Raport științific

privind implementarea proiectului în perioada ian- decembrie 2019

Contract TE 65/2018

Robust fractional order event-based control for optimized resource allocation in complex cyber-physical closed loop systems **Rezumat:** În cadrul Etapei 2 de implementare a grantului de cercetare TE 65/2018 s-au realizat activități de dezvoltare a noilor algoritmi, așa cum s-a propus în propunerea de proiect.

În prima parte a acestui raport se va prezenta un algoritm existent pentru determinarea parametrilor unui regulator fracționar de tip PI/PID pentru procesele cu timp mort, acesta fiind bazat pe o aproximare serie de ordin întâi a timpului mort. Acest algoritm a fost dezvoltat de către membri echipei de cercetare anterior proiectului TE65/2018. Tot în cadrul primei activități, se va prezenta algoritmul dezvoltat în cadrul proiectului pentru regulatoare discrete de tip FO-PID pe baza extinderii metodologiei IMC clasice pentru procesele cu timp mort. Acest algoritm are la baza o aproximare nouă a timpului mort, aproximare dezvoltată tot de către membri ai echipei de cercetare. Acordarea regulatorului se bazează pe specificarea unui set de marimi de performanță în domeniul frecvențial (margine de fază și frecvență de tăiere). În cadrul Activității 2.2 s-a testat și validat algoritmul de control dezvoltat în activitatea anterioara pentru diferite procese cu timp mort (sisteme WSAN) și anume: două sisteme de ordin întâi și un sistem de ordin superior cu timp mort.

Următoarea activitate a presupus proiectarea unui algoritm de tip PID bazat pe evenimente. Pașii necesari pentru implementarea unui regulator de tip PID sub forma unui algoritm bazat pe evenimente sunt descriși în aceasta parte a raportului științific. În cadrul activității 2.4 s-a testat și validat algoritmul de control bazat pe evenimente din activitatea anterioară pentru diferite procese cu timp mort (sisteme WSAN) și anume: două sisteme de ordin întâi și un sistem de ordin superior cu timp mort (aceleași procese ca și în cazul anterior).

În cadrul activității 2.5 s-a dezvoltat o strategie de control bazată pe evenimente ce are la bază regulatorul de ordin fracționar calculat prin extinderea metodologiei IMC clasice pentru procesele cu timp mort (Activitatea 2.1). Abordarea constă în implementarea regulatorului fracționar ca și un PI/PID clasic (bazat pe evenimente) împreună cu un filtru de ordin fracționar pentru filtrarea semnalului de eroare. În cadrul activității 2.6 s-a testat și validat algoritmul de control bazat pe evenimente de ordin fracționar din activitatea anterioară pentru diferite procese cu timp mort (sisteme WSAN) și anume: două sisteme de ordin întâi și un sistem de ordin superior cu timp mort (aceleași procese ca și în cazul anterior). În finalul acestui raport se prezintă succint lucrările realizate și finanțate din cadrul proiectului.

<u>Activitatea 2.1. Dezvoltarea unui algoritm specific pentru regulatoare discrete FO-PID pe</u> <u>baza extinderii metodologiei IMC clasice pentru procesele cu timp mort</u>

O primă variantă de acordare a regulatoarelor folosind metodologia IMC a fost propusă de către membri echipei de cercetare (Muresan et al., 2016) și se bazează pe determinarea parametrilor unui regulator PID de ordin fracționar pe criteriul echivalenței dintre structurile clasice de reglare cu reacție negativă și structurile de reglare de tip IMC. În această abordare inițială, timpul mort al procesului este aproximat în serie (ordinul întâi). Structura de reglare cu FO-IMC este redată în Fig. 1, unde $H_{FO-IMC}(s)$ este regulatorul IMC de ordin fracționar, $H_p(s)$ este procesul, $H_m(s)$ este modelul, $H_c(s)$ este regulatorul echivalent unei bucle clasice de reglare, d-perturbație, r- semnal de referință, y- ieșirea procesului.



Fig. 1. Structura de reglare FO-IMC

Acordarea parametrilor unui regulator FO-IMC pentru procese de ordin întâi cu timp mort

Următoarea funcție de transfer descrie dinamica unui proces de ordin întâi cu timp mort:

$$H_{p}(s) = \frac{k}{Ts+1}e^{-\tau s}$$
⁽¹⁾

Presupunând o aproximare serie de ordin întâi pentru timpul mort, $e^{-\tau s} \cong 1 - \tau s$, modelul procesului descris de (1) devine: $H_m(s) = \frac{k(1-\tau s)}{Ts+1}$. În acest caz, regulatorul FO-IMC este dat de:

$$H_{\text{FO-IMC}}(s) = \frac{Ts+1}{k} \frac{1}{\lambda s^{\alpha} + 1}$$
(2)

unde $\alpha \in [0 \div 2]$ este ordinul fracționar. Regulatorul echivalent se calculează astfel:

$$H_{c}(s) = \frac{H_{FO-IMC}(s)}{1 - H_{FO-IMC}(s)H_{m}(s)} = \frac{Ts+1}{k(\lambda s^{\alpha} + \tau s)}$$
(3)

Funcția de transfer în buclă deschisă cu regulatorul echivalent din (3) și procesul descris de (1) se obtțne ca fiind:

$$H_{ol}(s) = H_c(s)H_p(s) = \frac{1}{\lambda s^{\alpha} + \tau s}e^{-\tau s}$$
(4)

În comparație cu regulatorul clasic IMC, regulatorul IMC de ordin fracționar (2) are doi parametri de acordare, constanta de timp λ și ordinul fracționar α . În cazul reglării clasice IMC, constanta de timp λ este utilizată pentru a specifica un anumit timp de răspuns. În cazul unui regulator FO-IMC, două criterii de performanță pot fi impuse, cum ar fi spre exemplu timpul de răspuns și suprereglajul. Aceste două criterii de performanță pot fi specificate în domeniul frecvențial prin impunerea unei anumite frecvențe de taiere, ω_c , și a unei anumite margini de fază, PM, pentru funcția de transfer în buclă deschisă (4). Condiția matematică pentru marginea de fază este data de:

$$\angle H_{ol}(j\omega_{c}) = -\pi + PM$$
(5)

sau într-o formă detaliată:

$$a \tan\left(\frac{\lambda \sin\left(\frac{\alpha \pi}{2}\right) + \tau \omega_{c}^{1-\alpha}}{\lambda \cos\left(\frac{\alpha \pi}{2}\right)}\right) = \pi - PM - \tau \omega_{c}$$
(6)

Dacă se aplică funcția tangentă la (6), rezultă:

$$\lambda = -\frac{\tau \omega_{c}^{1-\alpha}}{\sin\left(\frac{\alpha\pi}{2}\right) - \tan\left(\pi - PM - \tau \omega_{c}\right)\cos\left(\frac{\alpha\pi}{2}\right)}$$
(7)

Condiția matematică pentru fracvența de tăiere constă în impunerea unui criteriu pentru modulul funcției de transfer în buclă deschisă:

$$\left|\mathbf{H}_{ol}(\mathbf{j}\boldsymbol{\omega}_{c})\right| = 1 \tag{8}$$

obținându-se de aici:

$$\lambda^2 \omega_c^{2\alpha} + \tau^2 \omega_c^2 + 2\lambda \tau \omega_c^{\alpha+1} \sin\left(\frac{\alpha \pi}{2}\right) = 1$$
(9)

Determinarea parametrilor FO-IMC se bazează pe impunerea unui set de performanțe ce se referă la (ω_c , PM) și rezolvarea sistemului de ecuatți rezultat (7,9) pentru a calcula parametri FO-

IMC (λ , α). Sistemul de ecuatii este complex si neliniar, astfel că pentru determinarea solutiei sunt necesare fie tehnici de optimizare, fie metode grafice.

Acordarea parametrilor unui regulator FO-IMC pentru procese de ordin doi cu timp mort

Următoarea funcție de transfer descrie dinamica unui proces de ordin 2 cu timp mort:

$$H_{p}(s) = \frac{k}{as^{2} + bs + c}e^{-\tau s}$$
(10)

Regulatorul FO-IMC este dat de:

$$H_{FO-IMC}(s) = \frac{as^2 + bs + c}{k} \frac{1}{\lambda s^{\alpha} + 1}$$
(11)

în timp ce regulatorul echivalent, obținut prin aproximarea serie a timpul mort, este:

$$H_{c}(s) = \frac{H_{FO-IMC}(s)}{1 - H_{FO-IMC}(s)H_{m}(s)} = \frac{as^{2} + bs + c}{k(\lambda s^{\alpha} + \tau s)}$$
(12)

În acest caz, modelul a fost aproximat astfel: $H_m(s) = \frac{k(1-\tau s)}{as^2 + bs + c}$. Funcția de transfer în buclă

deschisă este data de:

$$H_{ol}(s) = H_{c}(s)H_{p}(s) = \frac{1}{\lambda s^{\alpha} + \tau s}e^{-\tau s}$$
 (13)

Se constată că și în acest caz funcția de transfer în buclă închisă este identică celei obținute pentru procesele de ordin intâi cu timp mort. De aici se deduce ca același algoritm pentru determinarea parametrilor FO-IMC poate fi utilizat. Într-o buclă de reglare clasică, regulatoarele rezultate (3) și (12) pot fi implementate ca și regulatoare clasice de tip PI/PID cu filtre de ordin fractionar:

$$H_{FO}(s) = \frac{1}{\lambda s^{\alpha - 1} + \tau}$$
(14)

Metoda nouă de aproximare: abordarea Non-Rational Transfer Function (NRTF)

Metoda NRTF de aproximare a fost inițial propusă pentru sisteme de ordin fracționar (De Keyser et al., 2018), însă poate fi utilizată cu success pentru orice tip de funcții de transfer, inclusiv de ordin fractionar cu timp mort (Muresan et al., 2019). Metoda are la bază 4 pasi, detaliați mai jos.

Pasul 1: Se calculează o aproximare discrete a sistemului fractionar, înlocuind operatorul Laplace s cu următoarea funcție:

$$w(z^{-1}) = \frac{1+\delta}{T_s} \frac{1-z^{-1}}{1+\delta z^{-1}}$$
(15)

unde $\delta \in [0\div1]$ și T_s este perioada de eșantionare. Parametrul δ este folosit pentru corectarea erorii de aproximare (De keyser et al., 2018): o valoare mare δ va diminua eroarea de aproximare a fazei sistemului, în timp ce o valoare mică va reduce eroarea de aproximare a modulului. Acest prim pas duce la obținerea unui sistem discret de ordin fracționar $G(z^{-1})$. Aproximarea FO-IMC va fi realizată într-o anumită gamă de frecvențe, $\omega \in (0, \omega_h)$, astfel că tot la acest pas trebuie definită și frecvența maximă ω_h , ținând cont de criteriile de eșantionare Nyquist cu $\omega_h = \frac{\pi}{T_c}$.

Pasul 2: Se calculează răspunsul frecvențial al sistemului obținut la pasul 1 prin înlocuirea

operatorului discret z cu
$$e^{j\omega T_s}$$
, unde $\omega = \frac{2\pi}{T_s N_s} \begin{bmatrix} 0 & 1 & 2 & \dots & \frac{N_s}{2} \end{bmatrix}$. Parametrul N_s poate fi

utilizat tot în reducerea erorii. Cu cât valoarea acestuia este mai mare, cu atât aproximarea va fi mai eficientă. Rezultatul acestui pas constă într-un vector ce conține răspunsul frecvențial al sistemului discret de ordin fracționar.

<u>Pasul 3</u>: Se calculează răspunsul pondere al sistemului discret de ordin fracționar, pe baza algoritmului Fast Fourier Transform (FFT). Acest pas duce la obținerea unui vector de dimensiune N_s conținând răspunsul pondere:

$$g[n] = \frac{1}{N_s} \sum_{k=0}^{N_s - 1} G[k] e^{+j \frac{2\pi}{N_s nk}} n = 0, 1, 2, ..., N_s - 1$$
(16)

<u>Pasul 4</u>: Se determină o funcție de transfer rațională care să genereze un răspuns pondere similar celui obținut la pasul 3. Metoda Steiglitz-McBride este utilizată pentru a determina aceasta funcție de transfer. Ordinul N al funcției trebuie să fie definit. Acest pas duce la obținerea unei funcții de transfer discrete, de ordin întreg, care aproximează dinamica celui fracționar, inițial:

$$G(z^{-1}) = \frac{c_0 + c_1 z^{-1} + \dots + c_N z^{-N}}{d_0 + d_1 z^{-1} + \dots + d_N z^{-N}}$$
(17)

Metoda propusă pentru acordarea FO-IMC pentru sisteme de ordin întâi cu timp mort, pe baza aproximării NRTF

Pentru procesul descris de (1), regulatorul FO-IMC este dat de:

$$H_{\text{FO-IMC}}(s) = \frac{Ts+1}{k} \frac{1}{\lambda s^{\alpha} + 1}$$
(18)

iar regulatorul echivalent o

$$H_{c}(s) = \frac{H_{FO-IMC}(s)}{1 - H_{FO-IMC}(s)H_{m}(s)}$$
(19)
unde modelul este defin

$$H_{ol}(s) = H_{c}(s)H_{p}(s) = \frac{1}{2}$$
(20)

Similar cazului anterior, în care s-a utilizat aproximarea serie, și în acest caz, două performanțe pot fi impuse. Pentru a satisface cerințele legate de frecvența de tăiere și marginea de fază, ecuațiile de faza și modul din (5) și (8) sunt utilizate și aici pentru a determina parametri regulatorului FO-IMC. Pe baza condiției de fază (5) rezultă:

$$a \tan \left(\frac{\lambda \omega_{c}^{\alpha} \sin\left(\frac{\alpha \pi}{2}\right) + \sin\left(\tau \omega_{c}\right)}{\lambda \omega_{c}^{\alpha} \cos\left(\frac{\alpha \pi}{2}\right) + 1 - \cos\left(\tau \omega_{c}\right)} \right) = \pi - PM - \tau \omega_{c}$$
(21)

ceea ce duce la:

din (20), rezultă: (23)

Acordarea FO-IMC se bazează pe specificarea celor două mărimi de performanță (ω_c , PM) și rezolvarea sistemului de ecuații rezultat (22,23) pentru a determina (λ , α), cu λ >0. La o comparație între ecuațiile (22)/(23), respectiv (7)/(9), este evident faptul că pentru același set de performanțe, parametri FO-IMC vor fi diferiți. Simulările realizate, precum și datele experimentale, incluse în acest raport vor demonstra beneficiile utilizării metodei propuse în comparație cu cea existentă.

Metoda propusă pentru acordarea FO-IMC pentru sisteme de ordin doi cu timp mort, pe baza aproximării NRTF

Pentru procesul de ordin doi cu timp mort din (10), regulatorul FO-IMC este:

(22)

$$H_{FO-IMC}(s)$$

iar regulatorul echivalent va fi dat de:

$$H_{c}(s) = \frac{H_{FO-IMC}(s)}{1 - H_{FO-IMC}(s)H_{m}(s)}$$
(25)

fiind calculat considerând modelul procesulul dat de $H_m(s) = \frac{k}{as^2 + bs + c}e^{-\tau s}$. Funcția de transfer în

buclă deschisă este:

$$H_{ol}(s) = H_{c}(s)H_{p}(s) =$$
(26)

Aceiași metodă de acordare va fi utilizată și în acest caz. În cazul unei bucle de reglare clasice, regulatoarele FO-IMC propuse în acest raport vor fi implementate prin structurile (19), respectiv (25), aproximate în variantă discretă folosind metoda NRTF.

Referințe

- Muresan, C.I., Dutta, A., Dulf, E.H., Pinar, Z., Maxim, A., Ionescu, C.M. (2016), Tuning algorithms for fractional order internal model controllers for time delay processes, International Journal of Control, Vol. 89, No. 3, pp. 579-593
- De Keyser, R., Muresan, C.I., Ionescu, C.M. (2018), An efficient algorithm for low-order discrete-time implementation of fractional order transfer functions, ISA Transactions, vol. 74, pp. 229-238, DOI: 10.1016/j.isatra.2018.01.026
- Muresan, C.I., Birs, I., Prodan, O., Nascu, I., De Keyser, R. (2019), Approximation Methods for FO-IMC Controllers for Time Delay Systems, The 2nd International Conference on Electrical Engineering and Green Energy, E3S Web of Conferences, vol. 115, DOI:1051/e3sconf/201911501003, Rome, Italy, June 28-30

Activitatea 2.2. Testarea și validarea algoritmului de control dezvoltat în activitatea anterioară pentru diferite procese cu timp mort (sisteme WSAN)

Pentru testarea algoritmului realizat s-au considerat diverse tipuri de procese cu timp mort. Pentru simplitate, se redau aici, două cazuri particulare, precum și validarea experimentală pe un sistem de tip vertical take off and landing (VTOL).

Testarea algoritmului FO-IMC propus pe un sistem de ordin întâi (T>>τ)

(24)

Se consideră un sistem de ordin întâi cu timp mort dat de:

$$H_{p}(s) = \frac{1}{4s+1}e^{-s}$$
(27)

Pentru proiectarea unui regulator FO-IMC pe baza metodei propuse, se impun o margine de fază PM =85° și o frecvență de tăiere ω_c =0.3 rad/s. Soluția sistemului de ecuații (22)-(23) duce

Pentru implementarea regulatorului din (29), metoda NRTF este folosită cu următorii parametri: N=7, δ =1 și ω_h =10 π (T_s=0.1s). Având în vedere că rejectarea perturbațiilor constituie o problemă principală în cazul regulatoarelor de tip IMC, s-au realizat teste de acest tip pentru validarea algoritmului. Fig. 2 și Fig. 3 prezintă rezultatele simulării în buclă închisă.

Fig. 2. Semnalul de ieșire pentru un sistem de	Fig. 3. Semnalul de intrare pentru un sistem de
ordın întâi (T≫t)	ordın întâi (T>>τ)

Testarea algoritmului FO-IMC propus pe un sistem de ordin întâi (T<<τ)

Se consideră un sistem de ordin întâi cu timp mort dat de:

$$H_{p}(s) = \frac{2}{s+1}e^{-2s}$$
(30)

Pentru proiectarea unui regulator FO-IMC pe baza metodei propuse, se impun o margine de fază PM =80° și o frecvență de tăiere ω_c =0.3 rad/s. Soluția sistemului de ecuații (22)-(23) duce

Pentru implementarea regulatorului din (32), metoda NRTF este folosită cu următorii parametri: N=5, δ =0.5 și ω_h =10 π (T_s=0.1s). Având în vedere că rejectarea perturbațiilor constituie o problemă principală în cazul regulatoarelor de tip IMC, s-au realizat teste de acest tip pentru validarea algoritmului. Fig. 4 și Fig. 5 prezintă rezultatele simulării în buclă închisă.

Fig. 4. Semnalul de ieșire pentru un sistem de ordin întâi (T<<τ)	Fig. 5. Semnalul de intrare pentru un sistem de ordin întâi (T<<τ)

Testarea algoritmului FO-IMC propus pe un sistem de ordin superior cu timp mort. Studiu de caz pe un echipament de tip vertical take off and landing (VTOL)

Platforma experimentală VTOL folosită pentru teste este redată în Fig. 6. Dinamica acestui sistem a fost aproximată la o funcție de transfer de ordin doi cu timp mort (Birs et al., 2018):

$$H_{\rm VTOL}(s) = \frac{22.24}{s^2 + 0.6934s + 5.244} e^{-0.8s}$$
(33)

Pentru a determina parametri FO-IMC, următoarele criterii de performanță sunt impuse:



Fig. 6. Platforma VTOL folosită pentru validarea experimentală a algoritmului FO-IMC propus

Regulatorul din (35) este implementat folosind metoda NRTF cu N=5, δ =0.9 și ω_h =200 π (T_s=0.005s). Un filtru pentru diminuarea zgomotelor este adăugat, având constanta de timp T_f=0.1s. Relația de recurență pentru semnalul de comandă, rezultată în urma discretizării este:

$$c(k) = 5.79*c(k-1)-13.99*c(k-2)+18.04*c(k-3)-13.1*c(k-4)+5.1*c(k-5)-0.84*c(k-6)+ \\ +0.008*c(k-7)+0.53*e(k)-3.2*e(k-1)+7.96*e(k-2)-10.65*e(k-3)+8.15*e(k-4)- \\ -3.45*e(k-5)+0.69*e(k-6)-0.037*e(k-7)$$
(36)

Rezultatele experimentale, atât în ceea ce privește urmărirea referinței, cât și rejectarea perturbațiilor, sunt redate în Fig. 7 și 8, pentru semnalul de ieșire și cel de control.

Referințe

 Birs, I.R Muresan, C.I., Nascu, I., Folea, S., Ionescu, C. (2018), Experimental results of fractional order PI controller designed for second order plus dead time (SOPDT) processes, Proceedings of the 2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV), DOI: 10.1109/ICARCV.2018.8581241, 18-21 November 2018, Singapore 2. Muresan, C.I., Birs, I.R., De Keyser, R. (2019), An Improved Design Approach for Fractional Order Internal Model Controllers for Vertical Take-Off and Landing Platforms, Mechatronics, under review



Activitatea 2.3. Proiectarea unui algoritm clasic de tip PID bazat pe evenimente

Strategia de control PID bazat pe evenimente este o metodă de optimizare a alocării și utilizării resurselor (Birs et al., 2020a). Acordarea efectivă a regulatorului PID (sau derivate ale acestuia) se face independent de implementarea bazată pe evenimente, orice metodă de calcul ai parametrilor fiind acceptată. Pricipiul de funcționare al unui algoritm bazat pe evenimente este redat în Fig. 9 și constă din două componente principale: un detector de evenimente (event detector) și un generator al semnalului de control (control input generator) (Birs et al., 2020b). Așa cum se poate observa din Fig. 9, ieșirea procesului este măsurată la fiecare perioada de eșantionare h_{nom}. Această perioadă de eșantionare are aceiași semnificație fizică cu perioada de eșantionare folosită în teoria standard a regulatoarelor discrete și trebuie să satisfacă cerințele legate de teorema lui Shannon. Fiecare esantion măsurat este apoi preluat de către detectorul de evenimente. Acesta decide dacă generatorul semnalului de control trebuie sau nu să calculeze o nouă valoare a semnalului de control care să fie mai apoi transmis procesului. Valoarea în secunde dintre două calculări succesive ale semnalului de control este hact, aceasta fiind resetată după fiecare calcul. Se definește și o valoare maximă între două calcule succesive ca fiind h_{max} , ales ca multiplu de h_{nom}. Alegerea valorilor pentru h_{nom}, h_{act} și h_{max} cade în sarcina celui ce proiecteaza algoritmul.

Semnalul de eroare la momentul de timp t este definit ca diferența dintre semnalul de referință și ieșirea măsurata a procesului:

$e(t) = Y_{sp}(t) - Y(t)$	(37)
$I(t) = I_{sp}(t) I(t)$	(57)

Valoarea absolută a diferenței dintre eroarea curentă e și cea anterioară es va genera un eveniment dacă aceasta este în afara unui interval predefinit $[-\Delta_e, \Delta_e]$:

$$|\mathbf{e} - \mathbf{e}_{\mathrm{s}}| \ge \Delta_{\mathrm{e}} \tag{38}$$

sau dacă o condiție de siguranță apare, dată de:

 $h_{act} \ge h_{max}$

(39)

Pentru fiecare eșantion măsurat al ieșirii procesului, detectorul de evenimente va verifica dacă una din condițiile (38) sau (39) este validă. Va declanșa în acest caz generatorul semnalului de control care va calcula o nouă valoare a comenzii pe baza unui algoritm predefinit.



Fig. 9. Principiul de funcționare al unui algoritm bazat pe evenimente

Algoritmul PID bazat pe evenimente (Sanchez et al., 2012) se bazează pe forma standard a unui regulator:

$$C(s) = k_p \left(1 + \frac{1}{T_i s} + \frac{T_d s}{1 + \frac{T_d s}{N_d s}} \right)$$

$$\tag{40}$$

având semnalul de control:

$$U(s) = k_p \left(\beta Y_{sp}(s) - Y(s) + \frac{1}{T_i s} \left(Y_{sp}(s) - Y(s) \right) + \frac{T_d s}{1 + \frac{T_d s}{N_d}} \left(\gamma Y_{sp}(s) - Y(s) \right) \right)$$
(41)

Semnalul de control din (41) poate fi rescris pe componentele principale, proporțional u_p , integrator u_i și derivativ u_d :

$$u_{p}(s) = k_{p} \left(\beta Y_{sp}(s) - Y(s)\right)$$

$$u_{i}(s) = \frac{k_{p}}{T_{is}} \left(Y_{sp}(s) - Y(s)\right)$$

$$u_{d}(s) = \frac{T_{d}s}{1 + \frac{T_{d}}{N_{d}}s} \left(\gamma Y_{sp}(s) - Y(s)\right)$$
(42)

Implementarea bazată pe evenimente se bazează pe forma discretă a semnalelor din (42) însumate:

$$u_{p}(k) = k_{p} \left(\beta Y_{sp}(k) - Y(k)\right)$$

$$u_{d}(s) = \frac{T_{d}s}{1 + \frac{T_{d}}{N_{d}}s} \left(\gamma Y_{sp}(s) - Y(s)\right)$$
(43)

unde Y_{sp}(k) și Y(k) sunt semnalele de referință și de ieșire la momentul de timp curent. Pentru componenta integratoare se utilizează $s = \frac{1-z^{-1}}{h_{act}}$, obținându-se:

$$u_{i}(k) = u_{i}(k-1) + \frac{k_{p}}{T_{i}}h_{act}\left(Y_{sp}(k) - Y(k)\right)$$
(44)

Componenta derivativă se obține utilizând $s = \frac{1-z^{-1}}{z^{-1}h_{act}}$:

$$u_d(k) = \frac{T_d u_d(k-1)}{N_d h_{act} + T_d} + \frac{k_p T_d N_d}{N_d h_{act} + T_d} (E(k) - E(k-1))$$
(45)

Unde E(k) este eroare ponderată la momentul curent $E(k) = \gamma Y_{sp}(k) - Y(k)$, iar E(k-1) este eroare ponderată la momentul trecut, $E(k-1) = \gamma Y_{sp}(k-1) - Y(k-1)$.

Forma finală discretă a semnalului de control va fi:

$$U(k) = u_p(k) + u_i(k) + u_d(k)$$
(46)

14

Implementarea algoritmului PID bazat pe evenimente este dat de pseudocodul din Fig. 10 (Birs et al., 2020). Parametrii de ponderare ai referinței din implementarea algoritmului au fost aleși, $\beta = 1$ și $\gamma = 0$ pentru simplitatea demonstrării conceptelor regulatoarelor bazate pe evenimente. Algoritmul din Fig.10 este implementat utilizând mediul Simulink. Schema bloc a structurii de reglare cu reacție negativă cu PID bazat pe evenimente este prezentată în Fig. 11. Întreaga logică din spatele regulatorului PID bazat pe evenimente este în interiorul blocului *Matlab function* care reprezintă o funcție scrisă în Matlab utilizând limbajul bazat pe C și nu limbajul grafic specific Simulink-ului.

Data: Setpoint and current process variable **Result:** New command signal Main initialize h_{nom} , h_{max} , Δ_e , β , k_p , T_i , T_d , N_d , y_{sp} $h_{act} = 0$ $e_s = 0$ $u_i = 0$ while control ongoing do $e = y_{sp} - y$ $\begin{array}{l} h_{act} = h_{act} + h_{nom} \\ \text{if } |e - e_s| \ge \Delta_e \ OR \ h_{act} \ge h_{max} \ \text{then} \\ | \begin{array}{c} u = \\ control_input_generator(y_{sp}, y, \beta, k_p, T_i, T_d) \end{array} \end{array}$ end end return def control_input_generator($y_{sp}, y, \beta, k_p, T_i, T_d$): $\begin{aligned} u_p &= \kappa_p (\beta y_{sp} - y) \\ u_d &= T_d / (N_d h_{act} + T_d) u_d - k_p T_d N_d / (N_d h_{act} + T_d) (y - y_{old}) \\ u &= u_p + u_i + u_d \\ u_i &= u_i + k_p / T_i h_{act} (y_{sp} - y) \\ y_{old} &= y \\ h_{act} &= 0 \end{aligned}$ $e_s = e$

return u

Fig. 10. Implementarea algoritmului PID bazat pe evenimente

Blocul Matlab function încorporează logica din spatele detectorului de evenimente și a generatorului de semnale. Blocurile din partea de jos a figurii, hact, hnom, etc. sunt objecte de tip Memory Block, utilizate pentru a salva valorile unor variable între iterații, necesare în calculul parametrilor regulatorului. Aceasta este singura posibilitate de comunicare între schemele Simulink și codul sursă din Matlab funcțion. Parametrul hact reprezintă perioada de timp dintre momentul actual și momentul când a fost calculată ultima valoare a comenzii, inițializat cu 0. Parametrii ud, ui, u last oferă valorile comenzilor generate de componentele derivativă și integratoare, cât și valoarea totală a comenzii la ultima interație, inițializate cu 0. Blocurile e, es si v old retin valori ale erorii la iteratia trecută, erorii la iteratia la care s-a calculat ultimul semnal de comandă și valoarea iesirii la momentul când s-a calculat ultima valoare de comandă, initializate cu 0. Parametrul hnom este o variabilă globală care retine perioada de esantionare a algoritmului. Această valoare trebuie setată și ca parametru de configurare al programului Simulink. Valoarea hmax este variabila globală care reprezintă timpul maxim permis între două calcule ale valorii comenzii, hmax > hnom. Delta e reprezintă valoarea de prag pentru declanșarea unui eveniment. Așadar, pentru fiecare proces, se vor alege valori potrivite pentru hmax. Delta e si hnom.



Fig. 11. Schema bloc – structură de control cu regulator PID bazat pe evenimente În continuare se va prezenta codul din spatele functiei *Matlab function*.

```
function u = event_detector(ysp, y)
%% Parametrii PID

global hact hnom Delta_e hmax es e u_last
e = ysp - y;
hact = hact + hnom;

if (abs(e - es) > Delta_e || hact >= hmax)
u = control_input_generator(ysp, y, beta, Kp, Ti, Td, Nd);
u_last = u;
else
u = u_last;
end
end
```

La fiecare iterație a codului Simulink, funcția *event_detector* calculează valoarea erorii și verifică eligibilitatea calculării unei comenzi noi. În caz favorabil, se va apela funcția *control_input_generator* definită mai jos.

```
function u = control_input_generator(ysp, y, beta, Kp, Ti, Td, Nd)
global hact ud y_old ui es e
up = Kp * (beta*ysp - y);
ud = Td / (Nd*hact + Td)*ud - Kp*Td*Nd/(Nd*hact+Td)*(y-y_old);
u = up + ui + ud;
ui = ui + Kp/Ti*hact*(ysp-y);
y_old = y;
hact = 0;
es = e;
end
```

Referințe:

 Birs, C. Muresan, An event based implementation of a fractional order controller on a scalable nanorobot, The European Control Conference, St. Petersburg, Russia, 12-15 May 2020, submitted Birs, C. Muresan, I. Nascu, R. de Keyser, An event-based implementation of the KC autotuner for vertical take-off and landing control, The European Control Conference, St. Petersburg, Russia, 12-15 May 2020, submitted

Pentru validarea și testarea algoritmului bazat pe evenimente, descris în cadrul Activității 2.3, s-au folosit diferite procese cu timp mort. Pentru simplitate se prezinta mai jos 3 dintre aceste procese, fiind considerate exemplificative.

Pentru fiecare proces au fost efectuate trei teste care vizează urmărirea referinței, rejectarea perturbației pe semnalul de ieșire și robustețe la schimbări ale factorului de amplificare al procesului. Cele trei scenarii de test sunt printre cele mai reprezentative situații întâlnite în practică.

Testarea algoritmului de control bazat pe evenimente pe un sistem de ordin întâi (T>>\tau)

Se consideră în acest caz un sistem de ordin întâi cu timp mort dat de:

$$H_{p}(s) = \frac{1}{4s+1}e^{-s}$$
(37)

procesul fiind identic cu cel din (27). Pentru proiectarea unui regulator PI se utilizează tehnici standard de acordare în domeniul frecvențial, impunându-se aceleași performanțe ca în cazul regulatorului FO-IMC (Activitatea 2.2): margine de fază PM =85° și o frecvență de tăiere ω_c =0.3 rad/s. În acest caz, ecuațiile de fază și de modul conduc la soluția: k_p=1.3841 și T_i=6.3715, funcția de transfer a regulatorului PI fiind dată de:

$$C_{PI}(s) = 1.3841 \left(1 + \frac{1}{6.3715s} \right) \tag{38}$$

Implementarea regulatorului din (38) în varianta bazată pe evenimente se face folosind algoritmul prezentat în Activitatea 2.3.

Pentru procesul dat în (37) se aleg hnom = 0.1, hmax = 1 și Delta_e = 0.1. Valoarea hnom este aleasă astfel încât să respecte teorema lui Shannon, hmax este ales de 10 ori mai mare, iar Delta_e = 0.1 este ales arbitrar, valoarea fiind aceeași pentru fiecare exemplu.

Fig. 12 prezintă răspunsului sistemului în buclă închisă cu regulator PID bazat pe evenimente la o intrare treaptă unitară. Răspunsul ajunge în regim staționar cu un timp de răspuns de 25

secunde atât pentru sistemul compensat cât și cel necompensat. Pe graficul de comandă se pot observa intervalele de timp de lungimi neregulate la care s-a calculat valoarea comenzii.



Fig.12. Răspunsului sistemului în buclă închisă cu regulator PID bazat pe evenimente la o intrare treaptă unitară



Fig.13. Rejectarea perturbației pe ieșire

Fig. 13 prezintă capabilitățile de rejectare a perturbației pe ieșire. La momentul t=50s se introduce o perturbație care reprezintă 20% din valoarea semnalului de referință pe ieșirea procesului. Perturbația este anulată la momentul t=70. După cum se poate vedea, în ambele cazuri răspunsul se intoarce la valoarea de referință cu un timp de răspuns de aprox. 5s.

Robustețea este testată în Fig.14, unde același regulator din (38) este implementat pentru un proces cu un factor de amplificare crescut cu 50%. Și în acest caz, regulatorul duce ieșirea procesului la valoarea de referință unitară.



Fig.14. Verificarea robusteții la variații ale factorului de amplificare

Regulatorul PI bazat pe evenimente este validat cu succes în toate cele 3 cazuri.

Testarea algoritmului de control bazat pe evenimente pe un sistem de ordin întâi ($T \ll \tau$)

Se consideră un sistem de ordin întâi cu timp mort dat de:

$$H_{p}(s) = \frac{2}{s+1}e^{-2s}$$
(39)

procesul fiind identic cu cel din (30). Pentru proiectarea unui regulator PI se utilizează tehnici standard de acordare în domeniul frecvențial, impunându-se aceleași performanțe ca în cazul regulatorului FO-IMC (Activitatea 2.2): margine de fază PM =80° și o frecvență de tăiere ω_c =0.3

rad/s. În acest caz, ecuațiile de fază și de modul conduc la soluția: $k_p=0.3430$ și T_i=2.9055, funcția de transfer a regulatorului PI fiind dată de:

$$C_{PI}(s) = 0.3430 \left(1 + \frac{1}{2.9055s} \right) \tag{40}$$

Implementarea regulatorului din (40) în varianta bazată pe evenimente se face folosind algoritmul prezentat în Activitatea 2.3.

Pentru implementarea regulatorului din (40) se aleg următorii parametrii hnom = 0.1, hmax = 1 și delta_e = 0.1. Motivația alegerii este aceeași ca în cazul anterior.

Regulatorul PI bazat pe evenimente este testat din punct de vederea al erorii staționare în Fig. 15, unde urmărește cu succes referința treaptă unitară cu un timp de răspuns de 15s. Se poate observa că procesul necompensat se stabilizează la 2. Și în acest caz, se poate vedea cum comanda este calculată la intervale neregulate de timp.



Fig.15. Răspunsului sistemului în buclă închisă cu regulator PID bazat pe evenimente la o intrare treaptă unitară

Rejectarea perurbației este prezentată în Fig. 16. Aceeași perturbație ca în cazul anterior, având valoarea 0.2 este introdusă la momentul t=50s. Regulatorul rejectează perturbația cu un timp de răspuns de aprox. 10 secunde. Robustețea este testă în Fig. 17, unde se poate vedea suprareglajul provocat de o schimbare de 50% a factorului de amplificare. Cu toate că suprareglajul și timpul de răspuns sunt crescute față de cazul nominal, regulatorul bazat pe evenimente stabilizează sistemul, demonstrându-și robustețea, deși nu a fost acordat în acest sens.



Fig.16. Rejectarea perturbației pe ieșire



Fig.17. Verificarea robusteții la variații ale factorului de amplificare

Testarea algoritmului de control bazat pe evenimente pe un sistem de ordin superior cu timp mort. Studiu de caz pe un echipament de tip vertical take off and landing (VTOL)

Platforma experimentală VTOL folosită a fost descrisă anterior în Activitatea 2.3, dinamica procesului fiind aproximată la o funcție de transfer de ordin doi cu timp mort:

$$H_{\rm VTOL}(s) = \frac{22.24}{s^2 + 0.6934s + 5.244} e^{-0.8s}$$
(41)

procesul fiind identic cu cel din (33). Pentru proiectarea unui regulator PI se utilizează tehnici standard de acordare în domeniul frecvențial, impunându-se aceleași performanțe ca în cazul regulatorului FO-IMC (Activitatea 2.2): margine de fază PM =75° și o frecvență de tăiere ω_c =0.5 rad/s. În acest caz, ecuațiile de fază și de modul conduc la soluția: k_p=0.0458 și T_i=0.4178, funcția de transfer a regulatorului PI fiind dată de:

$$C_{PI}(s) = 0.0458 \left(1 + \frac{1}{0.4178s} \right) \tag{42}$$

Implementarea regulatorului din (42) în varianta bazată pe evenimente se face folosind algoritmul prezentat în Activitatea 2.3.

Procesul VTOL este un proces fizic care permite achiziția datelor cu o perioadă de eșantionare de 0.005s. Din acest motiv, se alege hnom=0.005. Valoarea hmax = 0.1 se alege ca fiind de 20 ori mai mare decât hnom, iar Delta_e = 0.1 este similar cu celelalte exemple.

Urmărirea referinței este prezentată în Fig. 18. După cum se poate vedea, eroarea staționară este nulă, iar timpul de răspuns este aprox. 15s. De asemenea, suprareglajul este mult redus. Implementarea practică a regulatorului PI bazat pe evenimente este imperativă pentru optimizarea resurselor necesare platformei VTOL, utilizarea CPU poate fi redusă cu 80% față de cazul implementării unui regulator clasic.

Perturbația este rejectată cu succes în Fig. 19. O perturbație reprezentând 20% din valoarea treptei unitare dată ca referință este introdusă pe ieșirea sistemului la t=50s. Rejectarea totală a perturbației este realizată cu un timp de răspuns de 20s. La momentul t=90s, perturbația este eliminată de pe ieșirea sistemului, iar ieșirea revine la valoarea de referință tot cu un timp de răspuns de 20s.

Robustețea este testată în Fig. 20. Testul surprinde o variație de 20% a factorului de amplificare, de la 22.24 la 26.68. Testul este imperativ pentru procesul fizic deoarece în cazul platformelor de aterizare și decolare verticală (Vertical Take-Off and Landing), greutatea obiectelor transportate variază, motivând necesitatea unui caracter robust al sistemului în buclă închisă. Fig. 20 arată un suprareglaj și un timp de răspuns crescute al procesului alterat. Cu toate acestea, referința ajunge la valoarea de referință cu o eroare staționară nulă.



Fig. 18. Răspunsului sistemului în buclă închisă cu regulator PID bazat pe evenimente la o intrare treaptă unitară



Fig.19. Rejectarea perturbației pe ieșire



Fig. 20. Verificarea robusteții la variații ale factorului de amplificare

Activitatea 2.5. Proiectarea unui algoritm de control bazat pe evenimente pornind de la regulatoarele FOPID dezvoltate în cadrul activității 2.1.

Pentru implementarea bazată pe evenimente a regulatoarelor FO-IMC se utilizeaza forma regulatoarelor echivalente, determinate în Activitatea 2.1. Astfel, în cazul unei *aproximări serie de ordin întâi* pentru timpul mort, regulatoarele echivalente de ordin fracționar obținute sunt date în (3) și respectiv (12), pentru procese de ordin întâi/doi:

$H_{c}(s) = \frac{H_{FO-IMC}(s)}{1 - H_{FO-IMC}(s)H_{m}(s)} =$	(43)
$H_{c}(s) = \frac{H_{FO-IMC}(s)}{1 - H_{FO-IMC}(s)H_{m}(s)}$	(44)

Într-o buclă de reglare clasică, regulatoarele rezultate (43) și (44) pot fi implementate ca și regulatoare clasice de tip PI/PID cu filtre de ordin fracționar, după cum urmează:

$$C_{PI}(s) =$$
 (45)

$$C_{PID}(s) = \frac{as^2 + bs + c}{ks} \text{ cu filtrul } H_{FO}(s) =$$
(46)

În cazul unei <u>aproximări NRTF</u> pentru timpul mort, regulatoarele echivalente de ordin fracționar sunt date în (19) și respectiv (25), pentru procese de ordin întâi/doi:

$$H_{c}(s) = \frac{H_{FO-IMC}(s)}{1 - H_{FO-IMC}(s)H_{m}(s)} =$$

$$H_{c}(s) = \frac{H_{FO-IMC}(s)}{1 - H_{FO-IMC}(s)H_{m}(s)} =$$
(47)
(48)

Într-o buclă de reglare clasică, regulatoarele rezultate (47) și (48) pot fi implementate ca și regulatoare clasice de tin PL/PLD cu filtre de ordin fracționar, după cum urmează:

$$H_c(s) = \tag{49}$$

$$H_c(s) =$$
(50)

unde în ambele cazuri filtrul de ordin fracționar este ana cazura este chema în buclă închisă este redată în Fig. 21, unde implementarea componentelor de ordin întreg PI (49) sau PID (50) se realizează pe baza algoritmului de control bazat pe evenimente (notat ca event-based PID control), iar filtrul de ordin fracționar va fi implementat ca un element de pre-filtrare a erorii, executându-se cu o perioadă de eșantionare constantă.



Fig. 21. Schema de reglare pentru FO-IMC ca structură bazata pe evenimente

Algoritmul de implementare a regulatoarelor fracționare bazate pe evenimente conform schemei din Fig. 21 suferă modificări fată de algoritmul ilustrat în Fig. 10.

Blocul *Discrete FO filter* reprezintă implementarea filtrului discretizat cu perioada de eșantionare *hnom*. Așadar, pentru fiecare test, valoarea *hnom* se alege egală cu perioada de eșantionare a filtrului fracționar.

Blocul *Discrete PID* reprezintă implementarea regulatorului PID discret bazat pe evenimente (similar cu Fig. 10). Blocurile *Trasfer Function* și *Transport Delay* reprezintă procesul cu timp mort.

Dificultatea de implementare a acestei structuri de reglare constă în implementarea regulatorului PID bazat pe evenimente. Diferența între acesta și cazul unei reglări utilizând un regulator de ordin întreg este dată de semnalul de intrare. Ecuațiile din Activitatea 2.4 sunt construite pentru o structură de reglare clasică, cu reacție negativă, unde intrarea în blocul PID este dată de semnalul de eroare (diferența între referință și ieșire). După cum se poate vedea în Fig. 21, în cazul implementării structurii de reglare folosind filtru fracționar, intrarea în blocul PID bazat pe evenimente este dată de semnalul de comandă generat de filtru.

În consecință, pentru un semnal de comandă generat de filtru, e_{fk} , componentele comenzii proporționale și derivative vor suferi modificări. Comanda proporțională devine $u_p(k) = k_p e_{fk}$, iar cea integratoare $u_i(k) = u_i(k-1) + \frac{k_p}{T_i}h_{act}e_{fk}$. Comanda derivativă este dată de $u_d(k) = \frac{T_d u_d(k-1)}{N_d h_{act}+T_d} + \frac{k_p T_d N_d}{N_d h_{act}+T_d} (e_{fk} - e_{fk}(k-1))$, unde $e_{fk}(k-1)$ este semnalul generat de filtru la momentul trecut.

Diagrama bloc a implementării Simulink este prezentată în Fig. 22.

Fig. 22. Implementarea Simulink a regulatoare FO bazate pe evenimente

Blocul *Matlab Function – FO filter* este implementarea relației de recurență a filtrului. S-a ales această implementare pentru a simula condițile de implementare practică. Codul sursă din spatele blocului este descris mai jos.



Blocul *Matlab function – Event based PID* are nevoie de trei intrări: semnalul de comandă generat de filtrul fracționar, notat cu *c*, și de semnalele de referință (ysp) și de ieșire (y). Semnalele ysp și y sunt folosite doar pentru condiția de declanșare a evenimentelor, bazate pe răspunsul procesului. O alternativă este de a declanșa evenimentele în funcție de variații ale semnalului dat de filtru. Dezavantajul acestei abordări este necesitatea de scalare a valorii de prag Delta_e în funcție de comenzi generate de fiecare filtru, în diverse situații. Alegerea verificării condiției de generarea a evenimentelor bazate pe valorile ieșirii procesului este mai avantajoasă și creează o corelație directă între regulatorul PID bazat pe evenimente și răspunsul sistemului. Codul sursă *Matlab function – Event based PID* este ilustrat în funcția *event_detector*

```
function u = event_detector(c, ysp, y)
%% Parametrii PID
%% Parametrii PID
%%
global hact hnom Delta_e hmax es e u_last
e = ysp - y;
hact = hact + hnom;

if (abs(e - es) > Delta_e || hact >= hmax)
u = control_input_generator(c, Kp, Ti, Td, Nd);
u_last = u;
else
u = u_last;
end
end
```

iar funcția control_input_generator pentru regulatoare fracționare este

```
function u = control_input_generator(c, Kp, Ti, Td, Nd)
global hact ud y_old ui es e
up = Kp * c;
ud = Td / (Nd*hact + Td)*ud + Kp*Td*Nd/(Nd*hact+Td)*(c-c_old);
u = up + ui + ud;
ui = ui + Kp/Ti*hact*c;
c_old = c;
hact = 0;
es = e;
end
```

<u>Activitatea 2.6. Testarea și validarea algoritmului de control dezvoltat în activitatea</u> <u>anterioară pentru diferite procese cu timp mort (sisteme WSAN)</u>

Pentru ilustrarea algoritmilor fracționari bazați pe evenimente s-au efectuate aceleași teste precum în Activitatea 2.4: urmărirea referinței, rejectarea perturbației și robustețea. Fiecare test prezintă trei aspecte: răspunsul sistemului în buclă închisă, comanda calculată de filtrul fracționar și comanda dată de regulatorul PID bazat pe evenimente. Pentru fiecare proces se aleg aceeași parametri precum în exemplele anterioare: hnom, hact și Delta_e.

Testarea algoritmului de control FO-IMC bazat pe evenimente pe un sistem de ordin întâi $(T >> \tau)$

Funcția de transfer utilizată este identică celei din activitățile anterioare:

$$H_{p}(s) = \frac{1}{4s+1}e^{-s}$$
(51)

Regulatorul FO-IMC a fost obținut pe baza algoritmului de control descris în Activitatea 2.1, performanțele impuse (PM=85° și ω_c =0.3 rad/s) fiind cele specificate în Activitatea 2.2, rezultând

parametri:
$$\lambda_{FO-IMC-NRTF}(s) =$$

Regulatorul echivalent (obținut în Activitea 2.2) ce urmează a fi implementat este dat de:

$$H_{c_FO_NRTF}(s) =$$
(52)

Regulatorul din (52) va fi implementat ca un regulator întreg PI (bazat pe evenimente) și un filtru de ordin fracționar, așa cum a fost specificat în Activitatea 2.5, ecuația (49):

 $H_c(s)$

Reprezentarea schematică a buclei închise rezultate este redată în Fig. 21. Ecuația generală a unui regulator PID bazat pe evenimente (descrisă și în cadrul Activității 2.3) este dată de:

$$U(s) = k_p \left(\beta Y_{sp}(s) - Y(s) + \frac{1}{T_i s} \left(Y_{sp}(s) - Y(s) \right) + \frac{T_d s}{1 + \frac{T_d s}{N_d}} \left(\gamma Y_{sp}(s) - Y(s) \right) \right)$$
(54)

În acest caz particular, regulatorul este de tip PI, astfel că $T_d=0$ în ecuația (54). Componenta de regulare tip PI din (53) este dată de:

$$H_c(s) =$$
(55)

unde $E_f(s)$ este transformata Laplace a semnalului de eroare filtrat prin filtrul de ordin fracționar $e_f(t)$ din Fig. 11, iar $k_p=4$ si $T_i=4$. În acest caz, ecuația regulatorului PI în forma prezentată pe caz general în (54) cu $\beta=1$ va fi:

$$U(s) = K_p\left(E_f(s) + \frac{1}{T_i s} E_f(s)\right)$$
(56)

Filtrul de ordin fracționar va fi implementat în forma sa discretă, iar acesta nu se va executa în forma bazată pe evenimente:

$$H_{FO}(s) =$$
(57)

ceea ce duce la următoarea relație de recurență:

 $e_f(k) =$



calculată folosind metoda NRTF cu N=7, δ =1 și ω_h =10 π (T_s=0.1 s).

Parametrii de implementare bazată pe evenimente sunt hnom = 0.1, hmax = 0.5 și delta_e = 0.1. Valorile hnom și Ts trebuie să fie egale pentru funcționarea corectă a filtrului fracționar.

Fig 23. Prezintă răspunsul la o referință treaptă unitară. Răspunsul procesului în buclă închisă este mai rapid decât răspunsul procesului necompensat. În graficul care ilustrează comanda

(53)

calculată de filtru se poate vedea că aceasta este calculată la intervale regulate de timp hnom = 0.1, iar graficul comenzii date de regulatorul bazat pe evenimente prezintă perioade de timp inegale între două valori diferite ale comenzii.



Fig. 23. Răspunsului sistemului în buclă închisă cu filtru fracționar și regulator PID bazat pe evenimente la o intrare treaptă unitară

Rejectarea perturbației este prezentată în Fig. 24. Se poate vedea eroarea staționară nulă și timpul de răspuns de aprox. 10s. De asemenea, graficele de comandă arată eficiența regulatorului PID bazat pe evenimente din punct de vedere al optimizării resurselor necesare pentrul calculul comenzii. Caracterul robust este verificat în testul prezentat în Fig. 25. Pentru o variație de 50% a factorului de amplificare, răspunsul procesului atinge valoarea referinței în 10s.

Pe baza testelor prezentare, se poate concluziona eficacitatea strategiei de reglare propuse pentru procesele de ordinul I de acest tip.



Fig. 24. Rejectarea perturbației pe ieșire



Fig. 25. Verificarea robusteții la variații ale factorului de amplificare

Testarea algoritmului de control FO-IMC bazat pe evenimente pe un sistem de ordin întâi $(T \leq \tau)$

Funcția de transfer a procesului este dată de aceiași ecuație ca în activitățile anterioare:

$$H_{p}(s) = \frac{2}{s+1}e^{-2s}$$
(59)

Regulatorul FO-IMC a fost obținut pe baza algoritmului de control descris în Activitatea 2.1, performanțele impuse (PM=80° și ω_c =0.3 rad/s) fiind cele specificate în Activitatea 2.2, rezultând parametri: λ =0.8826 și α =0.6165, funcția de transfer fiind: $H_{FO-IMC-NRTF}(s)$

Regulatorul echivalent (obținut în Activitea 2.2) ce urmează a fi implementat este dat de:

$$H_{c_NRTF}(s) =$$
(60)

Regulatorul din (60) va fi implementat ca un regulator întreg PI (bazat pe evenimente) și un filtru de ordin fracționar, așa cum a fost specificat în Activitatea 2.5, ecuația (49):

$$H_c(s) = \tag{61}$$

Reprezentarea schematică a buclei închise rezultate este redată în Fig. 21, ecuația generală folosită în implementarea regulatorului PI fiind identică celei prezentate anterior în (54), luând $T_d=0$ în ecuația (62). Componenta de reglare tip PI din (61) este dată de:

$$H_c(s) = \tag{62}$$

unde $E_f(s)$ este transformata Laplace a semnalului de eroare filtrat prin filtrul de ordin fracționar $e_f(t)$ din Fig. 11, iar $k_p=0.5$ și $T_i=1$. Și în acest caz, ecuația regulatorului PI în forma prezentată pe caz general în (54) cu $\beta=1$ va fi identică celei din (56). Filtrul de ordin fracționar va fi implementat în forma sa discretă, iar acesta nu se va executa în forma bazată pe evenimente:

$$H_{FO}(s) = \frac{(s)}{(s)}$$

ceea ce duce la următoarea relație de recurență:

 $e_{f}(k) =$

calculată folosind metoda NRTF cu N=5, δ =0.5 și ω_h =10 π (T_s=0.1 s).

(63)

Detaliile de implementare vizează alegerea timpului $hnom = T_s = 0.1s$. Perioada maximă între calculul comenzii bazată pe evenimente este ales de 5 ori mai mare, hact = 0.5. Delta_e = 0.1 este ales precum în toate celelalte exemple.

Fig. 26 arată o urmărire a referinței cu un timp de răspuns de 7s, suprareglaj nul și eroare staționară nulă.

Răspunsul procesului la perturbații este ilustrat în Fig. 27. La momentul t=25 este introdusă perturbația pe ieșire cu valoarea 0.2. Aceasta este eliminată complet cu un timp de răspuns de aprox. 10s. Eliminarea perturbației este realizată la t=40s, iar răspunsul procesului confirmă timpul de rejectare de 10s.

Robustețea la variații ale factorului de amplificare de 50% este demonstrată în Fig. 28. După cum se poate vedea, timpul de răspuns al procesului alterat este de 20s, fată de 7s al procesului nominal. Și pentru aceste tipuri de procese este demonstrată eficacitatea strategiei de control de ordin fracționar bazat pe evenimente.



Fig. 26. Răspunsului sistemului în buclă închisă cu filtru fracționar și regulator PID bazat pe evenimente la o intrare treaptă unitară

Fig. 27. Rejectarea perturbației pe ieșire

Fig. 28. Verificarea robusteții la variații ale factorului de amplificare

Testarea algoritmului de control FO-IMC bazat pe evenimente pe un sistem de ordin superior cu timp mort. Studiu de caz pe un echipament de tip vertical take off and landing (VTOL)

Procesul este descris de următoarea funcție de transfer:

$$H_{VTOL}(s) = \frac{22.24}{s^2 + 0.6934s + 5.244} e^{-0.8s}$$
(65)

iar pentru determinarea parametrilor FO-IMC s-au utilizat aceleași mărimi de performanță ca cele din Activitatea 2.2: PM=75° și ω_c =0.5 rad/s, rezultând: λ =1.156 și α =0.9379, precum și funcția de transfer: H_{FO-IMC_NRTF}(s) = Regulatorul echivalent,

identic celui obținut în cadrul Activitații 2.2, pentru aceiași set de performanțe, este:

$$H_{c_NRTF}(s) =$$
(66)

fiind implementat ca și un regulator clasic PID (în algoritm bazat pe evenimente) și un filtru de ordin fracționar:

$$H_c(s) = \frac{s^2}{2} \tag{67}$$

Reprezentarea schematică, în buclă închisă, este redată în Fig. 21, ecuația folosită pentru implementarea PID bazat pe evenimente fiind cea din (54), unde $\beta=1$ și $\gamma=1$, rezultând:

$$H_{PID}(s) =$$
(68)

Pentru atenuarea zgomotelor, se va adauga un filtru pe componenta derivativă, ecuația din (68) fiind astfel modificată:

$$H_{PID}(s) =$$
(69)

Pe baza echivalenței dintre (69) și (54), rezultă: $k_p\!=\!0.6934/22.24$, $T_i\!=\!0.6934/5.244$ și $T_d\!=\!1/0.6934,\,1\!+\!T_d/N_ds\!=\!1\!+\!0.1s$ ceea ce duce la $N_d\!=T_d/0.1\!=\!10/0.6934.$

Filtrul de ordin fracționar este dat de:

$$H_{FO}(s) =$$
(70)

pe baza ecuației (70) obținându-se relația de recurență:

$$e_f(k) =$$

. . .

calculată utilizând metoda NRTF de aproximare cu N=5, δ =0.9 și ω_h =200 π (T_s=0.005 s).

Parametrii de implementare sunt hnom = Ts = 0.005, hact = 0.05 de 10 ori mai mare decat hnom, iar Delta e = 0.1 rămane la fel ca în celelalte cazuri.

Răspunsul la o treaptă unitară este prezentat în Fig. 29. Se poate vedea o reducerea majoră a suprareglajului, un timp de răspuns de 10s și o eroare staționar nulă față de procesul necompensat. Figura 30 prezintă simularea sistemului în buclă inchisă la o perturbație pe ieșire, aceasta fiind eleiminată în 5s. Simulările în buclă închisă prezentate în Fig. 31 arată un caracter robust la variații de 20% ale factorului de amplificare. Regulatorul bazat pe evenimente pentru acest exemplu este de tip PID, față de celelalte două exemple care prezintă regulatoare PI. Cu toate acestea, strategia de reglare propusă iși demonstrează utilitatea.

Având în vedere testele prezentate, strategia propusă a fost validată pe o serie de procese cu timp mort considerate a fi reprezentative.

Fig. 29. Răspunsului sistemului în buclă închisă cu filtru fracționar și regulator PID bazat pe evenimente la o intrare treaptă unitară

Fig. 30. Rejectarea perturbației pe ieșire

Fig. 31. Verificarea robusteții la variații ale factorului de amplificare

Activitatea 2.7. Diseminarea rezultatelor

Toate activitățile menționate în planul de realizare pentru Etapa II, anul 2019, au fost realizate în proporție de 100%.

Rezultate estimate (menționate	Rezultate realizate în anul 2019
pentru anul 2019)	
• 3 lucrări trimise la reviste ISI	 2 lucrări publicate în reviste ISI (factor de impact cumulat 8.196) 1 lucrare ISI publicată în revista Heliyon, recent indexată ISI 1 lucrare aflată în recenzie la revista ISI (factor de impact de 3.526)
• 4 lucrări acceptate în cadrul unor conferințe internaționale	 2 lucrări prezentate și publicate în proceedings- urile unor conferințe indexate ISI 5 lucrări prezentate și publicate în proceedings- urile unor conferințe în curs de indexare ISI 1 lucrare prezentată și publicată în proceedings- urile unei conferințe indexate Scopus cu acces tip open access 2 lucrări trimise la conferință indexată ISI proceedings (European control conference)
• mobilități de cercetare	• 4 mobilități de cercetare finanțate (2 luni)

Lucrări publicate, în recenzie și în curs de publicare

 De Keyser, R., Muresan, C.I., Ionescu, C. (2019), Universal Direct Tuner for Loop Control in Industry, IEEE Access, vol. 7, pp. no. 1, pp. 81308-81320, DOI: 10.1109/ACCESS.2019.2921870, WOS:000474826900001 (ISI impact factor 4.098)

- Birs, I., Muresan, C.I., Nascu, I., Ionescu, C. (2019), A Survey of Recent Advances in Fractional Order Control for Time Delay Systems, IEEE Access, vol. 7, no. 1, pp. 30951-30965, DOI: 10.1109/ACCESS.2019.2902567, WOS:000462469400001 (ISI impact factor 4.098)
- C. Muresan, I. Birs, R. De Keyser (2019), An Improved Design Approach for Fractional Order Internal Model Controllers for Time Delay Systems, Mechatronics, under review (ISI impact factor 3.526)
- Juchem, J., Muresan, C.I., De Keyser, R., Ionescu, C.M. (2019), Robust fractional-order auto-tuning for highly-coupled MIMO systems, Heliyon, Vol. 5, No. 7, paper e02154, DOI: 10.1016/j.heliyon.2019.e02154, WOS:000478663100007 (ISI Emerging citations Index)
- De Keyser, R., Muresan, C.I. (2019), Validation of the KC Autotuning Principle on a Multi-Tank Pilot Process, 12th IFAC Symposium on Dynamics and Control of Process Systems, including Biosystems, Florianópolis – SC, Brazil, April 23-26, 2019, IFAC Papers Online, vol. 52, no. 1, pp. 178-183, DOI: 10.1016/j.ifacol.2019.06.05, WOS:000473270600031 (ISI PROCEEDINGS)
- Birs, I., Copot, D., Muresan, C.I., De Keyser, R., Ionesc, C.M. (2019), Robust Fractional Order PI Control for Cardiac Output Stabilisation, 12th IFAC Symposium on Dynamics and Control of Process Systems, including Biosystems, Florianópolis – SC, Brazil, April 23-26, 2019, IFAC-PapersOnLine, vol. 52, no. 1, pp. 994-999, DOI:10.1016/j.ifacol.2019.06.192, WOS:000473270600165 (ISI PROCEEDINGS)
- De Keyser, R., Muresan, C.I.(2019), Robust Estimation of a SOPDT Model from Highly Corrupted Step Response Data, European Control Conference, pp. 818-823, DOI:23919/ECC.2019.8796103, Naples, Italy, 25-28 June 2019 (in curs de indexare ISI PROCEEDINGS)
- Muresan, C.I., Copot, C., Ionescu, C., De Keyser, R. (2019), Robust Fractional Order Control of LPV Dynamic Mechatronic Systems, The 15th IEEE International Conference on Control and Automation (IEEE ICCA 2019), July 16-19, 2019, Edinburgh, Scotland, accepted (in curs de indexare ISI PROCEEDINGS)

- Muresan, C.I., Birs, I., Darab, C., Prodan, O., De Keyser, R. (2019), Alternative Approximation Method for Time Delays in an IMC Scheme, The Aegean Conference on Electrical Machines and Power Electronics and Optimization of Electrical & Electronic Equipment Conference, Istanbul, Turkey, August 27-29 2019, accepted (in curs de indexare ISI PROCEEDINGS)
- De Keyser, R., Muresan, C.I. (2019), Experimental Validation of an Efficient Disturbance Rejection Method for Dead-Time Processes using Internal Model Control, 24th IEEE Conference on Emerging Technologies and Factory Automation, DOI: 10.1109/ETFA.2019.8869004, Zaragoza, Spain, September 10-13 2019 (in curs de indexare ISI PROCEEDINGS)
- 11. Birs, I, Muresan, C.I., Nascu, I., Ionescu, C. (2019), Design and Practical Implementation of a Fractional Order Proportional Integral Controller (FOPI) for a Poorly Damped Fractional Order Process with Time Delay, The 7th International Conference on Control, Mechatronics and Automation, Delft, Netherlands, November 6-8, 2019, accepted (in curs de indexare **ISI PROCEEDINGS**)
- 12. Muresan, C.I., Birs, I., Prodan, O., Nascu, I., De Keyser, R. (2019), Approximation Methods for FO-IMC Controllers for Time Delay Systems, The 2nd International Conference on Electrical Engineering and Green Energy, Roma, E3S Web of Conferences, vol. 115, DOI:1051/e3sconf/201911501003, Italy, June 28-30, 2019 (OPEN ACCESS/ SCOPUS INDEXED)
- Birs, I., Muresan, C., An event based implementation of a fractional order controller on a scalable nanorobot, The European Control Conference, St. Petersburg, Russia, 12-15 May 2020, submitted (in curs de indexare ISI PROCEEDINGS)
- 14. Birs, I., Muresan, C., Nascu, I., de Keyser, R.. An event-based implementation of the KC autotuner for vertical take-off and landing control, The European Control Conference, St. Petersburg, Russia, 12-15 May 2020, submitted (in curs de indexare ISI PROCEEDINGS)

Muresan Cristina Ioana