al căror scop este să rezolve subprobleme constrânse, să penalizeze soluții nefezabile sau să caute punctul de optim global.

1.4 Aplicații privind metodele de Deep Learning utilizate în recunoașterea și detectarea defectelor din stațiile de epurare

Există multe lucrări care au ca subiect recunoașterea de defecte în stațiile de tratare a apelor uzate. Una dintre ele este [18], în care autorii folosesc rețele neuronale artificiale (ANNs) pentru a identifica șase tipuri de defecte pe senzori și actuatori, unul dintre defecte fiind o anomalie pe senzorul de oxigen dizolvat. Acuratețea generală de clasificare de 97.2% demonstrează astfel abilitatea rețelei neuronale artificiale utilizate de a identifica anomaliile. Pentru a recunoaște defecte complicate și modificări în valorile semnalului de ieșire a senzorului, autorii lucrării [19] folosesc metode de tip autoencoder. Mai mult, lucrarea [20] prezintă o strategie de monitorizare bazată pe Adaptive Dynamic Principal Component Analysis, care are abilitatea de a identifica anomaliile care apar în senzorul de oxigen dizolvat al sistemului BSM2. În [22], autorii creează o rețea neuronală de tip Radial Basis Function pentru a calcula limitele erorilor și pentru a identifica problemele care apar în senzorul de oxigen dizolvat.

Există la fel de multe articole care se ocupă de problema detectării defectelor în stațiile de epurare a apelor uzate. Foarte importantă este teza de doctorat [23], care utilizează rețele neuronale pentru detectarea anomaliilor din stații de epurare. Algoritmii de detecție sunt aplicații pe scenarii de defecte care conțin anomaliile complete și incipiente. Unele articole, cum ar fi [24], [25], [26], propun utilizarea metodelor de tip Autoencoder pentru a detecta defecte, cum ar fi schimbările bruște ale valorilor senzorilor. În [27], autorii folosesc o metoda de detecție tip variational autoencoder (VAE), luând în calcul modificările dinamice care apar în procesul de tratare a apei uzate, ca de exemplu defectul de creștere a nămolului și defectul de treaptă variabilă de mărime mică. Metode autoencoder de tip Convolutional (Conv) și Long Short-Term Memory (LSTM) sunt de asemenea folosite pentru a detecta defecte de tip drift, bias, precision degradation, spike și stuck, care afectează senzorul de oxigen dizolvat al stațiilor de epurare. Schimbând ordinea, severitatea și durata defectelor, au fost create astfel trei scenarii diferite. Măsurătorile arată că metoda Conv-AE are performanță mai bună decât LSTM-AE.

Capitolul 2. Modelarea proceselor complexe de tratare a apelor uzate

Pachetul de Simulink din cadrul programului MATLAB conține mai multe diagrame ale diverselor sisteme de tratare a apelor uzate. În acest capitol vom prezenta trei astfel de sisteme, și anume un sistem simplu cu patru stări ale unei stații de epurare, sistemul BSM1 și sistemul BSM2.

2.1 Sistemul de tratare a apelor uzate cu nămol activ

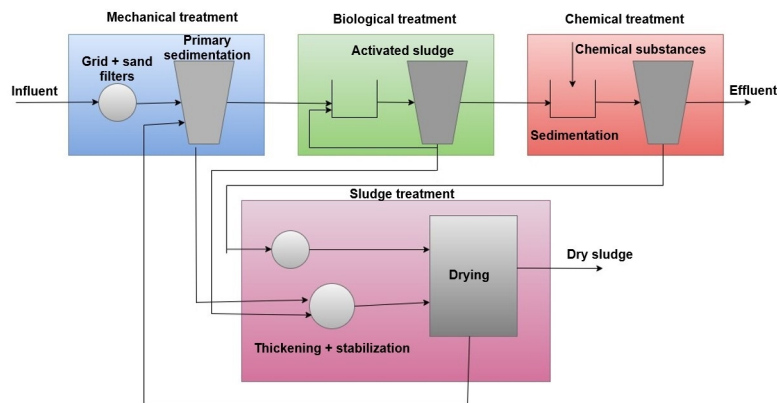


Figura 1: Schema generală a procesului de tratare a apelor uzate cu nămol activ [28]

Sistemul de tratare a apelor uzate cu nămol activ este prezentat într-o manieră simplă de autorii articolului [29], după ce a fost introdus pentru prima dată în [8]. Ecuațiile diferențiale care compun modelul matematic al sistemului cu patru stări este [29]:

$$\begin{aligned}
 \frac{dX}{dt} &= \mu(t)X(t) - D(t)(1+r)X(t) + rD(t)X_r(t) \\
 \frac{dS}{dt} &= -\frac{\mu(t)}{Y}X(t) - D(t)(1+r)S(t) + D(t)S_{in} \\
 \frac{dDO}{dt} &= -\frac{K_0\mu(t)}{Y}X(t) - D(t)(1+r)DO(t) + \alpha W(DO_{max} - DO(t)) + D(t)DO_{in} \\
 \frac{dX_r}{dt} &= D(t)(1+r)X(t) - D(t)(\beta+r)X_r(t), \\
 \text{cu } \mu(t) &= \mu_{max} \frac{S(t)}{K_S + S(t)} \frac{DO(t)}{K_{DO} + DO(t)},
 \end{aligned} \quad (1)$$

unde:

- variabila de intrare este W ;
- variabilele de stare sunt X , S , DO și X_r ;
- variabilele de ieșire (care pot fi măsurate) sunt S și DO .

Considerăm următorii parametri din cadrul modelului descris de ecuațiile (1): $\mu_{max} = 0.15$ mg/l, $K_S = 100$ mg/l, $K_{DO} = 2$ mg/l, $DO_{max} = 10$ mg/l, $D = 0.1\text{h}^{-1}$, $DO_{in} = 0.5$ mg/l, $S_{in} = 200$ mg/l, $Y = 0.65$, $K_0 = 0.5$, $\alpha = 0.018$, $\beta = 0.2$, $r = 0.6$.

De asemenea, valorile pentru variabila de intrare și variabilele de stare sunt: $X(0) = 200$ mg/l, $S(0) = 90$ mg/l, $DO(0) = 2$ mg/l, $X_r(0) = 320$, $W(0) = 30\text{m}^3/\text{h}$.

2.2 Benchmark Simulation Model no. 1 (BSM1)

Benchmark Simulation Model No. 1 (BSM1) este un model de simulare foarte des folosit pentru procesul de tratare a apelor uzate cu nămol activ și a fost creat de echipa intitulată International Water Association (IWA) Task Group for Mathematical Modelling of Activated Sludge Systems [30].

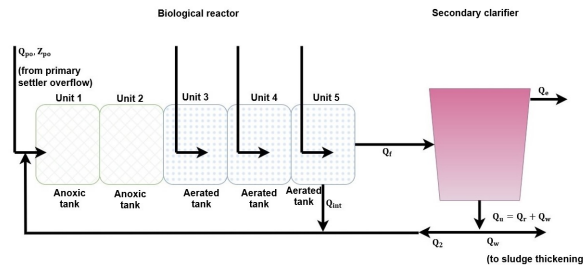


Figura 2: Schema generală a sistemului BSM1 [30]

Comportamentul elementelor importante și ale proceselor în acest sistem, cum ar fi degradarea materiei organice, îndepărtarea nutrienților (nitrogen și fosfor) și dezvoltarea biomasei, precum și ecuațiile diferențiale ale modelului se pot găsi în [30].

2.3 Benchmark Simulation Model no. 2 (BSM2)

Benchmark Simulation Model No. 2 (BSM2) reprezintă un model detaliat folosit pentru implementarea, analiza și evaluarea impactului și performanțelor atât pentru strategiile de control existente, cât și pentru cele nou apărute, aplicate pe stațiile de epurare [31]. Acest model folosește sistemul BSM1 pentru tratarea biologică a apei uzate și conține de asemenea și procesul de tratare cu nămol activ.

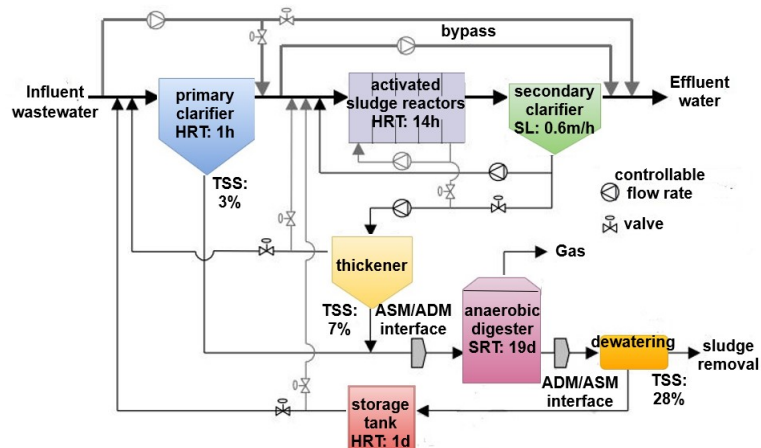


Figura 3: Schema generală a sistemului BSM2 [1]

Notațiile pentru variabilele și parametrii utilizați în sistemul BSM2, precum și ecuațiile modelului matematic, procedura de testare a implementării și criteriile de performanță pot fi găsite în [1].

Capitolul 3. Control optimal

3.1 Generalități

Domeniul vast al controlului optimal conține diverse ramuri, fiecare adresându-se unui aspect diferit al teoriei controlului și al optimizării sistemelor. Controlul optimal liniar se ocupă de sisteme reprezentate de ecuații diferențiale liniare, și are ca scop găsirea strategiilor de control care optimizează un obiectiv definit, în același timp ținând cont de constrângeri. Controlul optimal neliniar se extinde la sisteme guvernate de dinamici neliniare.

3.2 Control optimal neliniar

Tehnica Model Predictive Control neliniară (NMPC) este o strategie avansată de control utilizată în inginerie pentru a optimiza un sistem dinamic neliniar. Deoarece NMPC tratează sisteme cu dinamici neliniare, tehnica poate fi utilizată asupra unei varietăți de probleme [32].

Problema generală de NMPC poate fi formulată astfel:

$$\begin{aligned} & \int_0^T L(x, u) + V(x(T)) \\ \text{s.t. } & \dot{x} = F_c(x, u), x(0) = x_0 \text{ (given)} \\ & x \in \mathbb{X} \\ & u \in \mathbb{U} \end{aligned} \quad (2)$$

cu orizont de predicție de lungime T , și unde $F_c(x, u)$ descrie dinamicile continue ale sistemului. Aici, termenul $L(x, u)$ denotă costul intermediar, iar $V(x(T))$ este costul final, care sunt de obicei scrise în forma [33]:

$$\begin{aligned} L(x, u) &= x^T Q_x x + u^T R_u u, \\ V(x(T)) &= x^T P x, \end{aligned} \quad (3)$$

unde Q_x , R_u și P sunt matrice pozitiv definite.

3.2.1 ACADO Toolkit

ACADO Toolkit (Automatically Coded Algorithm for Dynamic Optimization) este un mediu software utilizat pentru abordarea problemelor de control optimal și optimizare [34].

Interfața MATLAB pentru ACADO Toolkit include integratori ACADO și algoritmi de MPC, control optimal direct și estimarea parametrilor [35].

Putem formula o problemă de MPC folosind mediul de simulare ACADO, proces ce se realizează în patru pași [35]:

- **Pasul 1** - se specifică obiectivele și constrângerile problemei de control optimal.
- **Pasul 2** - se pregătește procesul simulat.
- **Pasul 3** - se stabilește controllerul MPC.
- **Pasul 4** - procesul și controllerul trebuie legate printr-un obiect `acado.SimulationEnvironment`.

3.2.2 Studiu de caz - ACADO Toolkit

Rezultatele discutate în această secțiune au fost introduse prima dată prin lucrarea [36], prezentată la 2023 IEEE 28th International Conference on Emerging Technologies and Factory Automation (ETFA).

ACADO practic aplică practic control optimal pe un sistem neliniar. Pentru aceasta, considerăm costul cu formele (2) și (3) discutate în Secțiunea 3.2, unde: x conține stările, $x = [X \ S \ DO \ X_r]^T$, u reprezintă variabila de intrare, $u = W$ și pentru variabila de ieșire, pe care o putem nota y , este considerat substratul S .

Scopul este obținerea valorilor de până la 40 mg/l pentru substratul S , menținând în același timp valorile pentru oxigenul dizolvat în intervalul 1–3 mg/l. Mai mult, variabila de intrare W trebuie menținută sub $100m^3/h$.

Astfel, se consideră următoarele constrângeri: $0 \leq u \leq 100$ constrângerea pe variabila de intrare, $0 \leq y \leq 40$ constrângerea pe variabila de ieșire, $1 \leq x(3) \leq 3$ constrângerea pe variabila de stare și costul $\int_0^T S^2 + W^2$ [36].

În ACADO, se specifică ecuațiile diferențiale care compun modelul matematic al sistemului și de asemenea valorile inițiale pentru variabilele de stare și intrare (așa cum sunt specificate în Secțiunea 2.1). Mai mult, matricea Q care conține Q_x și R_u pentru cost este creată, unde

$$Q_x = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad R_u = 1 \text{ și } P = 0 \text{ [36].}$$

Valoarea pasului de eșantionare este de asemenea $\Delta T = 0.5$.

Tot ce rămâne de făcut este alegerea punctelor de echilibru în care sistemul se va stabiliza. Sunt alese două seturi de puncte de echilibru, așa cum se poate observa mai jos, iar reprezentările grafice pentru ambele scenarii sunt:

Cazul 1. $rr = [223.8 \ 38.9 \ 2.6 \ 447.6 \ 33]$.

Cazul 2. $rr = [228.7 \ 37.1 \ 2.9 \ 457.4 \ 35]$.

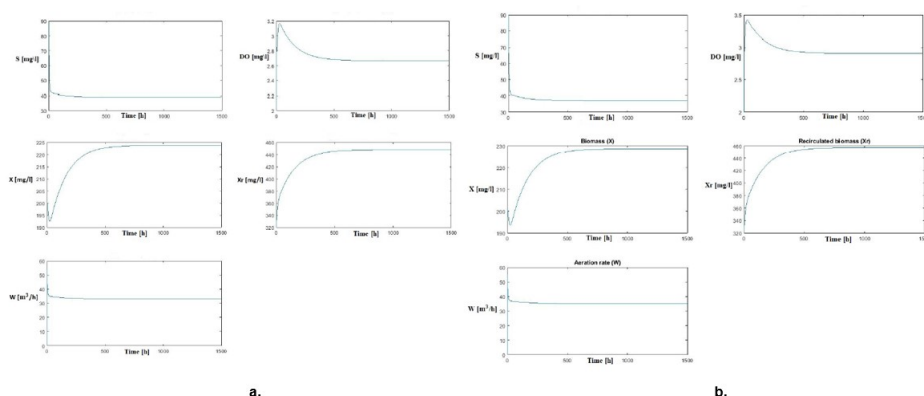


Figura 4: Simulări ACADO pentru: a. Cazul 1, b. Cazul 2 [36]

Se observă că atât variabilele de stare, cât și variabila de intrare ating valorile cerute, indiferent de punctele de echilibru setate.

3.2.3 Discretizare

Se consideră forma generală a unui sistem neliniar descris de o ecuație diferențială [33]:

$$\dot{x} = F_c(x, u), \quad (4)$$

având constrângerile pe stări și intrări $x \in \mathbb{X}$ și $u \in \mathbb{U}$, unde $F_c(x, u)$ descrie dinamicile continue.

Discretizarea sistemului se obține cu metoda Euler [37], și se obține că:

$$F_d(x, u) = x + \Delta T F_c(x, u), \quad (5)$$

unde $F_d(x, u)$ descrie dinamicile discrete.

3.2.4 Problema NLP

Pentru a identifica cele mai bune intrări ale sistemelor dinamice, în teoria controlului optimal neliniar se prezintă frecvent problema în formă de programare neliniară [38].

Forma generală a problemei NLP este [39]:

$$\begin{aligned} \text{(NLP):} \quad & \min_{x \in \mathbb{R}^n} f(x) & (6) \\ & \text{s.t.: } c(x) \leq 0 \\ & \quad c_{eq}(x) = 0, \end{aligned}$$

unde funcțiile $f : \mathbb{R}^n \rightarrow \bar{\mathbb{R}}$, $c : \mathbb{R}^n \rightarrow \mathbb{R}^m$ and $c_{eq} : \mathbb{R}^n \rightarrow \mathbb{R}^p$ se presupun continue și cel puțin o dată diferențiabile.

3.2.5 Studiu de caz - NLP

În acest studiu de caz, obiectivul este stabilizarea sistemului de tratare a apelor uzate cu nămol activ, sub forma unei probleme de programare neliniară, printr-un algoritm în MATLAB care utilizează funcția *fmincon*. Scopul principal rămâne obținerea valorilor de până la 40 mg/l pentru subtratul S , fiind important să menținem calitatea apei în parametrii dați de standardele naționale. Mai mult, trebuie să menținem valorile oxigenului dizolvat în intervalul 1–3 mg/l, care este un factor important în susținerea vieții acvatice și a echilibrului ecologic. În același timp, trebuie monitorizată și controlată variabila de intrare, reprezentată de W , pentru a-o asigura valoarea sub $100 \text{ m}^3/h$.

Problema NLP are următoarele componente:

- variabila de decizie x este reprezentată de variabila de intrare W ;
- funcția obiectiv este dată de costul considerat, care este $\int_0^T S^2 + W^2$;
- constrângerile din Secțiunea 3.2.2.

Observăm că, în toate trei cazurile, sistemul este stabilizat, iar obiectivul de a menține valoarea pentru S sub 40 mg/l, iar pentru DO în intervalul 1 – 3 mg/l a fost atins.

Se observă că în timp ce valoarea pentru ΔT scade, valoarea pentru timpul de rulare crește. Pentru $\Delta T = 0.1$, însă, valoarea care se obține pentru concentrația de DO este cea

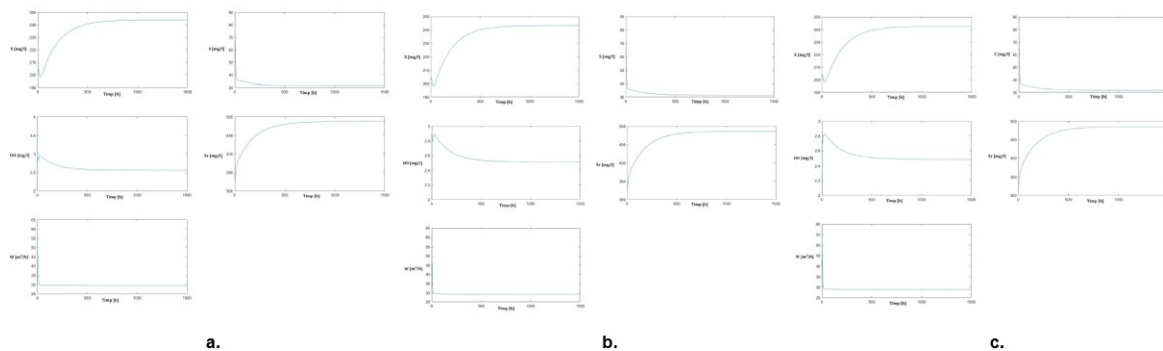


Figura 5: Simulări pentru problema NLP pentru: a. Cazul 1, b. Cazul 2, c. Cazul 3

Tabelul 1: Timpul de rulare pentru cele trei cazuri considerate

Cazuri	Timp de rulare (secunde)
Cazul 1. $\Delta T = 0.5$	450.05
Cazul 2. $\Delta T = 0.25$	859.75
Cazul 3. $\Delta T = 0.1$	2302.35

mai mică, așa cum se observă în reprezentarea grafică, și astfel, pentru cazul al treilea, costul operațional este cel mai mic.

3.3 Control optimal liniar

O altă abordare a problemei propuse presupune transformarea sistemului continuu într-o formă liniară. Această metodă începe cu discretizarea dinamicilor continue și apoi liniarizarea lor în jurul unui punct.

3.3.1 Liniarizare

Pentru a simplifica algoritmi, sistemul este aproximat cu principiul seriilor Taylor. Fie (x_0, u_0) punct de echilibru, astfel $F_d(x_0, u_0) = 0$. Apoi, se aplică seriile Taylor pe funcția F_d în jurul punctului de echilibru și se obține [41]:

$$\begin{aligned}
 F_d(x_k, u_k) &\approx F_d(x_0, u_0) + \frac{\partial F_d}{\partial(x_k, u_k)}(x_0, u_0) \begin{bmatrix} x_k - x_0 \\ u_k - u_0 \end{bmatrix} \\
 &= F_d(x_0, u_0) + \begin{bmatrix} \frac{\partial F_d}{\partial x_k}(x_0, u_0) & \frac{\partial F_d}{\partial u_k}(x_0, u_0) \end{bmatrix} \begin{bmatrix} x_k - x_0 \\ u_k - u_0 \end{bmatrix} \\
 &= F_d(x_0, u_0) + \frac{\partial F_d}{\partial x_k}(x_0, u_0)(x_k - x_0) + \frac{\partial F_d}{\partial u_k}(x_0, u_0)(u_k - u_0).
 \end{aligned}$$

Pentru simplitate, se notează $A_x = \frac{\partial F_d}{\partial x_k}(x_0, u_0)$ și $B_u = \frac{\partial F_d}{\partial u_k}(x_0, u_0)$. Deoarece $F_d(x, u) = x + \Delta T F_c(x, u)$, matricile pot fi scrise și în forma $A_x = \frac{\partial(x + \Delta T F_c)}{\partial x} = I_n + \Delta T \frac{\partial F_c}{\partial x}$ and $B_u = \frac{\partial(x + \Delta T F_c)}{\partial u} = \Delta T \frac{\partial F_c}{\partial u}$.

Având în vedere noțiunile discutate mai sus, se consideră astfel următoarea structura a mod-

elului liniarizat [42]:

$$x_{k+1} = A_x x_k + B_u u_k, \text{ with } x_k \in \mathbb{Z}, u_k \in \mathbb{V}, \quad (7)$$

unde x_k este vectorul de stări și u_k reprezintă vectorul de intrări.

3.3.2 Forma QP

Conform [39], problema poate fi rescrisă în formă de programare pătratică (QP) astfel:

$$\min_{x \in \mathbb{R}^{N(n_x+n_u)}} \frac{1}{2} x^T \bar{Q} x + \bar{q}^T x \quad (8)$$

$$\text{s.t. } Ax = b, \quad Cx \leq d,$$

$$\text{unde } \omega = [v_0^T \ z_1^T \ v_1^T \ z_2^T \ \dots \ v_{N-1}^T \ z_N^T], \quad Q = \begin{bmatrix} R_v & 0 & 0 & \dots & 0 & 0 \\ 0 & Q_z & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & R_v & 0 \\ 0 & 0 & 0 & \dots & 0 & Q_z \end{bmatrix} \text{ și } q = -Q\omega^{ref}.$$

3.3.3 Case study - Linearized model

Rezultatele discutate în această secțiune au fost introduse prima dată prin lucrarea [36], prezentată la 2023 IEEE 28th International Conference on Emerging Technologies and Factory Automation (ETFA).

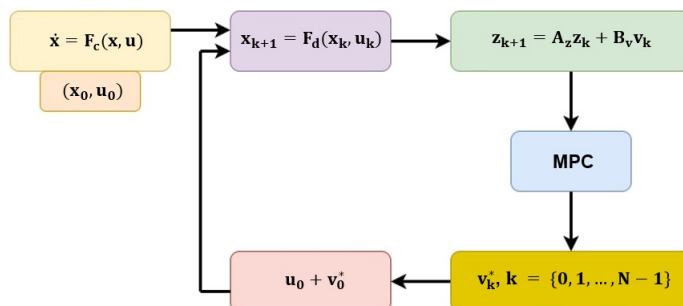


Figura 6: Schema de discretizare și liniarizare [36]

În acest studiu de caz, se construiește o funcție de MATLAB care calculează matricele A_z și B_v , și o funcție de MATLAB pentru forma QP rară, din care rezultă elementele Q , q , A , b , C , d care fac parte din structura formei QP (8).

Algoritmul din MATLAB urmărește schema din Figura 6.

În fișierul script care apelează aceste două funcții, inițializăm stările și intrările $z_0 = [200; 90; 2; 320]$ și $v_0 = 30$. Apoi, valoarea pentru N este $N = 200$, matricele Q_z și R_v care compun Q sunt

$$Q_z = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \text{ și } R_v = 1. \text{ Mai departe, } v_k^*, k = \{0, 1, \dots, N-1\} \text{ este calculat folosind fie co-}$$

manda *quadprog* [43], sau *software-ul de MATLAB CVX* [44]. Valoarea pentru ΔT este $\Delta T = 0.5$ și se rulează simulările pentru cele două cazuri de puncte de echilibru date în Figura 7.

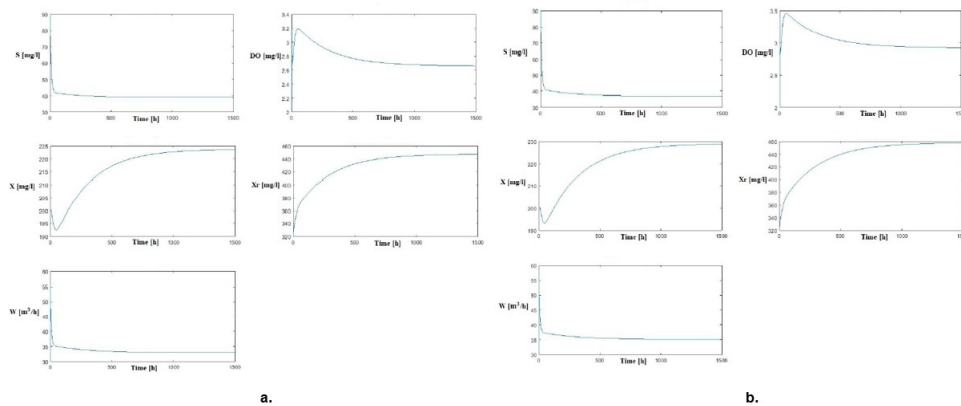


Figura 7: Simulări cu modelul liniarizat pentru: a. Cazul 1, b. Cazul 2 [36]

Este evident din observații că, indiferent de punctele de echilibru setate, variabilele de stare și intrare ating mereu punctele cerute.

3.3.4 Comparație - ACADO vs. modelul liniarizat

Reprezentările grafice următoare afișează comparația între rezolvarea problemei cu ACADO Toolkit și cu algoritmul de MATLAB pentru modelul liniarizat:

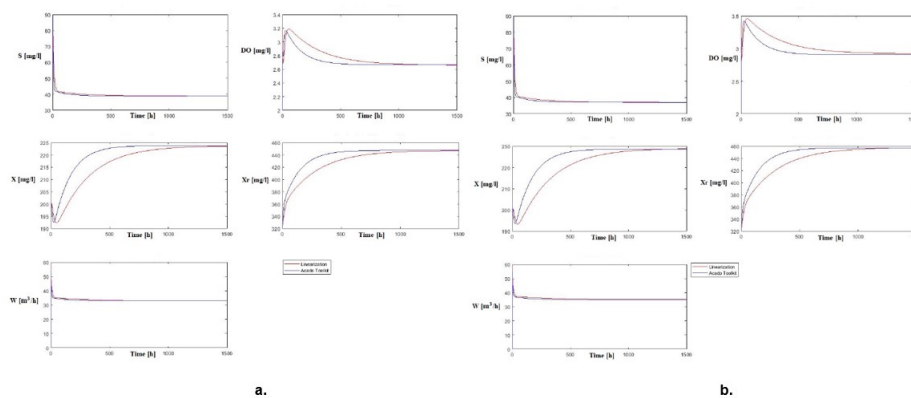


Figura 8: Comparație grafică pentru: a. Cazul 1, b. Cazul 2 [36]

În ceea ce privește timpul de rulare, algoritmul construit în MATLAB este mult mai rapid decât cel creat cu ACADO Toolkit (200 secunde vs. 2 zile complete). În ceea ce privește stabilizarea, este evident din grafice că algoritmul în ACADO stabilizează sistemul mai repede decât celălalt.

3.4 Concluzii

În concluzie, atât ACADO Toolkit, cât și metodele de rezolvare puse la dispoziție de MATLAB, oferă o platformă puternică pentru rezolvarea problemelor de control optimal. În lumea avansată din punct de vedere tehnologic de astăzi, controlul optimal este o disciplină importantă, iar ideile și metodele sale vor continua să fie utilizate în domeniul sistemelor de automatizare și control.

Capitolul 4. Algoritmi de optimizare pentru MPC liniar

4.1 Generalități

În acest capitol, începem cu explorarea conceptelor fundamentale de optimizare, concentrându-ne atenția pe metoda Coordinate Descent, o tehnică versatilă și des utilizată, care simplifică probleme complexe de optimizare prin împărțirea lor în subprobleme ușor de gestionat. Mai mult, ne ocupăm și de metodele de Augmented Lagrangian, o clasă de algoritmi cunoscuți pentru eficiența lor în abordarea problemelor de optimizare constrânsă. Metodele de Augmented Lagrangian folosesc multiplicatori Lagrange în căutarea soluției optime a problemei, ținând cont de constrângeri, fiind astfel instrumente indispensabile în domeniul Model Predictive Control.

4.2 Notății și definiții

Următoarele definiții și remarci sunt preluate din literatura de Coordinate Descent, ca de exemplu din [45].

Definiția 1 Fie funcția $\psi : \mathbb{R}^n \rightarrow \mathbb{R}$ continuă și diferentiabilă. Atunci, derivata $\nabla\psi$ este Lipschitz continuă dacă există $L^\psi > 0$, pentru care este adevărată următoarea relație:

$$\|\nabla\psi(x) - \nabla\psi(y)\| \leq L\|x - y\| \quad \forall x, y \in \mathbb{R}^n. \quad (9)$$

Definiția 2 O funcție $f : \mathbb{R}^n \rightarrow \mathbb{R}$ are gradient Lipschitz continuu, având constantele Lipschitz $L_i > 0$ dacă este adevărată următoarea inegalitate:

$$\|\nabla_i f(x + U_i h) - \nabla_i f(x)\| \leq L_i \|h\| \quad \forall h \in \mathbb{R}^{n_i} \quad \forall i = 1 : N. \quad (10)$$

Utilizând demonstrația Teoremei 2.1.5 din [46], obținem o consecință a acestei proprietăți:

$$f(x + U_i h) \leq f(x) + \langle \nabla_i f(x), h \rangle + \frac{L_i}{2} \|h\|^2 \quad \forall h. \quad (11)$$

4.3 Metoda Coordinate Descent

Metoda Coordinate Descent este un algoritm de optimizare care minimizează o funcție succesiv de-a lungul coordonatelor pentru a-i găsi punctul de minim. La fiecare iterație, algoritmul determină o coordonată (bloc) printr-o regulă de selecție, apoi minimizează exact sau inexact pe direcțiile coordonatei corespunzătoare, fixându-le pe celelalte. Actualizarea se poate face în manieră aleatoare sau ciclică.

4.4 Metoda Augmented Lagrangian și strategia MPC

Metoda Augmented Lagrangian [47] și strategia Model Predictive Control pot fi folosite împreună în anumite situații pentru a rezolva probleme de optimizare cu constrângeri, în contextul strategiilor de control bazate pe MPC. Încorporând metoda Augmented Lagrangian în MPC, constrângerile pot fi gestionate eficient, iar performanța generală a sistemului de control poate fi semnificativ îmbunătățită. Obiectivul nostru este să construim diverși algoritmi care sunt capabili să rezolve

probleme MPC derivate din sisteme liniare și neliniare, prin aplicarea metodei Augmented Lagrangian.

În cele ce urmează, folosim notațiile de mai jos, începând procesul de cercetare la fel ca în lucrarea [48]. Mai mult, algoritmi prezentați în pseudocod în acest capitol se regăsesc de asemenea și în articolul publicat [48].

- $H \succ 0$ ($H \succeq 0$) exprimă că o matrice pătratică H este pozitiv (semi-pozitiv) definită;
- H' (sau z') este notația pentru transpusa unei matrice H (sau a unui vector z);
- $H_{i,j}$ notează elementul de pe linia i și coloana j într-o matrice H ;
- $H_{i,\cdot}$, $H_{\cdot,j}$ reprezintă vectorul linie i , și respectiv vectorul coloană j într-o matrice H .

Pentru un vector z , $\|z\|_2$ exprimă norma euclidiană a vectorului z , $z_{\neq i}$ este un subvector obținut din z prin eliminarea componentei de pe poziția i , și anume z_i .

Începem astfel cu formularea MPC pentru urmărirea referințelor de mai jos [48]:

$$\begin{aligned} \min_{z_t, v_t} & \frac{1}{2} \sum_{t=1}^{N-1} \|y_{t+1} - y_{t+1}^{ref}\|_{Q_y}^2 + \frac{1}{2} \|v_t - v_t^{ref}\|_{R_v}^2 + \frac{1}{2} \|\Delta v_t\|_{W_u}^2 \\ \text{s.t.} & \quad z_{t+1} = A_z z_t + B_v v_t, \quad \forall t = \overline{0, N-1} \\ & \quad y_{t+1} = C_y z_{t+1}, \quad \forall t = \overline{0, N-1} \\ & \quad v_t = v_{t-1} + \Delta v_t, \quad \forall t = \overline{0, N-1} \\ & \quad lb_z \leq z_t \leq ub_z, \quad \forall t = \overline{1, N} \\ & \quad lb_v \leq v_t \leq ub_v, \quad \forall t = \overline{0, N-1} \\ & \quad z_0 = z, v_{-1} = v, \end{aligned} \quad (12)$$

unde $z_t \in \mathbb{R}^{n_z}$ este vectorul stărilor, $v_t \in \mathbb{R}^{n_v}$ este vectorul intrărilor, $y_t \in \mathbb{R}^{n_y}$ este vectorul ieșirilor, $\Delta v_t = v_t - v_{t-1}$ reprezintă vectorul incrementărilor de intrare, y_{t+1}^{ref} și v_t^{ref} sunt punctele de referință pentru stări și intrări, și nu în ultimul rând, z și v sunt valorile inițiale pentru stări și intrări. Pentru a simplifica notațiile, în cele ce urmează considerăm reformularea sistemului (12):

$$\begin{aligned} \min_{s_t} & \frac{1}{2} \sum_{t=1}^N s_t^T (C^T W C) s_t - s_t^T (C^T W r_k) + \frac{1}{2} \Delta v_{t-1}^T W_u \Delta v_{t-1} \\ \text{s.t.} & \quad s_{t+1} = A s_t + B u_t, \quad \forall t = \overline{0, N-1} \\ & \quad lb_s \leq s_t \leq ub_s, \quad \forall t = \overline{1, N} \\ & \quad lb_u \leq \Delta v_t \leq ub_u, \quad \forall t = \overline{0, N-1} \\ & \quad s_0 = \begin{bmatrix} z_0 \\ v_{-1} \end{bmatrix}, \end{aligned} \quad (13)$$

unde $s_t = \begin{bmatrix} z_t \\ v_{t-1} \end{bmatrix}$, $r_t = \begin{bmatrix} y_t^r \\ v_{t-1}^r \end{bmatrix}$, $A = \begin{bmatrix} A_z & B_v \\ 0 & I_{n_v} \end{bmatrix}$, $B = \begin{bmatrix} B_v \\ I_{n_v} \end{bmatrix}$, $C = \begin{bmatrix} C_y & 0 \\ 0 & I_{n_v} \end{bmatrix}$, $W = \begin{bmatrix} Q_y & 0 \\ 0 & R_v \end{bmatrix}$, limita inferioară $lb_s = \begin{bmatrix} lb_z \\ lb_v \end{bmatrix}$ și limita superioară $ub_s = \begin{bmatrix} ub_z \\ ub_v \end{bmatrix}$. Mai mult, notăm prin $\omega \in \mathbb{R}^{N \times n_x}$,

cu $n_x = n_z + 2n_v$ variabila de decizie care cuprinde stările, intrările și vectorul de incremente de intrare asupra întregului orizon de predicție N . Astfel, ω ia următoarea formă:

$$\omega = [\Delta v_0^T, s_1^T, \Delta v_1^T, s_1^T, \dots, \Delta v_{N-1}^T, s_N^T] \in \mathbb{R}^{N \times n_x}. \quad (14)$$

În aceeași manieră, constrângerile de inegalitate pe stări și intrări devin:

$$\begin{cases} lb_s \leq s_t \leq ub_s, & \forall t = \overline{1, N} \\ lb_u \leq \Delta v_t \leq ub_u, & \forall t = \overline{0, N-1}. \end{cases} \iff lb \leq \omega \leq ub. \quad (15)$$

La fiecare pas, problema MPC (12) poate fi rescrisă în forma QP dată mai jos:

$$\begin{aligned} \min_x & \frac{1}{2} \omega^T H \omega + f^T \omega \\ \text{s.t.} & \quad lb \leq \omega \leq ub \\ & \quad A_{eq} \omega = b_{eq} \end{aligned} \quad (16)$$

unde $H = H^T \succeq 0$, $H \in \mathbb{R}^{n \times n}$, $n = Nn_x$, $f \in \mathbb{R}^n$, $A_{eq} \in \mathbb{R}^{N(n_z+n_v) \times n}$ și $b_{eq} \in \mathbb{R}^{N(n_z+n_v)}$,

iar matricele sunt definite ca $H = \begin{bmatrix} R & 0 & 0 & \dots & 0 & 0 \\ 0 & Q & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & R & 0 \\ 0 & 0 & 0 & \dots & 0 & Q \end{bmatrix}$ with $R = W_u$ and $Q = C^T W C$, $f =$

$$\begin{bmatrix} -C^T W r_1 \\ -C^T W r_2 \\ \vdots \\ -C^T W r_N \end{bmatrix}, A_{eq} = \begin{bmatrix} B & -I & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & A & B & -I & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & A & B & -I \end{bmatrix}, b_{eq} = \begin{bmatrix} -A s_0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

Remarca 2 Atât matricele, cât și limitele inferioară și superioară se pot schimba la fiecare execuție.

Rezolvăm problema de programare pătratică (16) cu o metodă bazată pe tehnica Augmented Lagrangian (AL) și strategia Coordinate Descent (CD). Metodele primale nu pot fi aplicate când limitele problemei (16) sunt luate în considerare, deoarece proiectarea pe un set fezabil este imposibilă. De aceea, ne gândim să abordăm strategii duale, deoarece metoda duală rapidă de Augmented Lagrangian este una dintre cele mai eficiente. În consecință, punem la dispoziție expresia de mai jos pentru funcția constrânsă lagrangiană $\mathcal{L} : \mathcal{X} \times \mathbb{R}^n \rightarrow \mathbb{R}$, with $\mathcal{X} = \{lb \leq \omega \leq ub\}$:

$$\mathcal{L}(\omega, \lambda) = \frac{1}{2} \omega^T H \omega + f^T \omega + \lambda^T (A_{eq} \omega - b_{eq}), \quad (17)$$

unde $\lambda \in \mathbb{R}^{N(n_z+n_v)}$ este vectorul de multiplicatori Lagrange asociat constrângerilor de egalitate din problema QP (16). Problema duală a (16) este $\max_{\lambda} d(\lambda) = \max_{\lambda} \min_{\omega \in \mathcal{X}} \mathcal{L}(\omega, \lambda)$.

Conform [49], $d(\lambda)$ nu este diferențiabilă în general, deci orice metoda de subgradient utilizată pentru a rezolva problema duală poate avea o rată de convergență slabă. Astfel, funcția AL:

$$\mathcal{L}_\rho(\omega, \lambda) = \frac{1}{2}\omega^T H\omega + f^T\omega + \lambda^T(A_{eq}\omega - b_{eq}) + \frac{\rho}{2}\|A_{eq}\omega - b_{eq}\|^2 \quad (18)$$

este în schimb utilizată, unde $\rho > 0$ este un parametru de penalizare. Următoarea expresie reprezintă problema duală augmentată: $\max_\lambda d_\rho(\lambda) = \max_\lambda \min_{\omega \in \mathcal{X}} \mathcal{L}_\rho(\omega, \lambda)$ este diferentiabilă dacă $H + \rho A_{eq}^T A_{eq} \succ 0$.

Problema duală și problema duală augmentată au aceeași soluție [50], și cel mai important, $d_\rho(\lambda)$ este concavă și are gradient Lipschitz continuu, având constanta Lipschitz $\frac{1}{\rho}$ [49], [51], $\nabla d_\rho(\lambda) = A_{eq}\omega^*(\lambda) - b_{eq}$, unde $\omega^*(\lambda)$ este soluția optimală a problemei interioare $\min_{\omega \in \mathcal{X}} \mathcal{L}_\rho(\omega, \lambda)$ pentru λ dat.

Problema interioară poate fi scrisă în forma de mai jos:

$$\mathcal{L}_\rho(\omega, \lambda) = \frac{1}{2}\omega^T \mathcal{H}\omega + \ell(\lambda)^T\omega, \quad (19)$$

unde $\mathcal{H} = H + \rho A_{eq}^T A_{eq}$ and $\ell(\lambda) = f + A_{eq}^T \lambda - \rho A_{eq}^T b_{eq}$. Mai mult, \mathcal{H} are o structură bloc rară, ceea ce, conform [52], are forma următoare:

$$\mathcal{H} = \begin{bmatrix} \psi_1 & \psi_2 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ \psi_2^T & \psi_3 & \psi_4 & \psi_5 & 0 & \dots & 0 & 0 & 0 \\ 0 & \psi_4^T & \psi_1 & \psi_2 & 0 & \dots & 0 & 0 & 0 \\ 0 & \psi_5^T & \psi_2^T & \psi_3 & \psi_4 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \dots & \psi_3 & \psi_4 & \psi_5 \\ 0 & 0 & 0 & 0 & 0 & \dots & \psi_4^T & \psi_1 & \psi_2 \\ 0 & 0 & 0 & 0 & 0 & \dots & \psi_5^T & \psi_2^T & \psi_6 \end{bmatrix} \quad (20)$$

unde $\psi_1 = R + \rho B^T B$, $\psi_2 = -\rho B^T$, $\psi_3 = Q + \rho I + \rho A^T A$, $\psi_4 = \rho A^T B$, $\psi_5 = -\rho A^T$, $\psi_6 = Q + \rho I$.

Așa cum sunt prezentate în [53], iterațiile următoare compun algoritmul AL:

$$\omega_{t+1} = \arg \min_{\omega \in \mathcal{X}} \mathcal{L}_\rho(\omega, \lambda_t) \quad (21)$$

$$\lambda_{t+1} = \lambda_t + (A_{eq}\omega_{t+1} - b_{eq}). \quad (22)$$

Convergența algoritmului AL poate fi garantată pentru o varietate largă de valori ale parametru-lui de penalizare, așa cum este prezentat în [53]. Mai specific, cu cât este mai mare valoarea lui ρ , cu atât mai rapid converge algoritmul, însă cu atât mai greu de rezolvat devine subproblema.

Algoritmul accelerat AL are următoarea formă restrânsă:

Metoda Accelerated augmented Lagrangian (AALM) [54]

Intrare: Valoare inițială $\omega_0 \in \mathcal{X}$, $\lambda_0 \in \mathbb{R}^{N(n_z+n_v)}$, $\rho > 0$ parametru de penalizare, T număr maxim de iterații pentru problema exterioară, ϵ_{out} acuratețe exterioară

1. Setare $\tau_0 = 1$, $\nu_0 = \lambda_0$
 2. **for** $t = 0 : T - 1$ **do**
 - 2.1. $\omega_t = \arg \min_{\omega} \mathcal{L}_{\rho}(\omega, \nu_t)$
 - 2.2. $\lambda_{t+1} = \nu_t + \rho(A_{eq}\omega_t - b_{eq})$
 - 2.3. **if** $\|\lambda_{t+1} - \nu_t\|_2^2 \leq \epsilon_{out}$ **break**
 - 2.4. $\tau_{t+1} = \frac{1 + \sqrt{1 + 4\tau_t^2}}{2}$;
 - 2.5. $\nu_{t+1} = \lambda_{t+1} + \frac{\tau_t - 1}{\tau_{t+1}}(\lambda_{t+1} - \lambda_t)$;
 3. **end.**
-

Obiectivul nostru este să rezolvăm subproblema interioară puternic convexă din pasul 2.1 din algoritmul de mai sus. Deoarece funcția d_{ρ} este netedă și concavă, atunci rata de convergență pentru AALM este $O\left(\frac{1}{t^2}\right)$.

4.5 Algoritmi pentru problema interioară

Problema interioară care trebuie rezolvată are forma $\min_{\omega \in \mathcal{X}} \mathcal{L}_{\rho}(\omega, \nu) = \frac{1}{2}\omega^T \mathcal{H} \omega + \ell(\nu)^T \omega$, unde $\mathcal{H} \succ 0$. În consecință, costul este puternic convex, iar valoarea proprie minimă a matricei \mathcal{H} este parametrul de convexitate puternică, $\mu = \lambda_{min}(\mathcal{H})$. Există situații în care parametrul de convexitate puternică este dificil de determinat, și altele în care este ușor de aflat. Există astfel trei soluții: metoda Coordinate Descent, metoda Accelerated Proximal Coordinate Descent și metoda Restarted Accelerated Coordinate Descent. Contribuția noastră principală este ca adaptăm acești trei algoritmi pentru două cazuri: cazul scalar, unde minimizăm problema interioară de-a lungul unei singure coordonate alese aleatoriu la fiecare iterație, menținând celelalte coordonate constante, și cazul bloc, unde actualizăm componente bloc are stărilor și intrărilor sistemului în ordine ciclică inversă. În cazul scalar este nevoie de întreaga matrice \mathcal{H} ca variabilă de intrare. Nu și în cazul bloc, unde se folosește în schimb forma (20), mai specific, $\psi_1, \psi_2, \psi_3, \psi_4, \psi_5$ și ψ_6 . Mai mult, reamintim notația $s_k = \begin{bmatrix} z_k \\ v_{k-1} \end{bmatrix}$, astfel ω_k devine $\omega_k = \begin{bmatrix} \Delta v_{k-1} \\ s_k \end{bmatrix}$, astfel că, pentru cazul bloc, calculăm termenii $s_N, \Delta v_{N-1}, \dots, s_1, \Delta v_0$. De asemenea, constantele Lipschitz sunt calculate după cum urmează:

$$\begin{aligned} L_N^s &= \|\psi_6\| \\ L_{N-1}^s &= L_{N-2}^s = \dots = L_1^s = \|\psi_3\| \\ L_{N-1}^u &= L_{N-2}^u = \dots = L_0^u = \|\psi_1\|. \end{aligned}$$

Motivul pentru care implementăm regula ciclică inversată este deoarece utilizăm faptul că soluția optimală obținută la pasul anterior este folosită ca punct de start în această situație. Următorii algoritmi scriși în pseudocod prezintă cele trei metode (CD, APCG [55] și R-APPROX [56]) pe care le utilizăm pentru a rezolva problema interioară, toți trei fiind prezentați atât pentru cazul scalar, cât și pentru cazul bloc [48].

4.5.1 Metoda CD

CD (Metoda Coordinate Descent pentru cazul scalar)

Intrare: $\omega^0 \in \mathbb{R}^{n_x}$, $\nu \in \mathbb{R}^{N(n_z+n_p)}$, T_{inner} numărul de iterații, ϵ_{in} acuratețe interioară, M numărul de puncte considerate pentru criteriul de oprire.

1. **for** $\ell = 0 : T_{inner} - 1$
 - 1.1 $i = randi(n_x, 1)$
 - 1.2 $\omega_i^{\ell+1} = \left[\omega_i^\ell - \frac{1}{\mathcal{H}_{i,i}} (\mathcal{H}_{i,i} \omega_i^\ell + f_i(\nu)^\ell) \right]_{lb^\ell}^{ub^\ell}$
 - 1.3 $\omega_j^{\ell+1} = \omega_j^\ell, j \neq i$
 2. **if** $\max_{k=0:M-1} (\|\omega^{\ell-k+1} - \omega^{\ell-k}\|^2) > \epsilon_{in}$ **break**
 3. **end.**
-

CD BLOCK (Metoda Coordinate Descent pentru cazul bloc)

Intrare: $\omega^0 \in \mathbb{R}^{n_x}$, $\nu \in \mathbb{R}^{N(n_z+n_p)}$, T_{inner} numărul de iterații, $\psi_1, \psi_2, \psi_3, \psi_4, \psi_5, \psi_6$ (vezi (20)), ϵ_{in} acuratețe interioară, M numărul de puncte considerate pentru criteriul de oprire

1. **for** $\ell = 0 : T_{inner} - 1$
 - 1.1 $s_N^{\ell+1} = \left[s_N^\ell - \frac{1}{L_N^s} (\psi_5^T s_{N-1}^\ell + \psi_2^T \Delta v_{N-1}^\ell + \psi_6 s_N^\ell + f_N(\nu)) \right]_{lb_N}^{ub_N}$
 - 1.2 $\Delta v_{N-1}^{\ell+1} = \left[\Delta v_{N-1}^\ell - \frac{1}{L_{N-1}^v} (\psi_4^T s_{N-1}^\ell + \psi_1 \Delta v_{N-1}^\ell + \psi_2 s_N^\ell + f_{N-1}(\nu)) \right]_{lb_{N-1}}^{ub_{N-1}}$
 - 1.3 **for** $i = N - 1 : 2$ **do**
 - 1.3.1 $s_i^{\ell+1} = \left[s_i^\ell - \frac{1}{L_i^s} (\psi_5^T s_{i-1}^\ell + \psi_2^T \Delta v_{i-1}^\ell + \psi_3 s_i^\ell + \psi_4 \Delta v_i^\ell + \psi_5 s_{i+1}^\ell + f_i(\nu)) \right]_{lb_i}^{ub_i}$
 - 1.3.2 $\Delta v_{i-1}^{\ell+1} = \left[\Delta v_{i-1}^\ell - \frac{1}{L_{i-1}^v} (\psi_4^T s_{i-2}^\ell + \psi_1 \Delta v_{i-2}^\ell + \psi_2 s_{i-1}^\ell + f_{i-1}(\nu)) \right]_{lb_{i-1}}^{ub_{i-1}}$
 - 1.3.3 **end**
 - 1.4 $s_1^{\ell+1} = \left[s_1^\ell - \frac{1}{L_1^s} (\psi_2^T \Delta v_0^\ell + \psi_3 s_1^\ell + \psi_4 \Delta v_1^\ell + \psi_5 s_2^\ell + f_1(\nu)) \right]_{lb_1}^{ub_1}$
 - 1.5 $\Delta v_0^{\ell+1} = \left[\Delta v_0^\ell - \frac{1}{L_0^v} (\psi_1^T \Delta v_0^\ell + \psi_2 s_1^\ell + f_0(\nu)) \right]_{lb_0}^{ub_0}$
 2. **if** $\max_{k=0:M-1} (\|\omega^{\ell-k+1} - \omega^{\ell-k}\|^2) > \epsilon_{in}$ **break**
 3. **end.**
-

4.5.2 Metoda APCG

APCG (Metoda Accelerated Proximal Coordinate Gradient pentru cazul scalar)

Intrare: $\omega^0 \in \mathbb{R}^{n_x}$, $\nu \in \mathbb{R}^{N(n_z+n_p)}$, T_{inner} numărul de iterații, parametrul de convexitate puternică $\mu > 0$, ϵ_{in} acuratețe interioară, M număr puncte pentru criteriul de oprire

1. **set** $\phi^0 = \omega^0$, $w^0 = \omega^0$, $\alpha = \frac{\sqrt{\mu}}{n_x}$
 2. **for** $\ell = 0 : T_{inner} - 1$
 - 2.1 $w^{\ell+1} = \frac{\omega^\ell + \alpha \phi^\ell}{1 + \alpha}$
 - 2.2 $\phi^{\ell+1} = (1 - \alpha) \phi^\ell + \alpha w^{\ell+1}$
 - 2.3 $i = randi(n_x, 1)$
 - 2.3.1 $L_i = |\mathcal{H}_{i,i}|$
 - 2.3.2 $\phi_i^{\ell+1} = \left[\phi_i^\ell - \frac{1}{n_x \alpha L_i} (\mathcal{H}_{i,i} w^{\ell+1} + f_i(\nu)^\ell) \right]_{lb^\ell}^{ub^\ell}$
 - 2.3.3 $\omega^{\ell+1} = w^{\ell+1} + n_x \alpha (\phi^{\ell+1} - \phi^\ell) + n_x \alpha^2 (w^\ell - y^{\ell+1})$
 3. **if** $\max_{k=0:M-1} (\|\omega^{\ell-k+1} - \omega^{\ell-k}\|^2) > \epsilon_{in}$ **break**
 4. **end.**
-

APCG BLOCK (Metoda Accelerated Proximal Coordinate Gradient pentru cazul bloc)

Input: $\omega^0 \in \mathbb{R}^{n_x}$, $\nu \in \mathbb{R}^{N(n_z+n_\nu)}$, T_{inner} number of iterations, $\psi_1, \psi_2, \psi_3, \psi_4, \psi_5, \psi_6$ (see (20)), ϵ_{in} inner accuracy, M number of points to consider for stopping criteria

1. **set** $\phi^0 = \omega^0$, $w^0 = \omega^0$, $\alpha = \frac{\sqrt{\mu}}{n_x}$, $\eta = 2N$
2. **for** $\ell = 0 : T_{inner} - 1$
 - 2.1 $w^{\ell+1} = \frac{\omega^\ell + \alpha \phi^\ell}{1 + \alpha}$
 - 2.2 $\phi^{\ell+1} = (1 - \alpha)\phi^\ell + \alpha w^{\ell+1}$
 - 2.3 $s_N^{\ell+1} = \left[s_N^\ell - \frac{1}{\eta \alpha L_N^s} (\psi_5^T s_{N-1}^\ell + \psi_2^T \Delta v_{N-1}^\ell + \psi_6 s_N^\ell + \mathcal{L}_N(\nu)) \right]_{lb_N}^{ub_N}$
 - 2.4 $\Delta v_{N-1}^{\ell+1} = \left[\Delta v_{N-1}^\ell - \frac{1}{\eta \alpha L_{N-1}^u} (\psi_4^T s_{N-1}^\ell + \psi_1 \Delta v_{N-1}^\ell + \psi_2 s_N^\ell + \mathcal{L}_{N-1}(\nu)) \right]_{lb_{N-1}}^{ub_{N-1}}$
 - 2.5 **for** $i = N - 1 : 2$ **do**
 - 2.5.1 $s_i^{\ell+1} = \left[s_i^\ell - \frac{1}{\eta \alpha L_i^s} (\psi_5^T s_{i-1}^\ell + \psi_2^T \Delta v_{i-1}^\ell + \psi_3 s_i^\ell + \psi_4 \Delta v_i^\ell + \psi_5 s_{i+1}^\ell + \mathcal{L}_i(\nu)) \right]_{lb_i}^{ub_i}$
 - 2.5.2 $\Delta v_{i-1}^{\ell+1} = \left[\Delta v_{i-1}^\ell - \frac{1}{\eta \alpha L_{i-1}^u} (\psi_4^T s_{i-2}^\ell + \psi_1 \Delta v_{i-2}^\ell + \psi_2 s_{i-1}^\ell + \mathcal{L}_{i-1}(\nu)) \right]_{lb_{i-1}}^{ub_{i-1}}$
 - 2.5.3 **end**
 - 2.6 $s_1^{\ell+1} = \left[s_1^\ell - \frac{1}{\eta \alpha L_1^s} (\psi_2^T \Delta v_0^\ell + \psi_3 s_1^\ell + \psi_4 \Delta v_1^\ell + \psi_5 s_2^\ell + \mathcal{L}_1(\nu)) \right]_{lb_1}^{ub_1}$
 - 2.7 $\Delta v_0^{\ell+1} = \left[\Delta v_0^\ell - \frac{1}{\eta \alpha L_0^u} (\psi_1^T \Delta v_0^\ell + \psi_2 s_1^\ell + \mathcal{L}_0(\nu)) \right]_{lb_0}^{ub_0}$
 - 2.8 $\phi^{\ell+1} = \begin{bmatrix} s_0^{\ell+1} \\ \Delta v_0^{\ell+1} \\ s_1^{\ell+1} \\ \vdots \\ \Delta v_{N-1}^{\ell+1} \\ s_N^{\ell+1} \end{bmatrix}$
 - 2.9 $w^{\ell+1} = w^{\ell+1} + n_x \alpha (\phi^{\ell+1} - \phi^\ell) + n_x \alpha^2 (\phi^\ell - w^{\ell+1})$
3. **if** $\max_{k=0:M-1} (\|\omega^{\ell-k+1} - \omega^{\ell-k}\|^2) > \epsilon_{in}$ **break**
4. **end.**

4.5.3 Metoda R-APPROX**APPROX (Metoda Accelerated Parallel and Proximal Coordinate Descent pentru cazul scalar)**

Intrare: $\omega^0 \in \mathbb{R}^{n_x}$, $\nu \in \mathbb{R}^{N(n_z+n_p)}$, T_{inner} număr de iterații, parametrul de convexitate puternică $\mu > 0$ ϵ_{in} acuratețe interioară, M număr puncte criteriu de oprire.

1. **set** $m^0 = \omega^0$, $\theta^0 = \frac{1}{n_x}$
2. **for** $\ell = 0 : T_{inner} - 1$
 - 2.1 $y^{\ell+1} = (1 - \theta^\ell)\omega^\ell + \theta^\ell m^\ell$
 - 2.2 $i = randi(n_x, 1)$
 - 2.2.1 $L_i = |\mathcal{H}_{i,i}|$
 - 2.2.2 $m_i^{\ell+1} = \left[m_i^\ell - \frac{1}{\theta^\ell n_x L_i} (\mathcal{H}_{i,\cdot} y^{\ell+1} + \mathcal{L}_i(\nu)^\ell) \right]_{lb_i}^{ub_i}$
 - 2.2.3 $\omega^{\ell+1} = y^{\ell+1} + n_x \theta^\ell (m_i^{\ell+1} - m_i^\ell)$
 - 2.3 $\theta^{\ell+1} = \frac{\sqrt{(\theta^\ell)^4 + 4(\theta^\ell)^2 - (\theta^\ell)^2}}{2}$
3. **if** $\max_{k=0:M-1} (\|\omega^{\ell-k+1} - \omega^{\ell-k}\|^2) > \epsilon_{in}$ **break**
4. **end.**

APPROX BLOCK (Metoda Accelerated Parallel and Proximal Coordinate Descent pentru cazul bloc)

Intrare: $\omega^0 \in \mathbb{R}^{n_x}$, $\nu \in \mathbb{R}^{N(n_z+n_\nu)}$, T_{inner} număr de iterații, $\psi_1, \psi_2, \psi_3, \psi_4, \psi_5, \psi_6$ (vezi (20)), ϵ_{in} acuratețe interioară, M număr puncte criteriu de oprire

1. **set** $m^0 = \omega^0$, $\theta^0 = \frac{1}{n_x}$, $\eta = 2N$

2. **for** $\ell = 0 : T_{inner} - 1$

2.1 $y^{\ell+1} = (1 - \theta^\ell)\omega^\ell + \theta^\ell m^\ell$

2.2 $s_N^{\ell+1} = \left[s_N^\ell - \frac{1}{\theta^\ell \eta L_N^s} (\psi_5^T s_{N-1}^\ell + \psi_2^T \Delta v_{N-1}^\ell + \psi_6 s_N^\ell + f_N(\nu)) \right]_{lb_N}^{ub_N}$

2.3 $\Delta v_{N-1}^{\ell+1} = \left[\Delta v_{N-1}^\ell - \frac{1}{\theta^\ell \eta L_{N-1}^s} (\psi_4^T s_{N-1}^\ell + \psi_1 \Delta v_{N-1}^\ell + \psi_2 s_N^\ell + f_{N-1}(\nu)) \right]_{lb_{N-1}}^{ub_{N-1}}$

2.4 **for** $i = N - 1 : 2$ **do**

2.4.1 $s_i^{\ell+1} = \left[s_i^\ell - \frac{1}{\theta^\ell \eta L_i^s} (\psi_5^T s_{i-1}^\ell + \psi_2^T \Delta v_{i-1}^\ell + \psi_3 s_i^\ell + \psi_4 \Delta v_i^\ell + \psi_5 s_{i+1}^\ell + f_i(\nu)) \right]_{lb_i}^{ub_i}$

2.4.2 $\Delta v_{i-1}^{\ell+1} = \left[\Delta v_{i-1}^\ell - \frac{1}{\theta^\ell \eta L_{i-1}^s} (\psi_4^T s_{i-2}^\ell + \psi_1 \Delta v_{i-2}^\ell + \psi_2 s_{i-1}^\ell + f_{i-1}(\nu)) \right]_{lb_{i-1}}^{ub_{i-1}}$

2.4.3 **end**

2.5 $s_1^{\ell+1} = \left[s_1^\ell - \frac{1}{\theta^\ell \eta L_1^s} (\psi_2^T \Delta v_0^\ell + \psi_3 s_1^\ell + \psi_4 \Delta v_1^\ell + \psi_5 s_2^\ell + f_1(\nu)) \right]_{lb_1}^{ub_1}$

2.6 $\Delta v_0^{\ell+1} = \left[\Delta v_0^\ell - \frac{1}{\theta^\ell \eta L_0^s} (\psi_1^T \Delta v_0^\ell + \psi_2 s_1^\ell + f_0(\nu)) \right]_{lb_0}^{ub_0}$

2.7 $m^{\ell+1} = \begin{bmatrix} s_0^{\ell+1} \\ \Delta v_0^{\ell+1} \\ s_1^{\ell+1} \\ \vdots \\ \Delta v_{N-1}^{\ell+1} \\ s_N^{\ell+1} \end{bmatrix}$

2.8 $\omega^{\ell+1} = y^{\ell+1} + \theta^\ell \eta (m^{\ell+1} - m^\ell)$

2.9 $\theta^{\ell+1} = \frac{\sqrt{(\theta^\ell)^4 + 4(\theta^\ell)^2} - (\theta^\ell)^2}{2}$

3. **if** $\max_{k=0:M-1} (\|\omega^{\ell-k+1} - \omega^{\ell-k}\|^2) > \epsilon_{in}$ **break**

4. **end.**

Pentru a ne asigura că obținem convergență liniară, aplicăm schema de restart propusă în [56], care aproximează adaptiv parametrul de convexitate puternică.

R - APPROX (Metoda Accelerated Parallel and Proximal Coordinate Descent Method cu restart)

1. **alegere** $\omega_0 \in \mathbb{R}^{n_x}$

2. **setare** $\hat{\omega}_0 = \omega_0$

3. **alegere perioade de restart** $T_{inner}^0, T_{inner}^1, \dots, T_{inner}^r \subseteq \mathbb{N}$

4. **for** $r \geq 0$ **do**

4.1 $\hat{\omega}_{r+1} = \text{APPROX}(\text{problema interioară}, \hat{\omega}_r, T_{inner}^r)$

5. **end.**

4.6 Studiu de caz

Rezultatele prezentate în acest studiu de caz, care au fost obținute din algoritmi prezentati în acest capitol, au fost prima dată introduse în lucrarea [48], publicată în Journal of Control

Engineering and Applied Informatics.

4.6.1 Sisteme liniare și neliniare

Sistemele liniare folosite în cercetarea noastră sunt: modelul aviatic liniarizat și instabil în buclă închisă AFTI F-16 [57], [58], modelul de elicopter de laborator (Quanser 3-DOF Helicopter) [59], sistemul mass-spring [60], pendulum inversat [61] și sistemul cu patru rezervoare [62].

Sistemele neliniare utilizate în cercetarea noastră sunt: sistemul de tratare a apelor uzate cu nămol activ în patru stări descris în Secțiunea 2.1 și sistemul cu rezervor adiabatic cu amestecare continuă (CSTR), explicat în [63] și pus la dispoziție de site-ul Mathworks [64].

4.6.2 Rezultatele simulărilor

Se stabilesc condițiile de simulare. Astfel, ca metrică de performanță folosim un cost pătratic cu matricele Q și R definite în Secțiunea 2.3, $R_v = 0_{n_v}$, $W_u = 0.1I_{n_v}$ și $Q_y = 10I_{n_y}$. Mai mult, pasul de eșantionare este $dN = 0.05$, iar orizontul de predicție este $N = 5$.

Criteriile de convergență sunt [48]:

- pentru problema interioară, algoritmul se oprește când numărul de iterații atinge o anumită valoare: 200000 pentru CD, 50000 pentru APCG, 5000 pentru APPROX, 50000 pentru CD BLOCK, APCG BLOCK și APPROX BLOCK, sau când $\|\omega_i - \omega_{i-1}\|^2 > 10^{-7}$
- pentru problema exterioară, algoritmul se oprește când numărul de iterații atinge valoarea 5000 pentru oricare dintre metode, sau $\|\lambda - \Lambda\|^2 > 10^{-4}$.

Cele două tabele date mai jos prezintă numărul de iterații interioare și exterioare pentru fiecare metodă, atât pentru cazul scalar, cât și pentru cazul bloc, și de asemenea timpul de rulare pentru fiecare pas de MPC, și timpul mediu și maxim pentru fiecare iterație a problemei exterioare:

Algorithms		Systems						
		A-16	H	M-S	IP	FT	WWTP	CSTR
CD	Inner iterations	40115	32495	6365	2095	15620	20780	1085
	Outer iterations	80	25	40	75	15	5000	5000
	MPC step	5.8	1.6	1.5	0.9	0.5	115.4	53.7
	Average time	0.07	0.06	0.03	0.02	0.01	0.02	0.01
	Maximum time	0.1	0.4	0.09	0.1	0.02	0.07	0.04
APCG	Inner iterations	5070	4730	5245	2252	4795	4060	2060
	Outer iterations	80	20	40	40	15	5000	5000
	MPC step	1.04	0.2	0.6	0.3	0.1	38.6	12.1
	Average time	0.01	0.009	0.01	0.009	0.001	0.007	0.002
	Maximum time	0.01	0.1	0.03	0.04	0.002	0.03	0.03
R-APPROX	Inner iterations	2000	2000	2250	2000	2000	2000	2000
	Outer iterations	75	20	40	40	15	5000	5000
	MPC step	2.04	0.3	0.9	0.6	0.2	87.7	34.9
	Average time	0.02	0.01	0.02	0.01	0.02	0.01	0.007
	Maximum time	0.004	0.3	0.003	0.03	0.03	0.04	0.03

a.

Algorithms		Systems						
		A-16	H	M-S	IP	FT	WWTP	CSTR
CD BLOCK	Inner iterations	6113	3705	775	990	320	5020	325
	Outer iterations	75	20	40	40	15	5000	5000
	MPC step	13.01	1.3	3.1	2.3	0.5	528.9	238.1
	Average time	0.2	0.05	0.07	0.05	0.03	0.1	0.04
	Maximum time	0.3	0.1	0.4	0.1	0.1	0.2	0.1
APCG BLOCK	Inner iterations	1175	946	235	170	95	1010	135
	Outer iterations	75	20	40	40	15	5000	5000
	MPC step	3.01	0.4	1.3	0.3	0.2	132.3	28.5
	Average time	0.03	0.01	0.02	0.005	0.008	0.02	0.005
	Maximum time	0.04	0.04	0.2	0.02	0.02	0.06	0.04
R-APPROX BLOCK	Inner iterations	550	550	500	500	500	550	550
	Outer iterations	75	20	40	40	15	5000	5000
	MPC step	9.9	1.01	2.8	1.9	0.2	439.4	52.8
	Average time	0.1	0.004	0.06	0.04	0.01	0.08	0.01
	Maximum time	0.2	0.09	0.09	0.07	0.04	0.1	0.1

b.

Figura 9: Număr de iterații și timp de rulare pentru: a. cazul scalar, b. cazul bloc [48]

Afișăm de asemenea și comparațiile grafice rezultate pentru sistemul AFTI F - 16 și pentru sistemul de tratare a apelor uzate cu nămol activ. Comparăm cei trei algoritmi, atât pentru cazul scalar, cât și pentru cazul bloc.

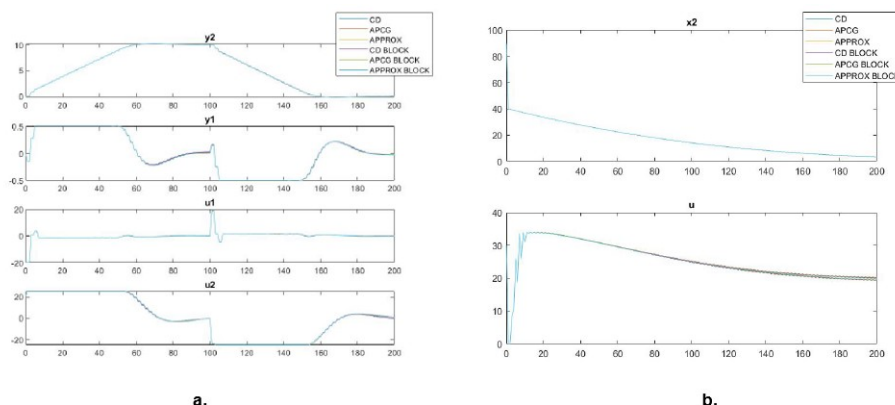


Figura 10: Comparație grafică pentru: a. sistemul AFTI-16, b. Sistemul de tratare a apelor uzate cu nămol activ [48]

Observăm că timpul de rulare pentru toate cazurile luate în considerare este mai redus atunci când problema este rezolvată cu APCG și R – APPROX decât în situația în care este rezolvată cu metoda clasică de Coordinate Descent.

Comparând APCG și R – APPROX, rezultă că numărul de iterații și timpul de rulare sunt mai mici pentru APCG decât pentru R – APPROX. Însă, un argument notabil în favoarea metodei R – APPROX este că, spre deosebire de metoda APCG, aceasta nu are nevoie de valoarea parametrului de convexitate puternică μ , metoda funcționând cu un parametru de convexitate puternică necunoscut.

4.7 Concluzii

În concluzie, metodele Coordinate Descent, Augmented Lagrangian și Model Predictive Control oferă perspective diferite în abordarea problemelor de optimizare. Mai mult, integrarea strategiilor de Augmented Lagrangian și Model Predictive Control în domeniul problemelor de optimizare reprezintă un avans considerabil în cercetare. Prin reducerea decalajului dintre optimizarea constrânsă și controlul predictiv, această metodă combinată aduce mai multe avantaje: extinde numărul și tipul de probleme care pot fi rezolvate și deschide calea pentru soluții inovative în diverse domenii. Alegerea metodei depinde de caracteristicile problemei și de compromisurile dintre eficiența computațională, acuratețe și ușurința de implementare.

Capitolul 5. Metode de Deep Learning pentru recunoașterea și detectarea defectelor

5.1 Generalități

Rețelele neuronale (NNs) sunt esențiale în domeniile de inteligență artificială, Machine Learning și Deep Learning. Principiul fundamental al rețelelor neuronale este de a analiza datele pentru a descoperi tipare și conexiuni [65]. Metricele de performanță sunt definite ca instrumente utilizate pentru evaluarea eficienței și calității predicțiilor unui model asupra unei sarcini date. Mai mult, ele permit actualizări ale modelului iterativ și ajută în luarea de decizii informate, oferind informații despre avantajele și dezavantajele modelului considerat [66], [67]. Metricele de performanță sunt calculate prin utilizarea elementelor matricei de confuzie [68]. În problemele de clasificare binară (care au două clase sau rezultate potențiale), o matrice de confuzie (numită de asemenea și matrice de eroare) este o matrice utilizată în măsurarea performanței unui model de clasificare de machine learning. Matricea de confuzie este formată din patru elemente importante [68]: True Positives (TP) - cazurile pe care modelul le-a recunoscut drept pozitive și sunt într-adevăr pozitive; True Negatives (TN) - cazurile pe care modelul le-a recunoscut drept negative și sunt într-adevăr negative; False Positives (FP) - cazurile eronat etichetate drept pozitive; False Negatives (FN) - cazurile eronat etichetate drept negative.

Metricile de performanță pe care le utilizăm în lucrarea noastră sunt descrise în cele ce urmează [69]:

- **Acuratețe**

Acuratețea reprezintă un indicator de performanță fundamental și des utilizat. Acuratețea este măsurată prin procentul de predicții corecte și este calculată prin ecuația $Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$.

- **Precizie**

Precizia măsoară procentul de predicții pozitive (corect detectate ca pozitive). Astfel, precizia este calculată ca $Precision = \frac{TP}{TP+FP}$.

- **Recall**

Recall este o statistică de performanță esențială care determină procentul de predicții pozitive. Astfel, recall este calculat ca $Recall = \frac{TP}{TP+FN}$.

- **F1-score**

F1-score este un indicator de performanță de machine learning care oferă o evaluare a performanței de clasificare a modelului creând un echilibru între precizie și recall. Astfel, F1-score este calculat ca $F1 - score = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} = \frac{2 \cdot TP}{2 \cdot TP + FP + FN}$.

5.2 Defecte injectate pe senzorul de oxigen dizolvat

Senzorul de oxigen dizolvat (DO) este una dintre cele mai importante componente ale unei stații de tratare a apelor uzate, și este esențial pentru păstrarea eficienței procesului și pentru respectarea cerințelor de mediu. În această secțiune ne vom referi la importanța, utilizările și contribuțiile sustenabile ale senzorului de oxigen dizolvat în contextul tratării apelor uzate.

Există multe defecte mecanice care pot afecta senzorul de oxigen dizolvat dintr-o stație de epurare, printre care: defectul de pompă de aerare, defectul drift, defectul bias, defectul stuck, defectul spike și defectul precision degradation (PD). În subsecțiunile ce urmează vom descrie fiecare dintre aceste defecte, și vom prezenta atât expresia matematică a lor, cât și ceea ce se întâmplă când apar aceste defecte într-o stație de tratare. Menționăm că aceste defecte au fost implementate în sistemul BSM2.

Diversele procese ale unei stații de tratare sunt simulate prin sistemul BSM2, așa cum putem observa mai jos, în schema simplificată a acestuia.

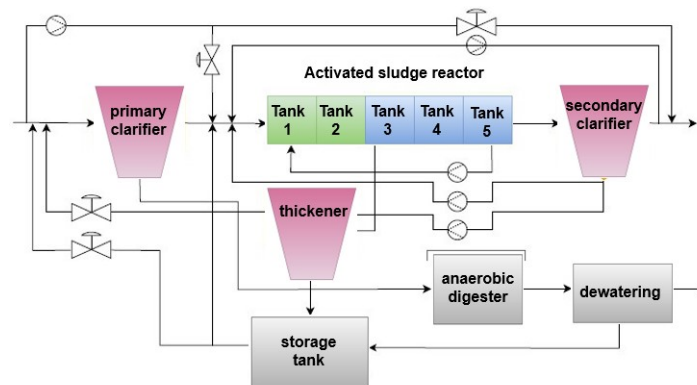


Figura 11: Schemă simplificată a modelului BSM2 [27]

Așa cum putem observa, există cinci rezervoare, primele două sunt anoxice, iar ultimele trei sunt oxigenate. Controlerul proporțional-integral (PI), care primește valorile de oxigen dizolvat măsurate în rezervorul al patrulea, le compară pe acestea cu valorile de referință, și apoi controlează pompele necesare pentru a menține nivelul de oxigen dizolvat la valoarea potrivită, și anume $2[mg/L]$ [27].

5.2.1 Defectul de pompă de aerare

Aerarea reprezintă un pas important în procesul de tratare a apelor uzate, incluzând adăugarea de oxigen în apa uzată pentru a permite proceselor biologice să aibă loc. Defectul de pompă de aerare, așa cum îl numim în lucrarea de față, este o anomalie completă, care afectează coeficientul de transfer de oxigen ($K_L a_4$) din rezervorul al patrulea pe o perioadă specifică de timp.

Defectele din secțiunile următoare pot fi injectate direct în senzorul de oxigen dizolvat, nu doar îl afectează ca o consecință. Considerăm astfel expresia $s(t) = h(t) + \eta$ pentru ieșirea așteptată a senzorului considerat, fără nicio anomalie injectată, unde $h(t)$ este ieșirea senzorului la timpul t și η este zgomotul care apare cu semnalul [70].

5.2.2 Defectul drift

Defectul drift este un tip de anomalie care apare când valoarea senzorului de oxigen dizolvat diverge treptat de la valoarea așteptată. Pe scurt, ieșirea crește liniar de la valoarea normală cu trecerea timpului. Formula matematică a defectului drift este $s(t) = h(t) + \eta + b(t)$, $b(t) = b(t-1) + v$, unde v este constantă, $b(t)$ este părtinirea adăugată la semnal la momentul de timp t , care crește după expresia $b(t) = b(t-1) + v$ [71].

5.2.3 Defectul bias

Atunci când senzorul de oxigen dizolvat întâmpină un defect de tip bias, măsurătorile senzorului deviază de la valoarea reală sau estimată într-o anumită direcție (fie mai sus, fie mai jos). În termeni simpli, când o valoare constantă este adăugată la ieșirea senzorului, apare o deviație față de valoarea esimată. Expresia matematică pentru defectul bias este $s(t) = h(t) + \eta + v$, unde v este constantă [71].

5.2.4 Defectul stuck

Defectul de tip stuck este o anomalie prin care senzorul de oxigen dizolvat devine fixat la o anumită valoare și nu poate livra informații corecte despre concentrația de oxigen dizolvat din apa tratată. Matematic, acest tip de defect poate fi descris ca $s(t) = v$, unde v este constantă [71].

5.2.5 Defectul spike

Defectul de tip spike este definit ca o creștere sau o scădere scurtă și neașteptată a nivelului de oxigen dizolvat, care nu corespund cu nivelul corect de oxigen din apă. Practic, deviații de amplitudine mare de tip spike sunt observate în ieșirea senzorului, și, pentru a obține un astfel de defect, se adaugă o partiție continuă b_t componentelor semnalului normal. Matematic, defectul de tip spike are forma $s(t) = h(t) + \eta + b_t$, $\forall t \in v \times \tau$, unde $v = \{1, 2, \dots\}$ este un set de numere naturale și $\tau \geq 2$ este intervalul în care anomaliile spike apar în ieșirea senzorului [71].

5.2.6 Defectul precision degradation

Defectul de tip precision degradation al senzorului de oxigen dizolvat descrie o circumstanță în care capacitatea senzorului de a produce măsurători credibile ale concentrației de oxigen dizolvat din apă se degradează de-a lungul timpului. Practic, acest tip de defect modifică valoarea ieșirii senzorului cu un zgomot cu medie zero și o variație mare. Defectul precision degradation este definit matematic ca fiind $s(t) = h(t) + \eta + \zeta$, $\delta_\zeta^2 \gg \delta_n^2$, unde δ_ζ^2 este zgomotul adăugat cu medie zero și o variație mare [71].

5.3 Recunoaștere de defecte

Recunoașterea de defecte joacă un rol important în identificarea anomaliilor care apar în sistemele de tratare a apelor uzate și asigură eficiența proceselor de îndepărtare a contaminanților [72]. Algoritmii de învățare supervizată pot fi antrenati să recunoască tipare care sugerează probleme, prin utilizarea informațiilor anterioare și prin eșantioane etichetate pentru stări operaționale corecte și incorecte.

Una dintre strategiile de învățare supervizată care apare ca un instrument deosebit de puternic pentru procesul de recunoaștere al defectelor este rețeaua neuronală tip Feedforward (FFNN) [73]. Cealaltă metodă care se remarcă este rețeaua neuronală de tip 1D Convolutional (1DCNN), o clasă de modele de Deep Learning construite pentru măsurători de senzori, printre altele [74].

5.4 Studiu de caz

Rezultatele expuse în acest studiu de caz, care au fost prima dată introduse în lucrarea [75], au fost de asemenea prezentate la 28th International Conference on Emerging Technologies and Factory Automation (ETFAs).

Pentru cercetarea noastră, am propus și am evaluat doi algoritmi de învățare supervizată (FFNN și 1DCNN) pentru recunoașterea atât a operării normale a senzorului, cât și a patru defecte mecanice (bias, stuck, spike și PD) care afectează senzorul de oxigen dizolvat și buna funcționare a stației de tratare a apelor uzate. Menționăm că am construit modelele neuronale în mediu Google Colaboratory (Colab) folosind librăriile open-source de Python Scikit-Learn 1.2.2 și TensorFlow 2.12 cu Keras, un API Deep Learning de nivel înalt integrat.

De asemenea, datele pentru funcționarea anormală a senzorului de oxigen dizolvat și a pompei de aerare au fost colectate prin injectarea blocurilor de defecte în modelul BSM2, folosind MATLAB Simulink. Modelul stației de tratare a apelor uzate a fost rulat pentru un total de 609 zile, așa cum este prezentat în [1], informațiile fiind analizate o dată la 15 minute, începând cu ziua a 245-a. Următorul tabel conține ziua de început și durata fiecărui defect:

Tabelul 2: Injectarea de defecte pe senzorul de oxigen dizolvat [75]

Defecte	Ziua de start	Durață [ore]
Bias	280	480
Stuck	350	600
Spike	400, 420, 440, 460	48
PD	500	720

Reprezentarea grafică a fiecărui defect este dată mai jos:

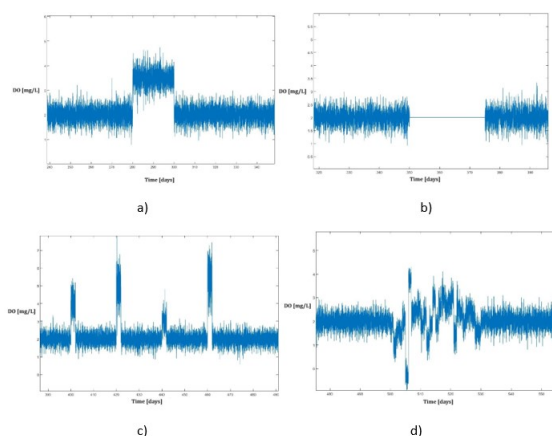


Figura 12: Defectele senzorului de oxigen dizolvat: a. bias, b. stuck, c. spike, d. precision degradation [75]

Scopul studiului nostru este să comparăm două metode de Deep Learning, FFNN și 1DCNN și să stabilim care tehnica de Deep Learning este capabilă să identifice în mod corect cele cinci stări de operare ale senzorului de oxigen dizolvat în cadrul stației de tratare a apelor uzate considerate în această lucrare. Cele cinci stări de operare sunt: starea normală de operare, pe care o notăm ca fiind clasa 0 în setul de date creat pentru simulările noastre, defectul de tip bias injectat în senzorul de oxigen dizolvat, considerat clasa 1, defectul de tip stuck injectat în senzorul de oxigen dizolvat,

considerat clasa 2, defectul de tip spike injectat în senzorul de oxigen dizolvat, considerat clasa 3 și defectul de tip precision degradation (PD) injectat în senzorul de oxigen dizolvat, considerat clasa 4.

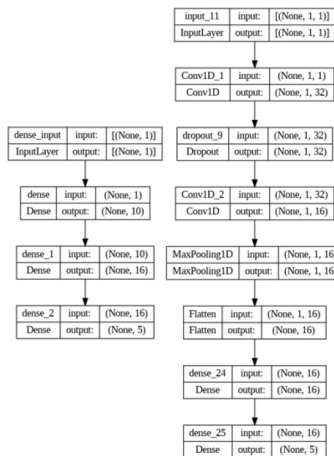


Figura 13: Arhitecturile rețelelor neuronale FFNN și 1SCNN [75]

Observația 1. Ambele rețele FFNN și 1DCNN sunt compilate cu funcția Categorical Cross Entropy Loss function, optimizatorul Adam cu rata de învățare 0.001 și metrică de acuratețe pentru evaluarea modelului în timpul antrenării și validării. Mai mult, antrenarea se efectuează cu metoda fit() din Keras API, iterată în 100 epochs.

Observația 2. 80% din date au fost selectate pentru antrenare și 20% din informațiile din setul de date au fost păstrate pentru testare.

Hiperparametrii celor două rețele neuronale au fost selectați prin Keras-Tuner [76], o librărie generală de tuning pentru hiperparametrii. Tot ce ne rămâne de făcut este să comparăm cele două metode de Deep Learning în ceea ce privește metricile de performanță (acuratețe, precizie, recall și F1-score). Date mai jos sunt matricile de confuzie obținute pentru cele două metode:

	normal	bias	stuck	spike	PD
normal	10020.0	2.0	29.0	0.0	37.0
bias	4.0	394.0	0.0	0.0	0.0
stuck	13.0	0.0	448.0	0.0	0.0
spike	19.0	0.0	0.0	139.0	0.0
PD	95.0	0.0	0.0	0.0	493.0

a.

	normal	bias	stuck	spike	PD
normal	9917.0	0.0	194.0	0.0	7.0
bias	31.0	357.0	0.0	0.0	0.0
stuck	0.0	0.0	473.0	0.0	0.0
spike	20.0	61.0	0.0	65.0	0.0
PD	466.0	50.0	1.0	0.0	51.0

b.

Figura 14: Matricile de confuzie pentru metodele de Deep Learning: a. FFNN și b. 1DCNN [75]

Rețeaua FFNN obține acuratețe de antrenare de 98.32% și acuratețe de testare 98.30%, iar rețeaua 1DCNN obține acuratețe de antrenare de 92.76% și acuratețe de testare 92.90% .

Matricea de confuzie a rețelei FFNN demonstrează ca modelul clasifică corect în proporție de 98.3% (11494 date din totalul de 11693) și incorect în proporție de 1.7% (199 date din totalul de 11693). Matricea de confuzie a rețelei 1DCNN arată că modelul clasifică corect în proporție de 92.90% (10863 date din totalul de 11693) și incorect în proporție de 7.1% (830 date din totalul de

11693). Aceste rezultate demonstrează ca performanța rețelei FFNN este mai bună decât cea a modelului 1DCNN.

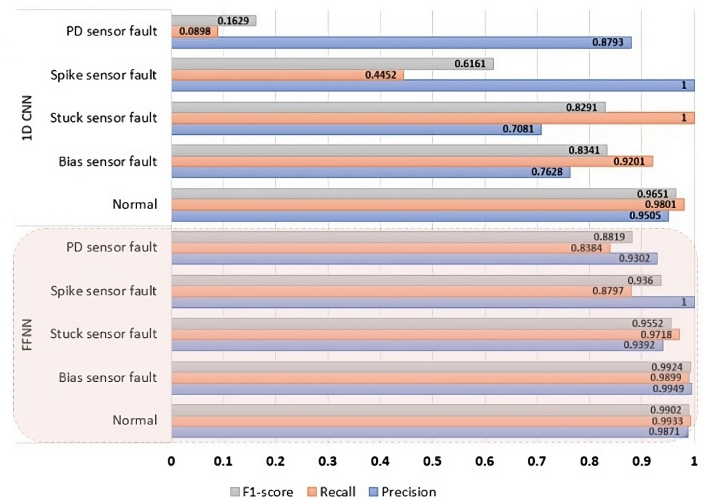


Figura 15: Precizie, Recall și F1-score pentru FFNN și 1DCNN [75]

Conform Figurii 13 de mai sus, cele mai bune rezultate de clasificare se obțin în scenariul cu defect de tip bias pe sensor pe care aplicăm rețeaua FFNN (99.49% precizie, 98.99% recall și 99.24% F1-score) iar cele mai slabe rezultate se obțin în scenariul cu defect de tip PD pe sensor pe care aplicăm rețeaua 1DCNN (87.93% precizie, 0.08% recall și 0.16% F1-score). Concluzia este că datele obținute din acest caz de simulare sunt dezechilibrate prin comparație cu alte clase. De aceea, e posibil să fie necesar să adunăm mai multe date pentru a îmbunătăți performanța modelului 1DCNN.

Studiu arată că rețeaua neuronală FFNN are performanță și adaptabilitate mai bune decât cea a modelului 1DCNN.

5.5 Detectare de defecte

Metodele de învățare semi-supervizate reprezintă o opțiune promițătoare pentru detectarea defectelor într-o stație de tratare a apelor uzate. Învățarea semi-supervizată, spre deosebire de cea supervizată, utilizează o combinație de date etichetate și neetichetate atât pentru condiții normale, cât și pentru cele cu anomalii. Învățarea semi-supervizată este utilă în cazul stațiilor de epurare, unde datele etichetate pot fi rare și greu de obținut. Aceste metode pot detecta neregularități sau erori în timp real sau aproape real folosind o cantitate mică de date etichetate pentru antrenare și o cantitate mare de date neetichetate din operarea continuă a stațiilor de epurare. Modelele semi-supervizate, așa cum sunt metodele tip autoencoder și algoritmi care se auto-antrenează, pot detecta în mod autonom deviații ale comportamentului prezis, contribuind la eficiența operațională a stațiilor de tratare a apelor uzate [77].

5.6 Metode semi-supervizate

5.6.1 Metode clasice semi-supervizate

Algoritmii clasici semi-supervizați sunt o clasă de metode care realizează conexiunea dintre tehnici de Machine Learning complet supervizate și nesupervizate. În cercetarea noastră, con-

siderăm trei metode clasice de învățare semi-supervizată: Isolation Forest (IF), prima dată introdusă în [78], Local Outlier Factor (LOF), prima dată publicată în [79], și One-Class Support Vector Machine (OCSVM), propus pentru prima dată în [80].

După ce am studiat aceste metode, am găsit câteva dezavantaje care au impact asupra eficienței lor, cum ar fi scalabilitate limitată, lipsă de adaptabilitate, sau dificultate în manipularea datelor dezechilibrate, așa că am decis să extindem cercetarea noastră la metode semi-supervizate de Deep Learning, care sunt mai potrivite pentru provocările specifice domeniului de detectării defectelor în stațiile de tratare a apelor uzate. Aceste metode moderne oferă o adaptabilitate mai bună, scalabilitate și robustețe în manipularea datelor complexe și cu zgomot.

5.6.2 Metode de Deep Learning semi-supervizate

Examinăm două strategii de Deep Learning în plus față de metodele clasice semi-supervizate din secțiunea anterioară: Multilayer Perceptron Autoencoder (MLP-AE) și Convolution Autoencoder (Conv-AE).

Multilayer Perceptron Autoencoder (MLP-AE) [81] include multiple straturi ascunse atât în encoder, cât și în decoder.

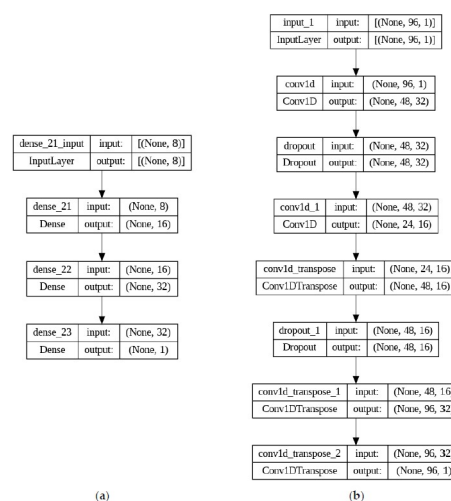


Figura 16: Arhitecturile pentru MLP-AE și Conv-AE [82]

Convolutional Autoencoder (Conv-AE) [83] este compus din encoder, care preia datele de intrare pentru compresie, stratul tip bottleneck ca strat final în encoder, care conține reprezentarea comprimată a datelor de intrare, și decoder, care reconstruiește datele originale de intrare printr-un anumit număr de straturi convoluționale.

5.7 Studiu de caz

Rezultatele afișate în acest studiu de caz, și care sunt obținute cu metodele prezentate în secțiunile anterioare, au fost prima dată publicate în lucrarea [82], publicată în revista Sensors.

Scopul cercetării noastre este să construim un sistem de detecție care folosește cea mai bună tehnică de învățare semi-supervizată din cele cinci pe care le studiem, și anume cele trei metode clasice semi-supervizate Isolation Forest (IF), Local Outlier Factor (LOF) și One-Class Support Vector Machine (OCSVM) și cele două metode de Deep Learning, Multilayer Perceptron Autoencoder (MLP-AE) și Convolutional Autoencoder (Conv-AE).

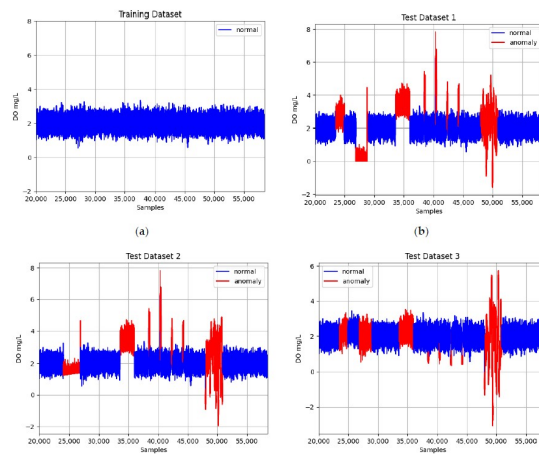


Figura 17: Seturile de testare pentru funcționarea normală și scenariile de anomalii [82]

Tabelul 3 oferă o analiză a celor trei scenarii considerate, examinând tipurile diferite de anomalii și detaliile privind ziua de start și durata fiecărui defect. Acest tabel reprezintă un punct de referință util, permițând o înțelegere clară a numeroaselor aspecte din fiecare scenariu.

Tabelul 3: Scenariile de defecte [82]

Set de date	Defect	Zi de start	Durăță (ore)
Set de date 1	Aeration valve	300	480
	Sensors drift	250	720
	Sensor bias	350	600
	4 × Sensor spike	400, 420, 440, 460	48, 48, 48, 48
	Sensor PD	500	720
Set de date 2: Defecte concurente	Aeration valve + Sensor drift	250	720
	Aeration valve + Sensor bias	350	600
	4 × (Aeration valve + Sensor spike)	400, 420, 440, 460	48, 48, 48, 48
	Aeration valve + Sensor PD	500	720
Set de date 3: defecte concurente + incipiente	80% Aeration valve + 20% Sensor drift	250	720
	80% Aeration valve + 20% Sensor bias	350	600
	4 × (80% Aeration valve + 20% Sensor spike)	400, 420, 440, 460	48, 48, 48, 48
	80% Aeration valve + 20% Sensor PD	500	720

Astfel, graficele din Figura 15 oferă o prezentare generală clară și concisă, furnizându-le cercetătorilor și experților o perspectivă detaliată asupra complexităților sistemului.

Tabelele 4 și 5 conțin hiperparametrii aleși pentru cele trei metode semi-supervizate și pentru cele două metode de Deep Learning.

Observația 3. În cazul în care hiperparametrii nu sunt specificați, se presupune ca ei iau valori standard.

Construim algoritmi pentru cele cinci metode semi-supervizate în mediul Google Colaboratory (Colab), folosind librăriile open-source Python Scikit-Learn 1.2.2 și TensorFlow 2.12 cu Keras, un API Deep Learning de nivel înalt integrat. Beneficiul este că mediul Colab oferă cooperare, accesibilitate și librării deja instalate, eliminând setările complicate. Cu GPU integrat, Colab eficientizează procesul de implementare, permițând o analiză eficientă, partajarea codului și docu-

Tabelul 4: Hiperparametrii metodelor clasice semi-supervizate [82]

Metode clasice	Hiperparametrii
Isolation Forest	n_estimators = 100 random_state = 5
Local Outlier Factor	n_neighbours = 20 metric = 'Euclidean'
One-Class Support Vector Machine	Kernel = 'rbf' degree = 3 gamma = 0.1 nu = 0.05

Tabelul 5: Hiperparametrii metodelor semi-supervizate de Deep Learning [82]

Metode autencoder	Hiperparametrii
Multilayer Perceptron Autoencoder	Encoder: Three layers: 1st—32 neurons, 2nd—16 neurons, 3rd—8 neurons. All with ReLU activation. Decoder: Three layers: 1st—16 neurons with ReLU activation, 2nd—32 neurons with ReLU activation, and 3rd—1 neuron with Sigmoid activation. Compiled with Adam optimizer, learning rate = 0.001, and MAE loss function
Convolutional Autoencoder	Encoder: First Transpose Conv. Layer with 32 filters, 7 kernel size, 'same' padding, 2 strides, and ReLU activation. Dropout layer: 0.2 rate. Second Transpose Conv. Layer with 16 filters, 7 kernel size, 'same' padding, 2 strides, and ReLU activation. Decoder: First Transpose Conv. Layer with 16 filters, 7 kernel size, 'same' padding, 2 strides, and ReLU activation. Dropout layer: 0.2 rate. Second Transpose Conv. Layer with 32 filters, 7 kernel size, 'same' padding, and ReLU activation. Third Transpose Conv. Layer with 1 filter, 7 kernel size, 'same' padding. Compiled with Adam optimizer, learning rate = 0.001, and MSE loss function.

mentarea. Sistemul avansat de detectare a defectelor este creat folosind algoritmul cu cele mai bune metrice de performanță dintre cele cinci metode semi-supervizate. Acesta generează o alarmă și determină timpul de detectare a întârzierii. Sistemul de detecție poate fi aplicat cu succes asupra fluxurilor de date în timp real dacă este definită o fereastră glisantă sau un interval de timp pentru procesarea datelor primite [27], [84], [85].

Tabelul 8 conține metricile de performanță pentru cele cinci tehnici semi-supervizate (IF, LOF, OCSVM, MLP-AE, și Conv-AE) folosite pentru detectarea defectelor din cele trei scenarii stabilite.

Matricile de confuzie din Tabelul 9 demonstrează că, atunci când datele sunt dezechilibrate, metodele tip autoencoder au performanță mai bună decât metodele tradiționale semi-supervizate Isolation Forest (IF), Local Outlier Factor (LOF), și One-Class SVM, în special când datele cu anomalii reprezintă un procent mic din totalul de date.

Metoda semi-supervizată care demonstrează că este cea mai eficientă în toate scenariile atât în ceea ce privește metricile de performanță și matricile de confuzie este metoda Convolutional Autoencoder (Conv-AE). Când comparăm cu tehnicile tradiționale, rezultatele obținute prin cele două metode de autoencoder arată că acestea din urmă sunt substanțial mai eficiente în abordarea problemei de detecție a defectelor în stațiile de epurare.

Tabelul 6: Metricile de performanță pentru cele cinci metode semi-supervizate [82]

Scenariu	Algoritmi	Acuratețe	Precizie	Recall	F1-Score
Set de date 1	IF	79.18	80.68	79.18	79.87
	LOF	73.09	74.97	73.09	73.98
	OCSVM	81.62	75.12	81.62	77.61
	MLP-AE	94.36	94.46	94.36	93.95
	Conv-AE	98.36	98.42	98.36	98.38
Set de date 2	IF	79.36	80.75	79.36	80.01
	LOF	75.36	76.97	75.36	76.14
	OCSVM	83.07	77.86	83.07	79.78
	MLP-AE	92.40	92.57	92.40	91.48
	Conv-AE	97.81	97.80	97.81	97.80
Set de date 3	IF	78.87	80.28	78.87	79.53
	LOF	75.29	76.91	75.29	76.07
	OCSVM	82.43	76.24	82.43	78.75
	MLP-AE	88.70	88.93	88.70	86.11
	Conv-AE	98.64	98.66	98.64	98.65

Tabelul 7: Matricele de confuzie ale celor cinci metode semi-supervizate [82]

Scenariu	IF		LOF		OCSVM		MLP-AE		Conv-AE	
Set de date 1	8521	1380	8165	1736	9418	483	48795	254	48244	710
	1055	737	1411	381	1666	126	3044	6372	246	9170
Set de date 2	8675	1346	8441	1580	9539	482	49267	263	48851	584
	1067	605	1301	371	1498	174	4181	4754	694	8241
Set de date 3	8646	1375	8437	1584	9529	492	49268	262	48933	502
	1096	576	1305	367	1562	110	6346	2589	292	8643

Pasul următor al analizei noastre presupune setarea alarmelor și calcularea timpului de detecție pentru fiecare anomalie folosind sistemul special de detecție construit cu metoda Convolutional Autoencoder (Conv-AE), pe care am identificat-o ca fiind cea mai eficientă strategie, și se obțin rezultatele din Figurile 16 - 18.

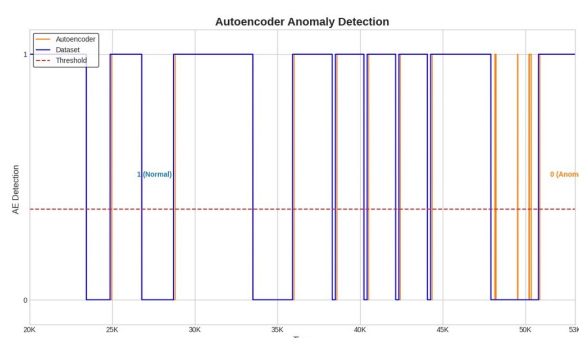


Figura 18: Sistemul de detecție. Detectarea defectelor cu algoritmul Conv-AE pentru Setul de date 1 [82]

Tabelul 8 conține conține timpul de detecție pentru fiecare scenariu de defecte (complete, concurente, complexe) generat de sistemul de detecție bazat pe algoritmul Conv-AE.

Timpul de întârziere obținut arată că sistemul de detecție creat este eficient și poate detecta rapid anomaliile, garantând operarea normală a stației de epurare. Mai mult, comparația rezultatelor

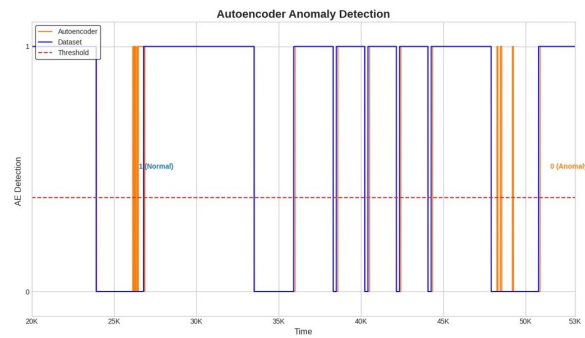


Figura 19: Sistemul de detecție. Detectarea defectelor cu algoritmul Conv-AE pentru Setul de date 2 [82]

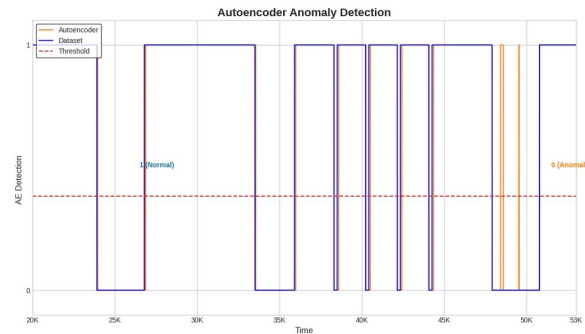


Figura 20: Sistemul de detecție. Detectarea defectelor cu algoritmul Conv-AE pentru Setul de date 3 [82]

Tabelul 8: Timpul de detecție pentru scenariile de anomalii [82]

Scenariu	Defect	Timp (ore)
Set de date 1	Aeration valve	0.72
	Sensors drift	3.84
	Sensor bias	0.72
	4 × Sensor spike	0.96, 0.48, 1.68, 1.68
	Sensor PD	2.4
Set de date 2	Aeration valve + Sensor drift	6
	Aeration valve + Sensor bias	0.72
	4 × (Aeration valve + Sensor spike)	0.96, 0.48, 1.68, 1.68
	Aeration valve + Sensor PD	0.48
Set de date 3	80% Aeration valve + 20% Sensor drift	11.52
	80% Aeration valve + 20% Sensor bias	5.52
	4 × (80% Aeration valve + 20% Sensor spike)	0.96, 1.68, 1.2, 0.96
	80% Aeration valve + 20% Sensor PD	0.48

noastre cu cele din lucrările [22], [27] arată nivelul de eficiență al sistemului nostru de detecție a defectelor.

5.8 Concluzii

În concluzie, rețelele tip FFNN devin metodele standard pentru procesul de recunoaștere de defecte în stațiile de tratare a apelor uzate, asigurând funcționarea precisă, eficientă și sustenabilă a acestora. Metoda Conv-AE devine metoda standard pentru procesul de detecție de defecte în stațiile de tratare a apelor uzate, asigurând o funcționare cât mai eficientă posibil.

Capitolul 6. Concluzii finale

Această teză de doctorat a pătruns în domeniul complicat al tratării apelor uzate, oferind o abordare multidimensională prin cercetarea tehnicilor de control optimal, algoritmilor de optimizare și strategiilor de Deep Learning. Prin metoda Model Predictive Control și programare neliniară, am dezvoltat strategii de control avansate pentru sistemele de tratare, optimizându-le performanța și reducându-le consumul de resurse. Încorporarea algoritmilor de optimizare, în special metodele Coordinate Descent și Augmented Lagrangian au îmbunătățit eficiența diverselor atribuții de control. Apoi, utilizarea metodelor de Deep Learning, a rețelelor neuronale artificiale și a metodelor de învățare semi-supervizată ne-au permis să abordăm probleme cum ar fi recunoașterea și detectarea anomaliilor care apar într-o stație de epurare. Pentru a concluzia, implicațiile acestei lucrări de cercetare se extind dincolo de teză. Aceste direcții oferă o fundație valoroasă pentru continuarea explorării domeniului tratării apelor uzate, având oportunități de rafinare și inovare a strategiilor, în căutarea soluțiilor sustenabile și responsabile cu mediul înconjurător.

6.1 Contribuții originale

Următoarele sunt contribuțiile originale ale acestei teze de doctorat:

Capitolul 3. Control optimal

- Construirea unui algoritm de MATLAB care imită tehnica utilizată de ACADO Toolkit, pentru a stabili un sistem de tratare a apelor uzate. Algoritmul folosește fie funcția *quadprog* din MATLAB, fie programul software de MATLAB Software **CVX**, rezolvând astfel problema MPC scrisă în formă QP (după ce au fost aplicate liniarizare și discretizare asupra sistemului), rezultatul final fiind stabilizarea sistemului de tratare, dar într-un timp de rulare mult mai scurt decât cel de care ACADO Toolkit are nevoie. Scopul este obținerea valorilor de până la 40 mg/l pentru substratul S , menținând în același timp valorile pentru oxigenul dizolvat în intervalul 1 – 3 mg/l și pentru valoarea variabilei de intrare W sub $100m^3/h$ prin acest algoritm de MATLAB.
- Construirea unui algoritm de MATLAB care tratează problema de MPC ca pe una de programare neliniară. Spre deosebire de programul precedent, unde problema era mai întâi liniarizată și discretizată, acest algoritm o rezolvă ca pe o problemă neliniară folosind funcția *fmincon* din MATLAB. Rezultatul final este că sistemul de tratare este stabilizat. Scopul este de a obține valori de până la 40 mg/l pentru substratul S , menținând în același timp valorile pentru oxigenul dizolvat în intervalul 1 – 3 mg/l și pentru valoarea variabilei de intrare W sub $100m^3/h$ prin acest algoritm de MATLAB.

Capitolul 4. Algoritmi de optimizare pentru MPC liniar

- Construirea a șase algoritmi de MATLAB care abordează probleme de optimizare fie în formă scalară, fie în formă de bloc, care utilizează diverse metode de optimizare (CD cu AALM, APCG cu AALM și R-APPROX cu AALM). Acești algoritmi sunt simulați pe șapte sisteme liniare și neliniare (printre care se află și sistemul de tratare a apelor uzate cu nămol activ în patru stări) cu obiectivul de a găsi cea mai bună metodă de optimizare capabilă să rezolve problema MPC derivată din fiecare sistem.

Capitolul 5. Metode de Deep Learning pentru recunoașterea și detectarea defectelor

- Construirea a două modele neuronale (rețea neuronală tip Feedforward și rețea neuronală tip 1d Convolutional) în mediul Google Colaboratory (Colab) folosind librăriile open-source de Python Scikit-Learn 1.2.2 și TensorFlow 2.12 cu Keras. Scopul este compararea acestor metode și găsirea celei mai eficiente pentru a recunoaște diverse defecte injectate în BSM2. Am implementat anomaliile folosind mediul MATLAB Simulink și extragem datele în format CSV.
- Construirea algoritmilor pentru cinci metode semi-supervizate (Isolation Forest, Local Outlier Factor, One-Class Support Vector Machine, Multilayer Perceptron Autoencoder și Convolutional Autoencoder) în mediul Google Colaboratory (Colab) folosind librăriile open-source de Python Scikit-Learn 1.2.2 și TensorFlow 2.12 cu Keras. Scopul este construirea unui sistem de detecție eficient care utilizează cea mai bună metodă din cele cinci, pentru a detecta diverse anomalii injectate în BSM2. Am implementat anomaliile folosind mediul MATLAB Simulink și extragem datele în format CSV.

6.2 Diseminarea rezultatelor

Această secțiune constă din colecția de lucrări de conferințe și articole de jurnal care conțin rezultatele prezentate în această teză de doctorat, în ordine cronologică:

Următoarele sunt lucrări publicate în reviste indexate ISI:

- **Liliana Maria Ghinea**, Daniela Lupu, Marian Barbu, Ion Necoară, "Fast Coordinate Descent Augmented Lagrangian Methods for linearized MPC", Journal of Control Engineering and Applied Informatics, Vol. 25, Nr. 1, pp. 49-58, 2023, **indexat ISI, Factor de impact 0.9**, ISSN 1454-8658, link: <http://www.ceai.srait.ro/index.php?journal=ceai&page=article&op=view&path%5B%5D=8332>.
- **Liliana Maria Ghinea**, Mihaela Miron, Marian Barbu, "Semi-Supervised Anomaly Detection of Dissolved Oxygen Sensor in Wastewater Treatment Plants", Sensors, Vol. 23, Nr. 19, 8022, 2023, **indexat ISI, Factor de impact 3.9**, ISSN: 1424-8220, link: <https://www.mdpi.com/1424-8220/23/19/8022>.

Următoarele sunt lucrări prezentate la conferințe internaționale:

- **Liliana Maria Ghinea**, Ion Necoară, Marian Barbu, "Random coordinate descent methods for non-separable composite optimization", 2021 European Control Conference (ECC), 29 iunie - 2 iulie 2021, conferință virtuală, pp. 2225-2230, **IEEE**.
- Daniela Lupu, Ion Necoară, Tudor C Ionescu, **Liliana Maria Ghinea**, Joseph Garrett, Tor A Johansen, "Dimensionality reduction of hyperspectral images using an ICA-based stochastic second-order optimization algorithm", 2023 European Control Conference (ECC), 13 - 16 iunie 2023, București, România, pp. 1-6, **IEEE**.
- **Liliana Maria Ghinea**, Daniela Lupu, Marian Barbu, Ion Necoară, "Control of a wastewater treatment process using linear and nonlinear model predictive control", 28th International Conference on Emerging Technologies and Factory Automation (ETFA), 12 - 15 septembrie 2023, Sinaia, România, pp. 1-6, **IEEE**.

- **Liliana Maria Ghinea**, Mihaela Miron, Harsha Ratnaweera, "A Deep Learning Approach For Faults Recognition of Dissolved Oxygen Sensor in Wastewater Treatment Plants", 28th International Conference on Emerging Technologies and Factory Automation (ETFAs), 12 - 15 septembrie 2023, Sinaia, România, pp. 1-5, **IEEE**.

Următoarele sunt prezentări poster sau de abstract extins:

- **Liliana Maria Ghinea**, Ion Necoara, Marian Barbu, "Stochastic Subspace Higher Order Methods", Scientific Conference of Doctoral Schools, June 18 - June 19 2020, Galați, România, **prezentare poster**.
- **Liliana Maria Ghinea**, Ion Necoara, Marian Barbu, "Wastewater Treatment Systems in ACADO Toolkit", Scientific Conference of Doctoral Schools, June 10 - June 11 2021, Galați, România, **prezentare poster**.
- **Liliana Maria Ghinea**, Ion Necoara, Marian Barbu, Daniela Lupu, "Optimal Control of the Wastewater Treatment Process in ACADO Toolkit", 2023 European Control Conference (ECC), June 13 - June 16 2023, București, România, **abstract extins**.

6.3 Direcții viitoare de cercetare

Concluzionând această teză de doctorat, se remarcă o serie de direcții viitoare de cercetare promițătoare care pot îmbunătăți starea sistemelor de tratare a apelor uzate prin integrarea metodelor de control optimal, a algoritmilor de optimizare și a strategiilor de Deep Learning. Astfel de direcții pot fi: explorarea interacțiunilor dintre Deep Learning, programare neliniară și Model Predictive Control într-un cadru hibrid de control, încorporarea metodelor hibrid supervizate - ne-supervizate pentru identificarea anomaliilor atât pentru sisteme simulate, cât și pentru stații de epurare reale, și de asemenea aplicarea cercetării noastre pe date obținute de la alți senzori față de senzorul de oxigen dizolvat, sau pe tipuri diferite de defecte (biologice, chimice, hidraulice etc.).

Bibliografie

- [1] J. Alex, L. Benedetti, J. Copp, K.V. Gernaey, U. Jeppsson, I. Nopens, M.N. Pons, C. Rosen, J.P. Steyer, P. Vanrolleghem, *Benchmark Simulation Model no. 2 (BSM2)*, Report by the IWA Taskgroup on benchmarking of control strategies for WWTP, 2008.
- [2] C. Cassardo, J. A. A. Jones, *Managing water in a changing world*, Water, vol. 3(2), pp. 618–628, 2011.
- [3] B. Houska, H. J. Ferreau, M. Diehl, *ACADO toolkit—An open-source framework for automatic control and dynamic optimization*, Optimal Control Applications and Methods, vol. 32, pp. 298-312, 2011.
- [4] M. Sbarciog, J. A. Moreno, A. Vande Wouwer, *A Biogas-Based Switching Control Policy for Anaerobic Digestion Systems*, 8th IFAC Symposium on Advanced Control of Chemical Processes, The International Federation of Automatic Control, Singapore, July 10-13, 2012.
- [5] M. Barbu, S. Caraman, E. Ceanga, *QFT robust control of biotechnological processes*, IEEE-TTTC International Conference on Automation, Quality and Testing, Robotics, AQTR 2004, pp. 129-134, 2004.
- [6] M. Barbu, S. Caraman, E. Ceanga, *QFT robust control of a wastewater treatment process*, IFAC Proceedings Volumes, vol. 38(1), pp. 109-114, 2005.
- [7] M. Barbu, S. Caraman, *QFT multivariable control of a biological wastewater treatment process using ASM1 model*, IFAC Proceedings Volumes, vol. 40(4), pp. 295-300, 2007.
- [8] F. Nejjari, B. Dahhou, A. Benhammou, G. Roux, *Non-linear multivariable adaptive control of an activated sludge wastewater treatment process*, Adaptive Control and Signal Processing, vol. 15(5), pp. 347-365, 1999.
- [9] M. Henze, C.P.L Grady Jr., W. Gujer, G.v.R. Marais, T. Matsuo, *Activated Sludge Model no. 1*, (1987) , Report by IAWQ Scientific and Technical, UK, 1987.
- [10] S. Cristea, C. de Prada, D. Sarabia, G. Gutiérrez, *Aeration control of a wastewater treatment plant using hybrid NMPC*, Computers & Chemical Engineering, vol. 35(4), pp. 638-650, 2011.
- [11] I. Vasiliev, I. Luca, L. Condrachi, L. Luca, M. Barbu, R. Vilanova, S. Caraman, *Model Predictive Control of a wastewater treatment process using neural networks*, 28th International Conference on Emerging Technologies and Factory Automation, Sinaia, Romania, 12-15 September 2023.

- [12] D. Sendrescu, D. Popescu, E. Petre, E. Bobasu, D. Selisteanu, *Nonlinear model predictive control of a lipase production bioprocess*, Carpathian Control Conference (ICCC), pp. 337-341, 2011.
- [13] S. E. Benattia, S. Tebbani, D. Dumur, D. Selisteanu, *Robust Nonlinear Model Predictive Controller based on sensitivity analysis—Application to a continuous photobioreactor*, 2014 IEEE Conference on Control Applications (CCA), pp. 1705-1710, 2014.
- [14] F. Chorobura, I. Necoara, *Random Coordinate Descent Methods for Nonseparable Composite Optimization*, SIAM Journal on Optimization, vol. 33(3), 2023.
- [15] I. A. C. P. Espirito Santo, E. M. G. P. Fernandes, M. M. Araujo, E. C. Ferreira, *Cost minimization of a WWTP using an augmented Lagrangian pattern search based solver*, 10th IWA Specialised Conference Design, Operation and Economics of Large Wastewater Treatment Plants, Vienna, Austria, 9-13 September 2007.
- [16] I. A. C. P. Espirito Santo, E. M. G. P. Fernandes, M. M. Araujo, E. C. Ferreira, *An Augmented Lagrangian Pattern Search Method for Optimal WWTP Designs*, 6th WSEAS International Conference on System Science and Simulation in Engineering, Venice, Italy, 21 - 23 November 2007.
- [17] I. A. C. P. Espirito-Santo, L. Costa, R. Denysiuk, E. M. G. P. Fernandes, *Hybrid genetic pattern search augmented Lagrangian algorithm: application to WWTP optimization*, 2nd International Conference on Applied Operational Research, Turku, Finland, 5 - 27 August 2010.
- [18] M. Miron, L. Frangu, S. Caraman, L. Luca, *Artificial neural network approach for fault recognition in a wastewater treatment process*, in 2018 22nd International Conference on System Theory, Control and Computing (ICSTCC), pp. 634–639, 2018.
- [19] K. Wang, P. Chang, F. Meng, *Monitoring of wastewater treatment process based on slow feature analysis variational autoencoder*, in 2021 IEEE 10th Data Driven Control and Learning Systems Conference (DDCLS). IEEE, pp. 495–502, 2021.
- [20] P. Ramin, X. Flores-Alsina, S. O. Nymann Topalian, U. Jeppsson, and K. Gernaey, *Fault detection in a benchmark simulation model for wastewater treatment plants*, Computer Aided Chemical Engineering, vol. 49, pp. 1363-1368, 2022.
- [21] R. Salles, J. Mendes, R. P. Ribeiro, J. Gama, *Fault Detection in Wastewater Treatment Plants: Application of Autoencoders Models with Streaming Data*, section in book “Machine Learning and Principles and Practice of Knowledge Discovery in Databases”, pp. 55-70, 2023.
- [22] X. Li, W. Chai, T. Liu, J. Qiao, *Fault detection of dissolved oxygen sensor in wastewater treatment plants*, IECON 2020 The 46th Annual Conference of the IEEE, 2020.
- [23] M. Miron, Teză de doctorat – *Contribuții privind diagnoza proceselor de tratare a apelor uzate utilizând rețele neuronale*, Universitatea “Dunărea de Jos”, Școala doctorală de Științe Fundamentale și Ingineresti, Galați, 2018.

- [24] D. Wang, S. Thunéll, U. Lindberg, L. Jiang, J. Trygg, M. Tysklind, N. Souihi, *A machine learning framework to improve effluent quality control in wastewater treatment plants*, Science of The Total Environment, vol. 784, 2021.
- [25] H. Xiao, D. Huang, Y. Pan, Y. Liu, K. Song, *Fault diagnosis and prognosis of wastewater processes with incomplete data by the auto-associative neural networks and ARMA model*, Chemom. Intell. Lab. Syst., vol. 161, pp. 96–107, 2017.
- [26] R. Salles, J. Mendes, R. Ribeiro, J. Gama, *Fault Detection in Wastewater Treatment Plants: Application of Autoencoders Models with Streaming Data*, European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, Grenoble, FRANCE, pp. 55-70, 19-23 September 2022.
- [27] R. Salles, J. Mendes, C. Henggeler Antunes, P. Moura, J. Dias, *Dynamic Setpoint Optimization Using Metaheuristic Algorithms for Wastewater Treatment Plants*, IECON 2022 – 48th Annual Conference of the IEEE Industrial Electronics Society, Brussels, Belgium, pp. 1-6, 17-20 October 2022.
- [28] Raport științific de etapă, director proiect: Sergiu Caraman, *Îmbunătățirea indicatorilor calitativi la tratarea biologică a apelor reziduale din industria alimentară pe baza unor sisteme de conducere avansată – APEPUR*, Grant CEEEX – MENER nr. 717/24.07.2006, pp. 7 -11, 2006.
- [29] S. Caraman, M. Sbarciog, M. Barbu, *Predictive control of a wastewater treatment process*, IFAC Proceedings Volumes, vol. 39(14), pp. 155 - 160, 2006.
- [30] J. Alex, L. Benedetti, J. Copp, K.V. Gernaey, U. Jeppsson, I. Nopens, M.N. Pons, J.P. Steyer, P. Vanrolleghem, *Benchmark Simulation Model no. 1 (BSM1)*, Report by the IWA Taskgroup on benchmarking of control strategies for WWTP, 2008.
- [31] I. Nopens, L. Bennedetti, U. Jeppsson, M.-N. Pons, J. Alex, J. B. Coop, K. V. Gernaey, C. Rosen, J.-P. Steyer, P. A. Vanrolleghem, *Benchmark simulation model No 2 - Finalisation of plant layout and default control strategy*, Water Science and Technology, vol. 62(9), pp. 1967–1974, 2010.
- [32] F. Allgower, A. Zheng, *Nonlinear Model Predictive Control*, Progress in Systems and Control Theory vol. 26, Springer Basel AG, 1991.
- [33] B. Dasarathy, *Optimal control of nonlinear systems*, IEEE Transactions on Automatic Control, vol. 15(6), pp. 690–692, 1970.
- [34] D. Ariens, B. Houska, H. Ferreau, *Acado toolkit website*, <http://www.acadotoolkit.org>.
- [35] D. Ariens, B. Houska, H. Ferreau, F. Logist, *ACADO for Matlab User's Manual*, Optimization in Engineering Center (OPTEC), 1.0 beta edition, May 2010.
- [36] L.M. Ghinea, D. Lupu, M. Barbu, I. Necoara, *Control of a wastewater treatment process using linear and nonlinear model predictive control*, 28th International Conference on Emerging Technologies and Factory Automation (ETFA), Sinaia, Romania, pp. 1-6, 12-15 September 2023.

- [37] T. I. Lakoba, *Simple Euler method and its modifications*, Lecture notes, University of Vermont, 2012.
- [38] M.S. Bazaraa, C.M. Shetty, *Nonlinear programming. Theory and algorithms*, John Wiley & Sons, United States, 1979.
- [39] I. Necoara, *Metode de optimizare numerica*, Editura Politehnica Press, Bucuresti, 2013.
- [40] ***, Documentation on fmincon, <https://www.mathworks.com/help/optim/ug/fmincon.html>.
- [41] L. Socha, *Linearization Methods for Stochastic Dynamic Systems*, Springer, 2007.
- [42] J. Berberich, J. Kohler, M. A. Muller, F. Allgower, *Linear Tracking MPC for Nonlinear Systems—Part I: The Model-Based Case*, IEEE Transactions on Automatic Control, vol. 67 (9), pp. 4390–4405, 2022.
- [43] ***, <https://www.mathworks.com/help/optim/ug/quadprog.html>.
- [44] ***, <https://cvxr.com/cvx/>.
- [45] Y. Nesterov, *Efficiency of Coordinate Descent Methods on Huge-Scale Optimization Problems*, SIAM Journal on Optimization, vol. 22(2), 2012.
- [46] Yu. Nesterov, *Introductory lectures on convex optimization. A basic course*, Kluwer Academic Publishers, 2004.
- [47] R. Andreani, E. G. Birgin, J. M. Martínez, M. L. Schuverdt, *On Augmented Lagrangian Methods with General Lower-Level Constraints*, SIAM Journal on Optimization, vol. 18(4), pp. 1286–1309, 2007.
- [48] L.M. Ghinea, D. Lupu, M. Barbu, I. Necoara, *Fast Coordinate Descent Augmented Lagrangian Methods for linearized MPC*, Journal of Control Engineering and Applied Informatics, vol. 25(1), pp. 49-58, 2023.
- [49] D. P. Bertsekas, *Nonlinear programming*, Journal of the Operational Research Society, vol. 48(3), pp. 334–334, 1997.
- [50] D. P. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods*, Athena Scientific, 1996.
- [51] Y. Nesterov, *Smooth minimization of non-smooth functions*, Mathematical programming, vol. 103(1), pp. 127–152, 2005.
- [52] L. Wu, A. Bemporad, *A Simple and Fast Coordinate-Descent Augmented-Lagrangian Solver for Model Predictive Control*, in IEEE Transactions on Automatic Control (Early Access), 2023.
- [53] D. P. Bertsekas, *Chapter 2 - The Method of Multipliers for Equality Constrained Problems*, in "Constrained Optimization and Lagrange Multiplier Methods", Academic Press, pp. 95–157, 1982.

- [54] B. He, X. Yuan, *On the acceleration of augmented Lagrangian method for linearly constrained optimization*, Computer Science, vol. 3, 2010, <https://optimizationonline.org/2010/10/2760/>.
- [55] Q. Lin, Z. Lu, L. Xiao, *An Accelerated Proximal Coordinate Gradient Method and its Application to Regularized Empirical Risk Minimization*, SIAM Journal on Optimization, vol. 25(4), 2015.
- [56] O. Fercoq, Z. Qu, *Restarting the accelerated coordinate descent method with a rough strong convexity estimate*, Computational Optimization and Applications, vol. 75, pp. 63–91, 2020.
- [57] P. Kamasouris, M. Athans, G. Stein, *Design of feedback control systems for stable plants with saturating actuators*, 27th IEEE Conference on Decision and Control, Austin, Texas, U.S.A., pp. 469–479, 1988.
- [58] A. Bemporad, A. Casavola, and E. Mosca, *Nonlinear control of constrained linear systems via predictive reference management*, IEEE transactions on Automatic Control, vol. 42(3), pp. 340–349, 1997.
- [59] P. Tondel, Tor A. Johansen, *Complexity reduction in explicit linear model predictive control*, IFAC Proceedings Volumes, vol. 35(1), pp. 189–194, 2002.
- [60] S. Longo, E. C. Kerrigan, G.A Constantinides, *A predictive control solver for low-precision data representation*, European Control Conference (ECC), Zurich, Switzerland, 17-19 July 2013.
- [61] U. Mäder, R. Cagienard, M. Morari, *Explicit Model Predictive Control*, Lecture Notes in Control and Information Sciences (LNCIS), 2007.
- [62] R. Annamalai, *Discrete time Linear Quadratic (LQ) optimal control Vs MPC: Integral action and handling constraints*, 2013.
- [63] B. Bequette, *Process Dynamics: Modeling, Analysis and Simulation*, Prentice-Hall, Module 8, pp. 641-660, 1998.
- [64] ***, CSTR model, <https://www.mathworks.com/help/mpc/gs/cstr-model.html>.
- [65] L. Hardesty, *Explained: Neural networks*, MIT News Office, retrieved 2022.
- [66] C. E. Metz, *Basic principles of ROC analysis*, Semin Nucl Med. vol. 8(4), pp. 283–298, 1978.
- [67] D. M. W. Powers, *Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness & Correlation*, Journal of Machine Learning Technologies, vol. 2(1), pp. 37–63, 2011.
- [68] T. Fawcett, *An Introduction to ROC Analysis*, Pattern Recognition Letters, vol. 27(8), pp. 861–874, 2006.
- [69] ***, Metrics and scoring, https://scikit-learn.org/stable/modules/model_evaluation.html.

- [70] S. U. Jan, Y. D. Lee, I. S. Koo, *A distributed sensor-fault detection and diagnosis framework using machine learning*, Information Sciences, vol. 547, pp. 777–796, 2021.
- [71] S. U. Jan and I. Koo, *A novel feature selection scheme and a diversified-input SVM-based classifier for sensor fault classification*, Journal of Sensors, vol. 2018, 2018.
- [72] R. Liu, B. Yang, E. Zio, X. Chen, *Artificial intelligence for fault diagnosis of rotating machinery: A review*, Mechanical Systems and Signal Processing, vol. 108, pp. 33–47, 2018.
- [73] J. Schmidhuber, *Deep learning in neural networks: An overview*, Neural Networks, vol. 61, pp. 85–117, 2015.
- [74] R. Venkatesan, B. Li, *Convolutional Neural Networks in Visual Computing: A Concise Guide*, CRC Press, US, 2017.
- [75] L.M. Ghinea, M. Miron, H. Ratnaweera, *A Deep Learning Approach For Faults Recognition of Dissolved Oxygen Sensor in Wastewater Treatment Plants*, 28th International Conference on Emerging Technologies and Factory Automation (ETFA), Sinaia, Romania, pp. 1-5, 12-15 September, 2023.
- [76] ***, Keras-Tuner, https://keras.io/keras_tuner/.
- [77] O. Chapelle, B. Schölkopf, A. Zien, *Semi-supervised learning*, Cambridge, Mass., MIT Press, 2006.
- [78] F. T. Liu, K. M. Ting, Z. H. Zhou, *Isolation Forest*, 2008 Eighth IEEE International Conference on Data Mining, Italy, pp. 413–422, 2008.
- [79] M.M. Breunig, H.P. Kriegel, R.T. Ng, J. Sander, *LOF: Identifying Density-Based Local Outliers*, ACM SIGMOD Rec., vol. 29, pp. 93–104, 2000.
- [80] B. Schölkopf, J.C. Platt, J. Shawe-Taylor, A. Smola, R.C. Williamson, *Estimating the Support of a High-Dimensional Distribution*, Neural. Comput., vol. 13, pp. 1443–1471, 2001.
- [81] R.G. Vieira, M. Dhimish, F.M.U. de Araújo, M.T. da Silva Guerra, *Comparing multilayer perceptron and probabilistic neural network for PV systems fault detection*, Expert. Syst. Appl. vol. 201, 117248, 2022.
- [82] L.M. Ghinea, M. Miron, M. Barbu, *Semi-Supervised Anomaly Detection of Dissolved Oxygen Sensor in Wastewater Treatment Plants*, Sensors, vol. 23(19), 8022, 2023.
- [83] D. Jana, J. Patil, S. Herkal, S. Nagarajaiah, L. Duenas-Osorio, *CNN and Convolutional Autoencoder (CAE) based real-time sensor fault detection, localization, and correction*, Mech. Syst. Signal Process, vol. 169, 108723, 2022.
- [84] I.T. Nicholas, J.R. Park, K. Jung, J.S. Lee, D.K. Kang, *Anomaly Detection of Water Level Using Deep Autoencoder*, Sensors vol. 21, 2021.
- [85] P. Park, P.D. Marco, H. Shin, J. Bang, *Fault Detection and Diagnosis Using Combined Autoencoder and Long Short-Term Memory Network*, Sensors. vol. 19, 2019.