

Algoritmi de Învățare cu Întărire cu Garanții de Funcționare în Regim de Siguranță pentru Sisteme Dinamice Complexe Neliniare cu Model Necunoscut

Teză de doctorat – Rezumat

pentru obținerea titlului științific de doctor

Universitatea Politehnica Timișoara

în domeniul de doctorat Ingineria Sistemelor

Autor: Timotei LALA

Conducător științific: Prof. Univ. Dr. Ing. Ioan SILEA

Mai 2026

Economia contemporană se bazează pe sisteme dinamice complexe (infrastructuri critice, procese industriale, sisteme robotice) guvernate de mecanisme de reglare automată în buclă închisă. Stabilitatea și funcționarea în regim de siguranță (RS) sunt fundamentale, deoarece eșecurile pot avea consecințe catastrofale.

Recent, s-au dezvoltat metode de acordare a reguletoarelor bazate pe **Învățare cu Întărire (ÎÎ)** sau Programare Dinamică Adaptivă (PDA), care învață din date istorice fără a necesita modele matematice explicite. ÎÎ combină explorarea (testarea comenzilor aleatorii) cu exploatarea (optimizarea reguletoarelor). Progresele în învățarea profundă și rețelele neuronale au permis succese remarcabile în sisteme simulate, jocuri (AlphaZero, AlphaGO, Agent57), sisteme de recomandare și chiar modele de limbaj. Însă tranziția către aplicații reale creează o tensiune fundamentală: învățarea necesită explorare, funcționarea în regim de siguranță o limitează. În sistemele de control reale, spre deosebire de cele simulate, eșecurile sunt ireversibile și inacceptabile. Scopul procesului de explorare este de a introduce comenzi secvențiale aleatorii la intrarea de control a sistemului condus și de a observa efectele acestora asupra întregii bucle de reglare. Însă, explorarea fără constrângeri poate încălca RS, iar restricționarea excesivă împiedică obținerea reguletoarelor optime.

Totodată, pe lângă menținerea RS în timpul explorării, există riscul ca reguletoarele învățate să fie destabilizante din cauza erorilor de aproximare, explorării insuficiente sau hiper-parametrilor greșiți. Provocarea centrală este garantarea funcționării în RS atât în timpul explorării, cât și în timpul testării reguletoarelor învățate - punct de plecare pentru contribuțiile acestei teze.

Pentru a răspunde limitărilor prezentate anterior, s-a dezvoltat subdomeniul Învățării cu Întărire în Regim de Siguranță (ÎÎRS) (engl. Safe Reinforcement Learning), care introduce o serie de algoritmi care asigură funcționarea

Școala Doctorală de Studii Inginerești

sistemelor în RS pe parcursul procesului de învățare. Spre deosebire de algoritmi de ÎÎ clasici, ÎÎRS tratează siguranța ca o cerință explicită, formulată sub forma unei constrângeri asupra stărilor, comenzilor sau traiectoriilor sistemului.

Literatura existentă poate fi organizată în mai multe direcții de studiu. O primă clasă de algoritmi se bazează pe formulări ale problemei de optimizare cu constrângeri, în care funcția obiectiv se mărește cu expresii de tip inegalitate care penalizează încălcarea condițiilor de siguranță. Aceste abordări oferă un cadru formal matematic elegant, folosind metode de optimizare clasice precum metoda multiplicatorilor Lagrange sau optimizare primară-duală în cadrul proceselor Markov de decizie cu constrângeri. În [6] a fost dezvoltată o metodă de optimizare cu constrângeri pentru a învăța un regulator care garantează satisfacerea aproximativă de la fiecare iterație de învățare, pe baza metodei derivate din metoda de optimizare a regulatorului cu regiuni de încredere (engl. Trust Region Policy Optimization). În [7] s-a dezvoltat o metodă de a incorpora constrângerile aferente proceselor Markov de decizie cu constrângeri în funcția recompensă, dezvoltând o metodă generală de rezolvare ale acestor probleme. Deși aceste metode oferă un cadru formal matematic elegant, ele nu garantează lipsa încălcărilor constrângerilor în timpul etapei de învățare, ceea ce este inacceptabil pentru multe aplicații practice.

A doua clasă de metode este reprezentată de algoritmi de filtrare ale comenzilor cu potențial destabilizant, algoritmi numiți “safety layers”. Aceste metode sunt exogene optimizării ÎÎ, acționând la ieșirea regulatorului ca niște filtre ale comenzii pe baza unor certificate de stabilitate. Printre aceste metode se disting algoritmi care se bazează pe funcțiile barieră de control (FBC) (engl. control barrier function). În timpul execuției sistemului condus, este necesar ca regimul de funcționare să rămână în interiorul FBC predefinit pentru a garanta funcționarea în RS. În [8], autorii prezintă un regulator hibrid care combină ÎÎ fără model și algoritmul FBC cu modelul matematic învățat de la sistem, utilizând metoda Trust Region Policy Optimization (TRPO) pentru actualizarea adaptivă a regulatorului și procese Gaussiene pentru modelarea dinamică a sistemului, asigurând funcționarea în RS prin optimizare pătratică în timp real. Totodată, folosirea certificatelor de învățare permit modificarea comenzilor regulatorului pentru a garanta funcționarea sistemului în RS, după cum se prezintă în [9]-[11]. Acestea însă generează un comportament conservator în timpul rulării, fiind dependente de calitatea modelului identificat sau a certificatului învățat din date.

A treia clasă de metode utilizează concepte din teoria sistemelor precum funcțiile Lyapunov cu control sau seturile invariante, pentru a oferi garanții formale de siguranță. În [12], autorii au utilizat o metodă de verificare a stabilității Lyapunov și modele statistice ale dinamicii pentru a obține un control de înaltă performanță. Pe baza proceselor de decizie Markov cu constrângeri, metoda descrisă în [13] garantează funcționarea în RS prin construirea unor constrângeri liniare locale pe baza funcțiilor Lyapunov. Această clasă de algoritmi necesită însă fie cunoașterea precisă a ecuațiilor de dinamică a sistemului, fie a unui model nominal inițial al acestuia care se îmbunătățește cu ajutorul datelor.

Deși domeniul a înregistrat progrese semnificative, algoritmi existenți prezintă multiple limitări. Multe metode necesită cunoașterea ecuațiilor de dinamică ale sistemului pentru a prezice evoluția viitoare și a selecta comenzile care mențin sistemul în regim sigur (RS). Însă obținerea acestor ecuații pentru sisteme complexe neliniare este dificilă. Unele abordări nu garantează stabilitatea în buclă închisă sau funcționarea în RS pe parcursul explorării, ci doar la convergență. Alți algoritmi sunt limitați la anumite clase de sisteme, nefiind aplicabili sistemelor neliniare generale continue. Majoritatea metodelor nu oferă garanții teoretice de stabilitate sau siguranță, fiind prezentate doar ca metode euristice.

Aceste limitări evidențiază necesitatea unor algoritmi de ÎÎ pentru sisteme neliniare generale cu model necunoscut, care să ofere garanții explicite de siguranță atât în timpul explorării, cât și în timpul evaluării reguletoarelor învățate.

Obiectivele cercetării. Pe baza analizei literaturii științifice menționate anterior, această teză își propune să analizeze metodic problema de menținere a stabilității sistemului în buclă închisă cu regulatorul acordat cu ÎÎ și de

Școala Doctorală de Studii Inginerești

funcționare a acestuia într-un RS, în vederea îmbunătățirilor metodelor curente și a rezolvării problemelor existente din domeniu. Concret, această teză de doctorat propune următoarele obiective de cercetare (OC):

OC1. Analiza literaturii de specialitate în domeniul ÎÎRS cu scopul de a identifica limitările metodelor existente de menținere a sistemelor conduse în RS.

OC2. Dezvoltarea unor garanții teoretice de convergență a algoritmilor de ÎÎ, pentru a demonstra capacitatea procesului de învățare de a obține regulatorul și funcția de cost optimală.

OC3. Dezvoltarea unei metode de menținere a stabilității și a funcționării în RS a unui sistem cu model necunoscut în buclă închisă cu un regulator învățat cu algoritmi de ÎÎ.

OC4. Dezvoltarea unei metode de menținere a stabilității și a funcționării în RS a sistemului în procesul de explorare a domeniului stare-acțiune.

OC5. Validarea practică a acestor algoritmi pe mai multe clase de sisteme neliniare în vederea demonstrării aplicabilității metodei în mai multe domenii de activitate.

Primul obiectiv de cercetare are ca scop înțelegerea comprehensivă a domeniului ÎÎRS, atât din literatura de specialitate aferentă domeniului Inteligenței Artificiale, cât și cea din domeniul reglării automate. Datorită faptului că literatura de specialitate din domeniul Inteligenței Artificiale abordează regulatoare și sisteme stohastice de tip Procese Markov de Decizie, aceasta s-a dezvoltat paralel cu literatura din domeniul reglării automate care tratează regulatoarele și sistemele deterministe. Totodată, din cauza faptului că metodele de ÎÎRS s-au dezvoltat inițial în literatura aferentă domeniului Inteligenței Artificiale, scopul urmărit la acest pas este acela de a înțelege și analiza metodele dezvoltate, atât în vederea adaptării acestora în domeniul de control cât și pentru a găsi noi soluții la problemele nerezolvate. Al doilea obiectiv de cercetare urmărește dezvoltarea unor garanții teoretice de convergență a algoritmilor de învățare de tip ÎÎ. Garanțiile de convergență sunt esențiale pentru asigurarea unui comportament predictibil și determinist al algoritmilor, aspect critic în aplicațiile cu constrângeri de funcționare în RS, oferind totodată baze teoretice solide pentru analiza și compararea diversilor algoritmi. Cel de-al treilea obiectiv abordează problema produsă de un proces de învățare eronat, care poate genera regulatoare care fie destabilizează sistemul, fie îl scot din RS. Astfel că este necesară dezvoltarea unei metode de menținere a stabilității și a funcționării în RS a unui sistem cu model necunoscut în buclă închisă cu un regulator obținut în urma iterațiilor procesului de învățare aferent metodei de ÎÎ. Următorul obiectiv al acestei teze de doctorat este cel al menținerii stabilității și a funcționării în RS a sistemului în procesul de explorare al domeniului stare-acțiune. Această necesitate vine din impredictibilitatea și complexitatea pe care zgomotul aditiv exploratoriu de la intrarea de comandă îl adaugă buclei de reglare, ceea ce poate avea ca efect destabilizarea sistemului și totodată ieșirea acestuia din RS. Ultimul obiectiv adresează validarea tuturor metodelor anterior dezvoltate pentru atingerea obiectivelor precedente. Validarea trebuie realizată pe multiple sisteme din domenii de activitate disjuncte, pentru a valida generalizarea acestor metode pe o clasă mai largă de dinamici neliniare.

Contribuții majore. În această teză de doctorat sunt propuse sistematic multiple metode de menținere a stabilității sistemului în buclă închisă cu regulatorul acordat cu ÎÎ și de funcționare a acestuia într-un RS. Prin tratarea acestor probleme, teza își propune să contribuie la dezvoltarea unor metode de ÎÎ atât teoretic fundamentate, cât și aplicabile în contexte reale, unde siguranța nu poate fi compromisă în niciun moment. Detaliile acestor contribuții majore (CM), prezentate și în lucrările de specialitate ale autorului tezei curente sunt prezentate în continuare:

CM1. Analiza convergenței algoritmului Programării Dinamice Euristice Dependente de Acțiune cu Loturi în mod Offline (PDEDA-LO) pentru aproximatori de tip RN multistrat.

Școala Doctorală de Studii Inginerești

CM2. Dezvoltarea unei metode de menținere în RS a sistemului cu model necunoscut condus de un regulator acordat cu metoda PDA cu Constrângeri cu Gradientul Politicii Deterministe (PDACGPD).

CM3. Dezvoltarea unei metode de menținere a stabilității și a funcționării în RS a unui sistem cu model necunoscut folosind un regulator inițial stabilizant în comutație cu regulatorul acordat PDA, numită IV fără model în comutație în regim de siguranță (IVFMCRS).

CM4. Dezvoltarea unei metode de menținere a funcționării în regimul de siguranță în timpul procesului de explorare, dezvoltându-se algoritmul Explorare Iterativă fără model în regim de siguranță (EIFMRS).

Teza este structurată în 8 capitole care tratează succesiv etapele de abordare ale contribuțiilor majore prezentate anterior. În Capitolul 1 se prezintă introducerea, motivația, obiectivele de cercetare împreună cu direcțiile de dezvoltare majoră ale acestei teze de doctorat. În Capitolul 2 se introduc aspecte fundamentale teoretice ale algoritmilor ÎÎ, precum algoritmul Iterație după Valoare (IV) și algoritmul Iterație după Legea de Reglare (ILR), implementarea lor în funcții Q și teoria de convergență a algoritmului IV, atât în cazul funcțiilor de valoare, cât și pentru cazul funcțiilor Q, cu și fără aproximatori de funcții. În Capitolul 3 se analizează stabilitatea teoretică a algoritmului PDEDA-LO folosind analiza de convergență UAM cu funcții Lyapunov. Validarea superiorității performanței algoritmului PDEDA-LO cu RN multi-strat față de implementarea cu RN mono-strat se realizează pe un sistem aerodinamic cu două rotoare (SADR), demonstrând performanțele de reglare îmbunătățite în cazul aproximatorilor de tip RN multistrat. În Capitolul 4 se introduce implementarea algoritmului PDACGPD cu aproximatori de funcții. Analiza de convergență UAM cu funcții Lyapunov se utilizează pentru demonstrația de convergență a algoritmului PDACGP. Implementarea practică se realizează cu aproximatori de funcție de tip RN, iar convergența UAM se realizează cu ajutorul funcțiilor Lyapunov. Algoritmul este validat pe un proces de tip turbină eoliană și pe un sistem de tip suspensie activă, demonstrând capacitatea algoritmului de a învăța un regulator optimal și de a limita comenziile generate într-un interval predefinit față de algoritmul PDAGPD. În Capitolul 5 se aplică metoda PDACGPD pentru acordarea reguletoarelor liniare generale, validarea realizându-se pe un sistem neliniar de ordinul 3. În Capitolul 6 se dezvoltă algoritmul IVFMCRS, împreună cu analiza de stabilitate FLM și mecanismul de detecție timpurie a instabilității. Validarea acestui algoritm se realizează pe un sistem liniar și pe SFE cu ajutorul MVS, demonstrând capacitatea acestui algoritm de a menține în RS sistemul condus în cazul obținerii unui regulator destabilizant în urma procesului de învățare. În Capitolul 7 se descrie algoritmul EIFMRS prin dezvoltarea mecanismului de menținere a stabilității sistemului cu buclă închisă cu regulatorul perturbat cu zgomot aleatoriu de magnitudine finită și a mecanismul explorare cu menținere a funcționării în RS. Validarea acestui algoritm se realizează pe SFE, asigurând astfel stabilitatea și funcționarea în RS în timpul explorării a sistemului cu ecuații de dinamică necunoscute, comparat cu celelalte metode din literatură care necesită cel puțin cunoașterea unui model nominal al procesului. În capitolul 8 se prezintă concluziile și direcții viitoare de cercetare.

În paragrafele următoare se sintetizează contribuțiile și ideile principale ale tezei din Capitolele 2-6, și se prezintă inovațiile aduse în domeniul ÎÎRS. În Capitolul 2 se descrie sistemul neliniar în timp discret, descris prin ecuația de stare

$$\begin{cases} \xi_{k+1} = F(\xi_k, \mathbf{u}_k), \\ \mathbf{z}_k = g(\xi_k), \end{cases} \quad (1)$$

unde $k \in \mathbb{N}$ reprezintă indicele de timp, $\xi_k = [\xi_{1,k}, \dots, \xi_{n,k}]^T \in \Omega_{\Xi} \subset \mathbb{R}^n$ vectorul de stare al sistemului, $\mathbf{u}_k = [u_{1,k}, \dots, u_{m,k}]^T \in \Omega_U \subset \mathbb{R}^m$ vectorul de intrare de control, $\mathbf{z}_k = [z_{1,k}, \dots, z_{p,k}]^T \in \Omega_Y \subset \mathbb{R}^p$ este ieșirea sistemului, $F: \Omega_{\Xi} \times \Omega_U \rightarrow \Omega_{\Xi}$ și $g: \Omega_{\Xi} \rightarrow \Omega_Y$ funcții de sistem continuu diferențiabile necunoscute ale sistemului, unde Ω_{Ξ} , Ω_U și Ω_Y reprezintă submulțimi compacte ale $\mathbb{R}^n$, $\mathbb{R}^m$ și respectiv $\mathbb{R}^p$. Intrarea de control este generată de $\mathbf{u}_k = C(\xi_k)$, unde $C: \Omega_{\Xi} \rightarrow \Omega_U$ este o funcție vectorială continuă și invariantă în timp în raport cu starea ξ . În continuarea acestui capitol se descriu algoritmi IV, ILR, precum și variațiile acestora cu funcții Q.

Algoritmul analizat în CM1 este derivat din algoritmul PDA, folosindu-se aproximatori de tip RN care sunt antrenați în mod offline. Pentru demonstrația convergenței algoritmului se generează stabilitatea Uniform Asimptotică Mărginită (UAM) a procesului de învățare, folosindu-se o abordare de tip Lyapunov. Această demonstrație extinde analizele de convergență dezvoltate anterior pentru algoritmi de tip PDA care folosesc RN cu un singur strat ca aproximatori de funcții și unde învățarea se face folosind câte o singură tranziție de stare, generalizând teoria pentru $L_{Q|C}$ straturi ascunse ale RN aferente funcției Q și a regulatorului, și pentru M tranziții de stare în baza de date.

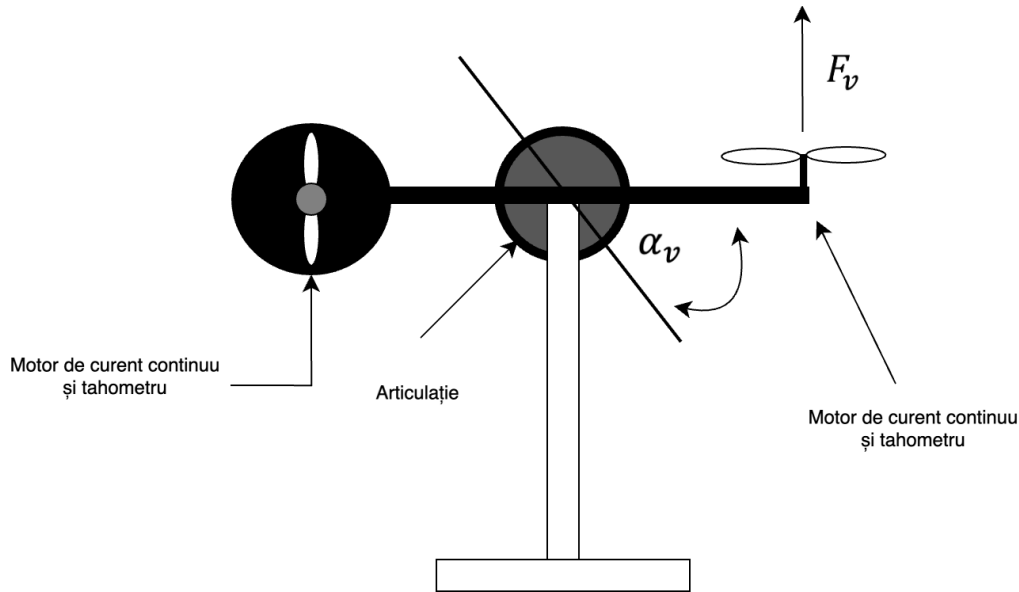
În Capitolul 3 se prezintă algoritmul PDEDA-LO ce utilizează RN antrenați în mod offline pentru aproximarea funcției de cost și a regulatorului. Pentru învățarea în loturi, se utilizează pentru învățare toate tranzițiile de stare $(\xi_k, \mathbf{u}_k, \xi_{k+1})$ stocate în baza de date $DTB_M = \{(\xi_k, \mathbf{u}_k, \xi_{k+1})\}$, cu $k = 1:M$, ce au fost colectate în timpul procesului de explorare. În Algoritmul 3.1 se descriu pașii algoritmului PDEDA-LO, ce folosește metoda căutării de gradient actualizarea ponderilor RN multistrat aferente funcției de cost stare acțiune (FCSA) și a regulatorului. Pentru demonstrația de convergență a acestui algoritm, se descrie în Secțiunea 3.2 analiza de convergență UAM, ce folosește metoda Lyapunov. Această demonstrație extinde analizele de convergență dezvoltate anterior pentru algoritmi de tip PDA care folosesc RN cu un singur strat ca aproximatori de funcții și unde învățarea se face folosind câte o singură tranziție de stare, generalizând teoria pentru $L_{Q|C}$ straturi ascunse ale RN aferente funcției Q și a regulatorului, și pentru M tranziții de stare în baza de date.

Necesitatea acestei contribuții teoretice se bazează pe creșterea în complexitate a algoritmilor de ÎÎ în ultima perioadă, care folosesc diverse arhitecturi de RN multistrat și metodologii de învățare mai complexe, bazate fie pe loturi, fie pe mini-loturi. Astfel că, existența unei teorii generale a convergenței algoritmului PDEDA este imperativă pentru garanțiile de funcționare ale algoritmului.

Pe baza a 4 Leme (Lema 3.1-3.4) ce descriu diferențele funcțiilor Lyapunov de la doi pași iterativi de învățare. Apoi se formulează Teorema principală 3.2, ce descrie convergența procesului de învățare PDEDA-LO condiționată de valoarea ratei de învățare, și anume

$$\alpha_Q < \frac{2(\bar{\theta}_{Q,max}^2 \pi_{Q,max}^2 + \sum_{l_Q=1}^{L_Q-1} \bar{\theta}_{Q,max} \pi_{Q,max}^2 \bar{\theta}_{Q,max} \prod_{l=l_Q}^{L_Q-1} \bar{\theta}_{Q,max} \pi_{Q,max}^2)}{\bar{\theta}_{Q,max}^2 \pi_{Q,max}^4 + \sum_{l_Q=1}^{L_Q-1} \bar{\theta}_{Q,max}^2 \pi_{Q,max}^4 \prod_{l=l_Q}^{L_Q-1} \bar{\theta}_{Q,max}^2 \pi_{Q,max}^2} = \alpha_{Q,max}, \quad (2)$$

$$\alpha_C < \frac{2(\bar{\theta}_{C,max} \Omega_{max} \pi_{max} + \sum_{l_C=1}^{L_C-1} \bar{\theta}_{C,max} \phi_{max} \Omega_{max} \prod_{l=l_C}^{L_C-1} \bar{\theta}_{C,max} \pi_{C,max}^2)}{\Omega_{max}^2 \pi_{max}^2 + \sum_{l_C=1}^{L_C-1} \pi_{max}^2 \Omega_{max}^2 \prod_{l=l_C}^{L_C-1} \bar{\theta}_{C,max}^2 \pi_{C,max}^2} = \alpha_{C,max}. \quad (3)$$

**Figura 1.** Sistemul SADR.

Pentru validarea superiorității RN multistrat comparat cu RN monostrat, s-a aplicat PDEDA-LO pe un scenariu de urmărire a unui model de referință (MR) pe sistemul aerodinamic cu dublu rotor (SADR), care este de tip MIMO (engl. multiple input multiple output) cu două intrări de comandă și două ieșiri, neliniar, cu 6 stări, proces descris în Fig. 1. În această validare s-a demonstrat faptul că în cazul RN multistrat convergența procesului de învățare este mai rapidă, iar performanța de reglare pe scenariul de test este totodată superioară. Această metodă a fost publicată în [14], îndeplinind CM1.

Metoda PDACGPD descrisă în Capitolul 4 are ca obiectiv principal constrângerea comenzilor generate de regulatorul acordat între două iterații de învățare a metodei PDA cu Gradientul Politicii Deterministe (PDAGPD). Pornind de la analogul acestei metode pentru regulatoare stohastice [19]-[21], acest algoritm extinde metodele descrise în [22] și [23] adăugând o constrângere ce limitează ieșirile de comandă ale regulatorului acordat între două iterații succesive de învățare într-un anumit interval definit de ϑ . Această metodă pornește de la observația practică a faptului că în timpul rulării algoritmilor de învățare de tip ÎÎ, performanța regulatorului acordat poate să varieze drastic, existând posibilitatea obținerii unui regulator destabilizant într-o singură iterație de învățare. Acest lucru se datorează mai multor factori precum erorile de aproximare inerente RN, explorării ineficiente a domeniului stărilor-comenzilor, zgomotului pe semnalele culese în timpul explorării care corup baza de date folosită pentru acordarea aproximărilor sau chiar a inițializării algoritmului cu hiper-parametrii de învățare incorecți. Astfel că, pornind de la un regulator inițial stabilizant și limitând într-un interval ϑ diferența dintre comenzile aferente regulatorului acordat într-o singură iterație de învățare, se restricționează impactul aspra sistemului condus a unui regulator destabilizant obținut la o iterație oarecare. Folosind acest algoritm, garanția de funcționare în RS este doar circumstanțială procesului de învățare cu constrângeri, neexistând niște constrângeri aferente stărilor sistemului.

Actualizarea RN aferentă regulatorului se face prin căutare de gradient cu proiecție, având următoarea formă

$$\hat{C}(\xi_k, \hat{\theta}_c^{1,j,i_c+1}, \hat{\theta}_c^{2,j,i_c+1}) = \hat{C}(\xi_k, \hat{\theta}_c^{1,j,i_c}, \hat{\theta}_c^{2,j,i_c}) - \alpha \frac{\partial \hat{Q}(\xi_k, u, \hat{\theta}_c^{1,j+1}, \hat{\theta}_c^{2,j+1})}{\partial \hat{C}(\xi_k, \hat{\theta}_c^{1,j,i_c}, \hat{\theta}_c^{2,j,i_c})} \Big|_{u=\hat{C}(\cdot, \hat{\theta}_c^{1,j,i_c}, \hat{\theta}_c^{2,j,i_c})}, \quad (4)$$

$$s. t. |\hat{C}(\xi_k, \hat{\theta}_C^{1,j+1}, \hat{\theta}_C^{2,j+1}) - \hat{C}(\xi_k, \hat{\theta}_C^{1,j}, \hat{\theta}_C^{2,j})| \leq \vartheta.$$

Diferența principală dintre PDAGPD și PDACGPD constă în modul în care semnalul de gradient a RN funcției Q este retro-propagat în RN regulatorului. Pentru algoritmul PDAGPD se calculează gradientul RN funcției Q față de intrarea de control a regulatorului, $\partial E_{C,i}^j / \partial \hat{C}(\Xi_p, \hat{\theta}_{C,i}^{1,j}, \hat{\theta}_{C,i}^{2,j})$, care este apoi multiplicată cu gradientul ieșiri regulatorului în raport cu ponderile sale, $\partial \hat{C}(\Xi_p, \hat{\theta}_{C,i}^{1,j}, \hat{\theta}_{C,i}^{2,j}) / \partial \hat{\theta}_{C,i}^{1,j}$. Pentru actualizarea regulatorului folosind PDACGPD, gradientul RN funcției Q față de intrarea de control, $\omega_k^{j,lc}$, actualizează mai întâi ținta regulatorului, care este apoi folosită pentru a actualiza ponderile RN regulatorului folosind algoritmul de retro-propagare. Astfel, gradientul funcției Q în raport cu intrarea este propagat indirect către ponderile RN a regulatorului, prin intermediul țintelor, diferind de propagarea directă a informației de gradient precum în cazul PDAGPD. Fig. 2 ilustrează această diferență arhitecturală fundamentală între cele două metode.

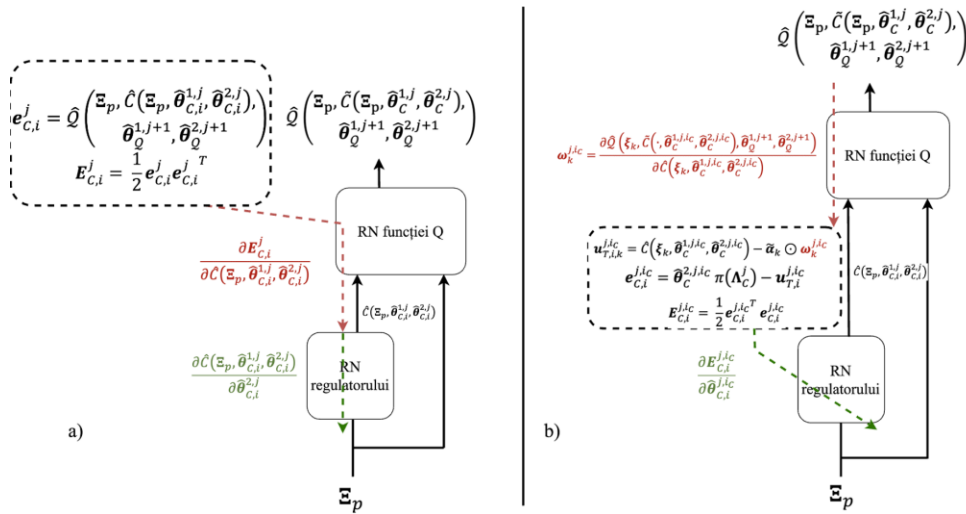


Figura 2. Diagrama de actualizare a regulatorului folosind a) PDAGPD, b) PDACGPD.

Acest algoritm s-a validat pe un scenariu de control al unghiului de înclinare al palelor unei turbine eoliene și pe un sistem de tip suspensie activă pentru un sfert de automobil. Pentru validarea mecanismului de constrângere, s-a setat diferența maximă a ieșirii regulatorului ca $\vartheta = 0.5$ și, aplicând un singur pas de învățare PDACGPD, s-a calculat diferența normată ridicată la pătrat dintre ieșirile reguletoarelor succesive, adică $\|\hat{C}(\xi_k^e, \theta_C^{1,7,0}, \theta_C^{2,7,0}) - \hat{C}(\xi_k^e, \theta_C^{1,7,1}, \theta_C^{2,7,1})\|_2^2$. Această diferență poate fi vizualizată în Fig. 3 pentru toate stările din baza de date generată în timpul procesului de explorare, pentru sistemul de tip turbină eoliană.

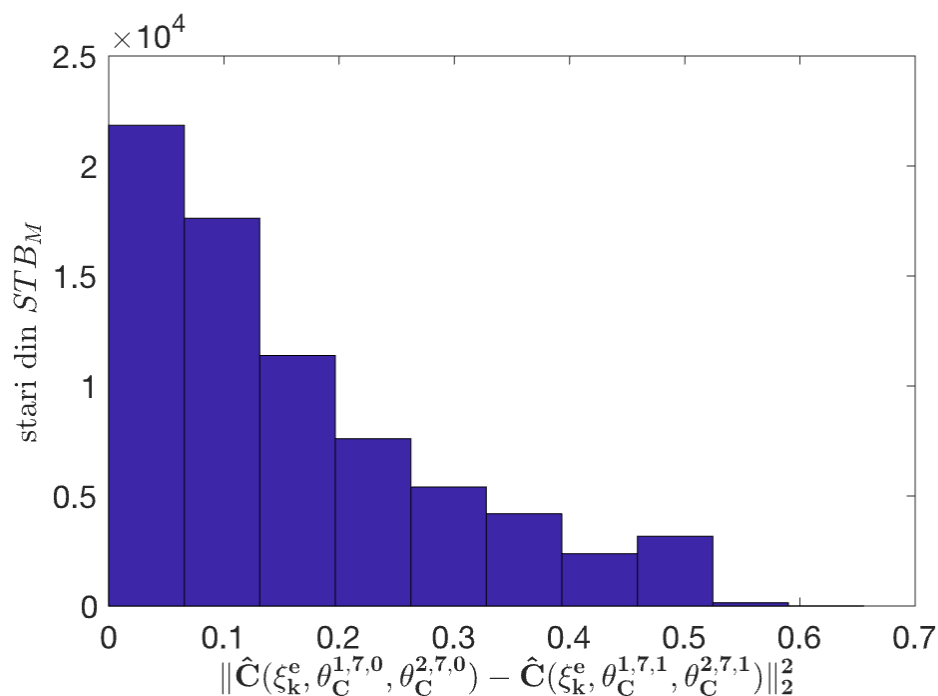


Figura 3. Histograma diferenței normalizate dintre ieșirile de comandă a două regulatoare de la iterații succesive.

Pentru sistemul de tip suspensie activă pentru un sfert de automobil, algoritmul PDACGPD a reușit să mențină diferențele normalizate dintre ieșirile de comandă a două regulatoare de la iterații de învățare succesive în limita $\vartheta = 1e^{-5}$. Se poate observa în Fig. 4 faptul că regulatorul acordat cu PDACGPD obține o diferență normală între două iterații de învățare mai mică decât cea a regulatorului acordat cu PDAGPD.

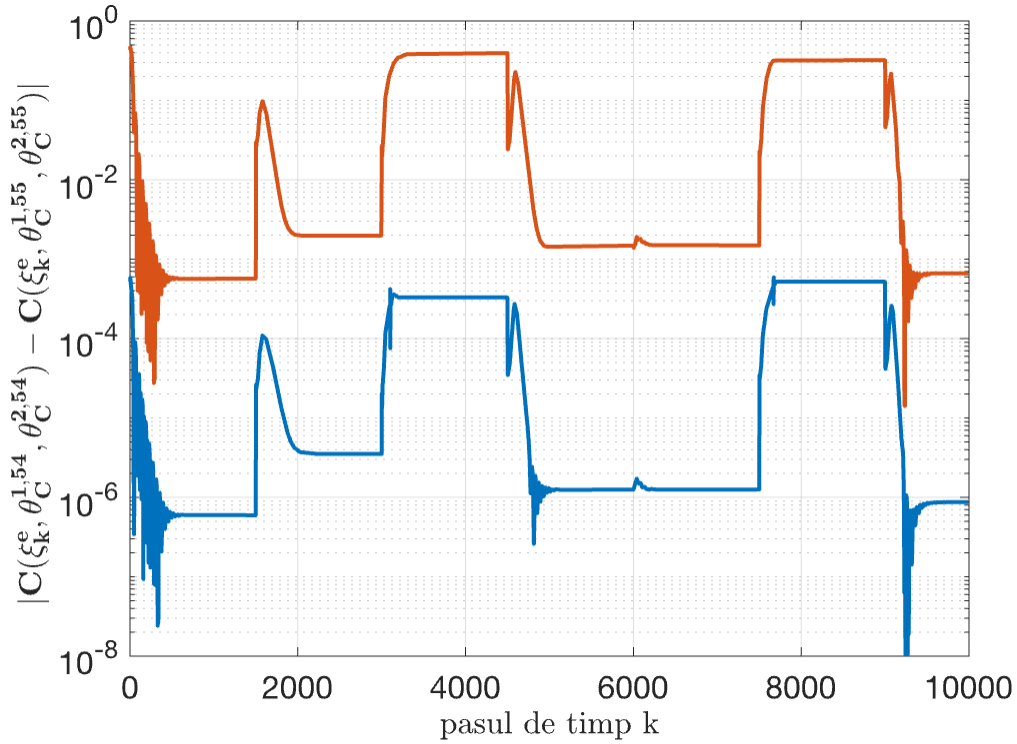


Figura 4. Diferența normată dintre regulatorul de la iterația $j = 54$ și $j = 55$ pentru algoritmul PDAGPD (portocaliu) și PDACGPD (albastru).

În Capitolul 5 s-a aplicat algoritmul PDACGPD pentru actualizarea reguletoarelor liniare generale cu parametrii invariabili în timp, fiind descrisă de ecuația

$$\mathbf{u}_k = \frac{a_1}{a_0} \mathbf{u}_{k-1} - \dots - \frac{a_{n_u}}{a_0} \mathbf{u}_{k-n_e} + \frac{b_0}{a_0} \boldsymbol{\varepsilon}_k + \frac{b_1}{a_0} \boldsymbol{\varepsilon}_{k-1} + \dots + \frac{b_{n_e}}{a_0} \boldsymbol{\varepsilon}_{k-n_e}, \quad (5)$$

unde n_u și n_e reprezintă gradele polinoamelor asociate cu $\mathbf{u}(q)$ și $\boldsymbol{\varepsilon}(q)$, cu $n_e \leq \chi + 1$ și $n_u \leq \chi$, iar $\boldsymbol{\varepsilon}_k = \mathbf{r}_k - \mathbf{z}_k$ reprezintă eroarea de urmărire, cu $\mathbf{r}_k$ semnalul de referință. Validarea s-a realizat pe un sistem neliniar de ordinul 3, demonstrând atât capacitatea acestui algoritm de a converge către un punct fix, [precum se poate observa în Fig. 5](#), cât și de a constrânge ieșirile de comandă ale unor reguletoare succesive. CM2 este dezvoltată astfel în Capitolele 4 și 5 ale tezei și respectiv în două lucrări de specialitate ale autorului, și anume [15], [16].

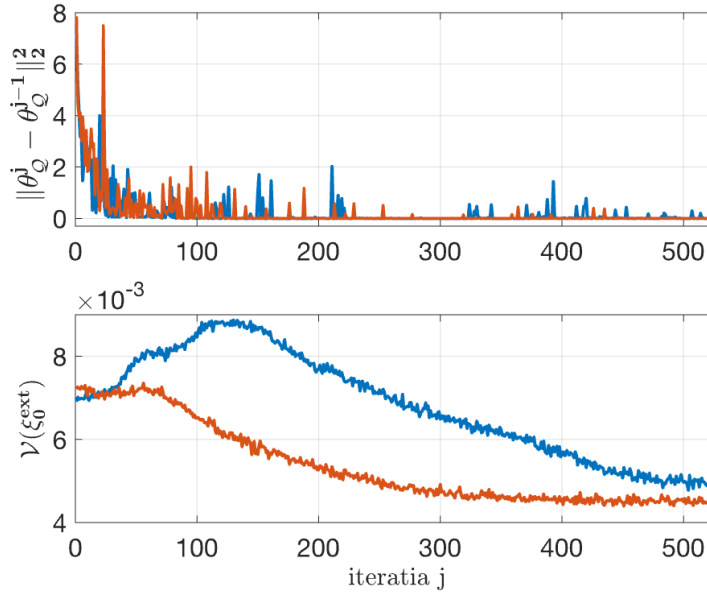


Figura 5. Convergența algoritmului PDACGPD (portocaliu) și PDAGPD (albastru).

Metoda IVFMCRS descrisă în Capitolul 6, în comparație cu metoda PDACGPD descrisă anterior, impune asupra sistemului condus cu regulatorul acordat cu IV fără model (IVFM) constrângerea de a nu tranzitiona către un subdomeniu al domeniului stărilor unde există riscul deteriorărilor fizice (mecanice, electrice ș.a.m.d.) ale sistemului condus, supranumită și regiune nesigură. Pentru definirea funcționării RS, se folosesc metode geometrice, caracterizând regiunea nesigură ca mulțimea compactă de stări $Y \subset \Omega_{\Xi}$ și cea sigură ca $S = \Omega_{\Xi} \setminus Y$. Obiectivul IVFMCRS este de a evita tranzițiile de stare în Y . Pentru acest lucru s-a definit o regiune de operare E unde regulatorul aferent iterației curente de învățare IV rulează și o regiune Π unde rulează regulatorul $C_0(\xi_k^{ext})$, definindu-se un regulator în comutație între cele două regiuni

$$u_k = \begin{cases} C_j(\xi_k^{ext}), \{\xi_k^{ext} | \xi_k \in E\}, \\ C_0(\xi_k^{ext}), \{\xi_k^{ext} | \xi_k \in \Pi\}. \end{cases} \quad (6)$$

Regiunea E reprezintă subdomeniul domeniului stărilor Ω_{Ξ} unde sistemul rulează în timpul procesului de explorare. Din cauza faptului că sistemul condus (1) în buclă închisă cu regulatorul (6) este hibrid, stabilitatea acestuia este realizată cu ajutorul metodei aferente sistemelor hibride, anume metoda funcțiilor Lyapunov multiple (FLM), unde se identifică câte o funcție Lyapunov atât pentru sistemul în buclă închisă cu regulatorul acordat cu IVFM cât și pentru sistemul în buclă închisă cu regulatorul inițial stabilizabil. Pentru acest lucru se definește o funcție Lyapunov pentru sistemul în buclă închisă cu $C_0(\xi_k^{ext})$ și anume

$$Q_{C_0}(\xi_k^{ext}, C_0(\xi_k^{ext})) = \varsigma(\xi_k^{ext}, C_0(\xi_k^{ext})) + \gamma Q_{C_0}(\xi_{k+1}^{ext}, C_0(\xi_{k+1}^{ext})), \quad (7)$$

și o funcție Lyapunov aferentă sistemului în buclă închisă cu regulatorul $C_j(\xi_k^{ext})$, și anume funcția Q de la iterația curentă

$$Q_j(\xi_k^{ext}, u_k) = \varsigma(\xi_k^{ext}, u_k) + \gamma Q_{j-1}(\xi_{k+1}^{ext}, C_j(\xi_{k+1}^{ext})). \quad (8)$$

Teorema 6.3 demonstrează că dacă valoarea funcției Lyapunov a unei regiuni de operare înainte de comutație este mai mare sau egală cu valoarea funcției Lyapunov după comutația înapoi în aceeași regiune, atunci sistemul în

comutație este stabil. Acest lucru poate fi vizualizat în Fig. 6 unde se demonstrează o posibilă evoluție a traiectoriei sistemului (1) în buclă închisă cu regulatorul (6) unde acesta este stabil. Totodată, în subcapitolul 6.3.3 s-a dezvoltat o metodă de detecție timpurie a instabilității sistemului, iar în 6.3.4 o metodă de atenuare a fenomenului de comutație rapidă inerent sistemelor hibride.

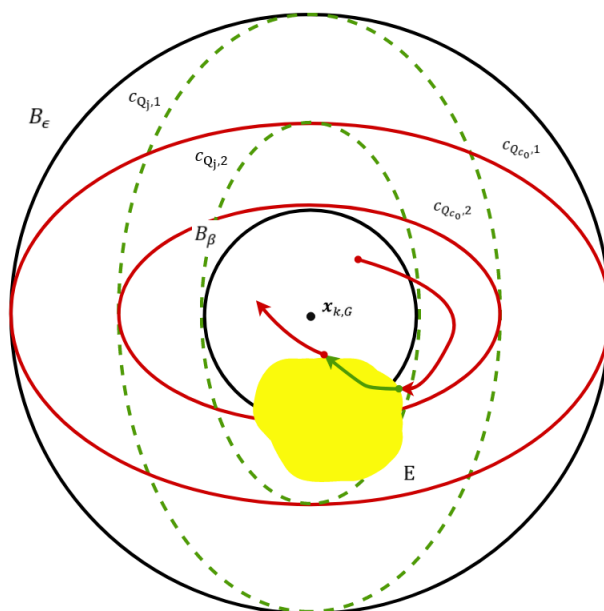


Figura 6. Exemplificarea unei evoluții a sistemului cu regulatorul (6).

Pentru implementarea practică a acestui algoritm s-au folosit RN pentru aproximarea funcțiilor (7) și (8), antrenarea acestor rețele fiind descrisă prin Algoritmii 6.2 și 6.3. Pentru implementarea regulatorului (6), s-a aproximat regiunea E cu ajutorul unei metode de estimare a tranziției maxime δ a stării pe un pas descrisă în Algoritmul 6.4, folosind doar date colectate în timpul explorării, precum este descris în Algoritmul 6.5. Pe baza acestui parametru s-au filtrat datele care au un pas de tranziție maximă mai mare decât δ . În final, s-a antrenat offline un clasificator mono-clasă de tip Mașină cu Vector Suport (MVS) pentru discriminarea regiunii E, urmând ca verificarea acestui clasificator să se realizeze online, în timpul execuției sistemului. Pentru toate aproximările s-au furnizat demonstrații de convergență, în vederea determinării efectelor erorilor de aproximare. Diagrama arhitecturii de control IVFMCRS este prezentată în Fig. 7.

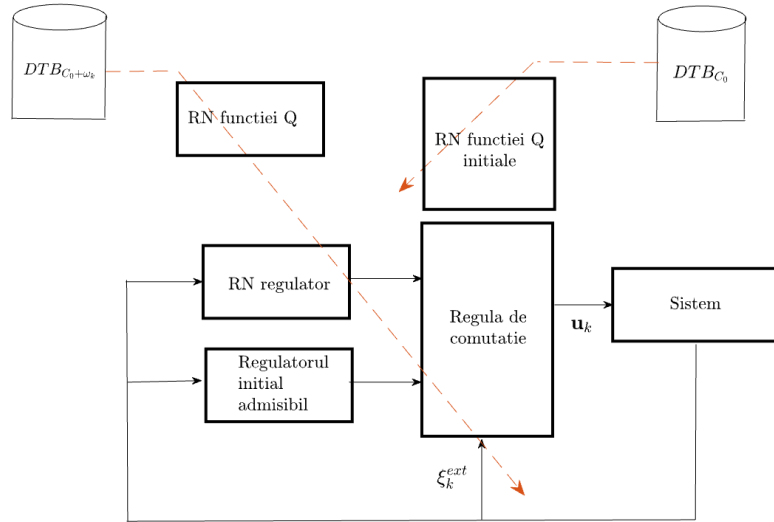


Figura 7. Arhitectura de control și învățare IVFMCRS

Validarea metodei IVFMCRS pe sistemul liniar

$$\xi_{k+1} = 0.6703\xi_k + 0.3297u_k, \quad (9)$$

și pe SFE, descris în Fig. 8, demonstrează capacitatea acestei metode de a constrânge evoluția traiectoria sistemului în regiunea de operare E, indiferent de performanța de reglare a regulatorului de la iterația de învățare curentă, $C_j(\xi_k^{ext})$. Pentru ambele tipuri de sisteme metoda garantează neîncălcarea constrângerilor de non-tranzitare în regiunile nesigure ale domeniului stărilor, în cazul existenței unui regulator destabilizant $C_{US}(\xi_k^{ext})$.

În Fig. 9 se prezintă performanța sistemului liniar de ordinul 1 în buclă închisă atât cu regulatorul $C_{US}(\xi_k^{ext})$, cât și cu regulatorul în comutație (6) care folosește $C_{US}(\xi_k^{ext})$ în loc de regulatorul $C_j(\xi_k^{ext})$. $F_{MVS}(\xi_k^{ext})$ reprezintă semnalul aferent frontierei de decizie implementată cu algoritmul MVS, σ_k reprezintă semnalul de comutație (1 – regulatorul inițial stabilizant $C_0(\xi_k^{ext})$, 0 – regulatorul destabilizant $C_{US}(\xi_k^{ext})$), r_k semnalul de referință de tip treaptă, $Q(\xi_{k+1}^{ext}, C_0(\xi_{k+1}^{ext}), \theta_{Q_{C_0}}^{50})$ funcția Q de la iterația $j = 50$. Regiunea sigură este definită ca $S = \{(\xi_k, r_k) | \xi_k \in [-\infty; 3.5], r_k \in [-\infty; 3.5]\}$, și este reprezentată de regiunea de culoare mov.

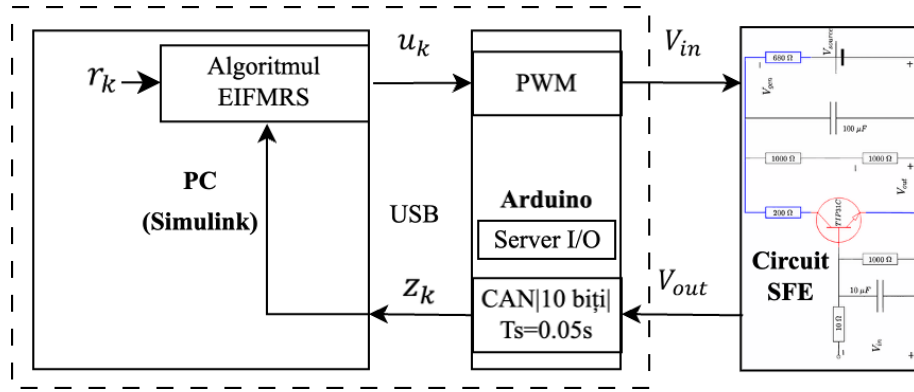


Figura 8. Diagrama SFE.

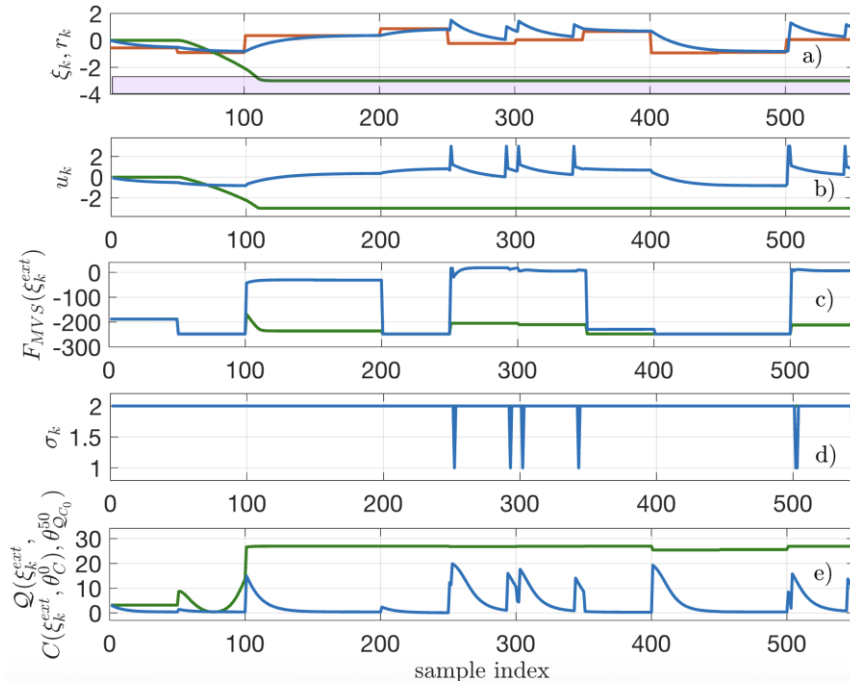


Figura 9. Performanța urmăririi referinței folosind IVFMCRS cu regulatorul destabilizant $C_{US}(\xi_k^{ext})$: a) ξ_k, r_k (roșu); b) u_k ; c) $F_{MVS}(\xi_k^{ext})$; d) σ_k ; e) $Q(\xi_k^{ext}, C_0(\xi_{k+1}^{ext}), \theta_{Q_{C_0}}^{50})$. Semnalele albastre din graficele a) – e) reprezintă semnalele interne IVFMCRS cu regulatorul $C_{US}(\xi_k^{ext})$ în buclă închisă cu procesul (9). Semnalele albastre din graficele a)–e) corespund regulatorului destabilizant $C_{US}(\xi_k^{ext})$ folosind legea de control (6) în buclă închisă cu procesul (9).

Pentru SFE, obținem în timpul învățării un regulator destabilizant la iterația $j = 180$, $C(\xi_k^{ext}, \pi_c^{180})$. Regiunea sigură este definită ca $S = \{(z_k, r_k) | z_k \in [-\infty; 6.5], r_k \in [-\infty; 6.5]\}$, și este reprezentată de regiunea de culoare mov. Folosind metoda exponențial stabilizantă relaxată $\psi(\varepsilon_0)\lambda^k + \tau$, se demonstrează performanța algoritmului IVFMCRS de a menține sistemul în buclă închisă cu regulatorul în comutație (6) cu utilizarea timpului de staționare în Fig. 10.

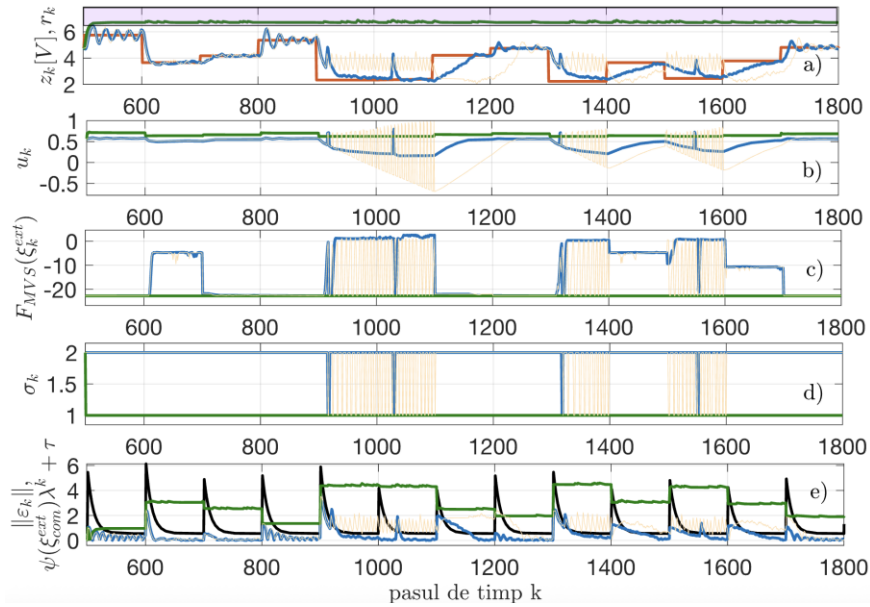


Figura 10. Performanța de urmărire a referinței a metodei IVFMCRS pentru regulatorul destabilizant $C(\xi_k^{ext}, \pi_c^{180})$: a) z_k, r_k (roșu); b) u_k ; c) $F_{MVS}(\xi_k^{ext})$; d) σ_k ; e) semnalul de referință ESR $\psi(\epsilon_0)\lambda^k + \tau$ (negru). Semnalele albastre din graficele a)–e) corespund urmăririi referinței cu algoritmul IVFMCRS folosind regula (6) cu regulatorul destabilizant $C(\xi_k^{ext}, \theta_c^{180})$ împreună cu timpul de staționare în buclă închisă cu SFE. Semnalele verzi din graficele a)–e) corespund regulatorului destabilizant $C(\xi_k^{ext}, \theta_c^{180})$ în buclă închisă cu SFE. Semnalele galbene din graficele a)–e) corespund urmăririi referinței cu algoritmul IVFMCRS folosind regula (6) cu regulatorul destabilizant $C(\xi_k^{ext}, \theta_c^{180})$ în buclă închisă cu procesul SFE.

Cele două aplicații validează astfel mecanismul de garantare a funcționării în RS și în cazul existenței unui regulator destabilizant. Această metodă reprezintă subiectul CM3, fiind prezentată în lucrarea științifică [17].

Atât metoda PDACGPD cât și IVFMCRS abordează problema menținerii stabilității și a siguranței doar în cadrul testării reguletoarelor acordate, bazându-se pe ipoteza existenței unei baze de date pentru implementarea procesului de învățare. Funcționarea în RS nu este garantată în cadrul procesului de explorare din cauza impredictibilității generate de zgomotul aditiv pe comanda generată de regulatorul inițial stabilizant. Astfel că, în ultima parte a tezei se abordează problema menținerii funcționării în regimul de siguranță în timpul procesului de explorare, dezvoltându-se algoritmul Explorare Iterativă fără model în regim de siguranță (EIFMRS). CM4 este descrisă în lucrarea trimisă spre publicare [18].

Contribuțiile descrise anterior au fost publicate în jurnale și conferințe indexate WOS, prezentate sumar prin următoarea listă:

- [14] T. Lala, "Constrained Adaptive Dynamic Programming for PID Controllers," in *Proc. IEEE 19th Int. Conf. Control Autom. (ICCA)*, Tallinn, Estonia, Jun. 30-Jul. 3, 2025, pp. 1-6.
- [15] T. Lala, "Stability Analysis of Batch Offline Action-Dependent Heuristic Dynamic Programming Using Deep Neural Networks," *Mathematics*, vol. 13, no. 2, Jan. 2025.
- [16] T. Lala, "Safe Switching Model-Free Value Iteration for General Nonlinear Systems," *IEEE Access*, vol. 13, pp. 132170-132193, May 2025.
- [17] T. Lala, "Constrained deterministic policy gradient adaptive dynamic programming using projected gradient update," *Syst. Sci. Control Eng.*, vol. 13, no. 1, art. 2546823, 2025.

- [18] T. Lala, I. Silea, “Iterative Model Free Safe Exploration using Event-triggered Impulsive Perturbations”, **acceptat la congresul IFAC, Busan, Korea, 23-28 August 2026**, <https://www.ifac2026.org/fairDash.do>.

Referințe

- [1] D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel, T. Lillicrap, K. Simonyan, and D. Hassabis, "A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play," *Science*, vol. 362, no. 6419, pp. 1140-1144, Dec. 2018.
- [2] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, et al., "Mastering the game of go without human knowledge," *Nature*, vol. 550, pp. 354-359, 2017.
- [3] V. Mnih, K. Kavukcuoglu, D. Silver, et al., "Human-level control through deep reinforcement learning," *Nature*, vol. 518, pp. 529-533, 2015.
- [4] M. M. Afsar, T. Cremonesi, and P. Turrin, "Reinforcement learning based recommender systems: A survey," *ACM Comput. Surv.*, vol. 55, no. 7, pp. 1-38, Dec. 2022.
- [5] L. Ouyang et al., "Training language models to follow instructions with human feedback," in *Advances in Neural Information Processing Systems*, vol. 35, pp. 27730-27744, 2022.
- [6] J. Achiam, D. Held, A. Tamar, P. Abbeel. Constrained policy optimization," in *Proc. of Int. Conf. on Mach. Learn.*, Sydney, Australia, 2017, pp. 22–31.
- [7] C. Tessler, D. J. Mankowitz, and S. Mannor, "Reward Constrained Policy Optimization," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2019.
- [8] R. Cheng, G. Orosz, R. M. Murray, and J. W. Burdick, "End-to-end safe reinforcement learning through barrier functions for safety-critical continuous control tasks," in *Proc. AAAI Conf. on Artif. Int.*, Honolulu, Hawaii, USA, feb. 2019, pp. 3387–3395.
- [9] M. Alshiekh, R. Bloem, R. Udiger Ehlers, B. Konighofer, S. Niekum, U. Topcu, "Safe Reinforcement Learning via Shielding," in *Proc. of the AAAI Conf on Artif. Int.*, Palo Alto, USA, 2018, pp. 1-6.
- [10] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, P. Tabuada, "Control Barrier Functions: Theory and Applications," in *Proc. Europ. Contr. Conf.*, Piscataway, NJ, 2019, pp. 3420-3431.
- [11] I. Mitchell, A. Bayen, and C. Tomlin, "A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games," *IEEE Trans. Autom. Control*, vol. 50, no. 7, pp. 947-957, 2005.
- [12] F. Berkenkamp, M. Turchetta, A. P. Schoellig and A. Krause, "Safe Model-based Reinforcement Learning with Stability Guarantees," in *Proc. Neur. Inf. Proc. Syst.*, Long Beach, USA, Dec. 2017, pp 908-919.
- [13] Y. Chow, O. Nachum, E. Duenez-Guzman and M. Ghavamzadeh, "A Lyapunov-based Approach to Safe Reinforcement Learning," in *Proc. Neur. Inf. Proc. Syst.*, Montreal, Canada, Dec. 2018.
- [14] T. Lala, "Constrained Adaptive Dynamic Programming for PID Controllers," in *Proc. IEEE 19th Int. Conf. Control Autom. (ICCA)*, Tallinn, Estonia, Jun. 30-Jul. 3, 2025, pp. 1-6.
- [15] T. Lala, "Stability Analysis of Batch Offline Action-Dependent Heuristic Dynamic Programming Using Deep Neural Networks," *Mathematics*, vol. 13, no. 2, Jan. 2025.
- [16] T. Lala, "Safe Switching Model-Free Value Iteration for General Nonlinear Systems," *IEEE Access*, vol. 13, pp. 132170-132193, May 2025.

- [17] T. Lala, "Constrained deterministic policy gradient adaptive dynamic programming using projected gradient update," *Syst. Sci. Control Eng.*, vol. 13, no. 1, art. 2546823, 2025.
- [18] T. Lala, I. Silea, "Iterative Model Free Safe Exploration using Event-triggered Impulsive Perturbations", **acceptat la congresul IFAC, Busan, Korea, 23-28 August 2026**, <https://www.ifac2026.org/fairDash.do>.
- [19] J. Schulman, P. Moritz, M. Jordan, P. Abbeel, "Trust Region Policy Optimization," in *Proc. Int. Conf. on Learn. Repr.*, San Diego, USA, 2015, pp. 1-6.
- [20] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," ArXiv, abs/1707.06347, 2017.
- [21] J. Achiam, D. Held, A. Tamar, P. Abbeel. Constrained policy optimization," in *Proc. of Int. Conf. on Mach. Learn.*, Sydney, Australia, 2017, pp. 22–31.
- [22] B. Luo, D. Liu, H. Wu, D. Wang, and F. L. Lewis, "Policy gradient adaptive dynamic programming for data-based optimal control," *IEEE Trans. Cybern.*, vol. 47, no. 10, pp. 3341-3354, 2017.
- [23] Y. Zhang, B. Zhao, and D. Liu, "Deterministic policy gradient adaptive dynamic programming for model-free optimal control," *Neurocomputing*, vol. 387, pp. 40-50, 2020.