

DETERMINISM ȘI CONTROLABILITATE ÎNTR-O CLASĂ DE SISTEME HIBRIDE DE COMANDĂ

dr.ing. Virginia Ecaterina Oltean
Universitatea "Politehnica" București

Rezumat: În clasa de sisteme hibride de comandă (SHC) studiată în lucrare, un proces continuu este condus, prin intermediul unei interfețe, de un sistem cu evenimente discrete (SED), numit controler. Pornind de la obiectivul primar de comandă, spațiul continuu de stare este partiționat, prin intermediul unor hipersuprafețe netede, în celule, iar procesul continuu, cuplat cu interfața se comportă ca un SED, numit SED-proces. Lucrarea analizează câteva aspecte privind relația între comportarea nedeterministă a modelului automat al SED-proces și partiția spațiului continuu de stare, precum și implicațiile nedeterminismului asupra necontrolabilității unei evoluții dorite a SED-proces. Sunt introduse conceptelor de stare discretă absorbantă și de evoluție continuă mascată și sunt discutate implicațiile acestora asupra definiției nedeterminismului modelului logic al SED-proces.

Cuvinte cheie: sistem cu evenimente discrete (SED), sistem hibrid de comandă (SHC), automat, traiectorie continuă de stare, hipersuprafețe, nedeterminism, controlabilitate.

1. Introducere

Într-o clasă de sisteme hibride de comandă (SHC), un sistem cu evenimente discrete (SED), reprezentând controlerul, interacționează, prin intermediul unei interfețe, cu un proces continuu (figura 1a). Această structură, numită de unii autori abordarea de tip *supervizor* a SHC, a fost propusă inițial de P.J. Antsaklis și colaboratorii săi din cadrul Grupului ISIS (Notre-Dame, SUA) [1], [2], precum și de Nerode și Kohn [3].

Controlerul este modelat ca o mașină Moore. Procesul cuplat cu interfața se comportă ca un SED, numit SED-proces și este modelat ca un automat pur logic. Comportarea logică a SHC este descrisă de interacțiunea, în buclă închisă, a controlerului și a SED-proces (figura 1b). Pornind de la un obiectiv primar de comandă, reprezentat de una sau mai multe secvențe admisibile (ciclice sau aciclice) de restricții de tip inegalitate strictă, impuse variabilelor de stare ale procesului continuu, spațiul continuu de stare este partiționat, prin intermediul unor hipersuprafețe netede, într-un set finit de celule deschise, două câte două disjuncte; fiecare celulă este o intersecție de semispații deschise și corespunde unei stări discrete a automatului SED-proces. Partiția de stare este implementată de generatorul de evenimente din interfață. Un *eveniment-proces* se produce și *simbolul-proces* corespunzător este trimis, prin interfață, spre controler, ori de câte ori o hipersuprafață este traversată, într-un anumit sens, de către o traiectorie continuă de stare. Un simbol-proces etichetează semispațiul deschis în care traiectoria de stare este pe punctul de a pătrunde, și nu celula destinație. Producerea unui eveniment-proces este simultană cu executarea unei tranziții de stare discrete în

automatul SED-proces. Interfața mai are și rolul de a converti, prin subsistemul numit, în mod abuziv, element de execuție, o secvență de *simboluri-comandă*, generate de controler, într-un semnal real de comandă, constant pe porțiuni.

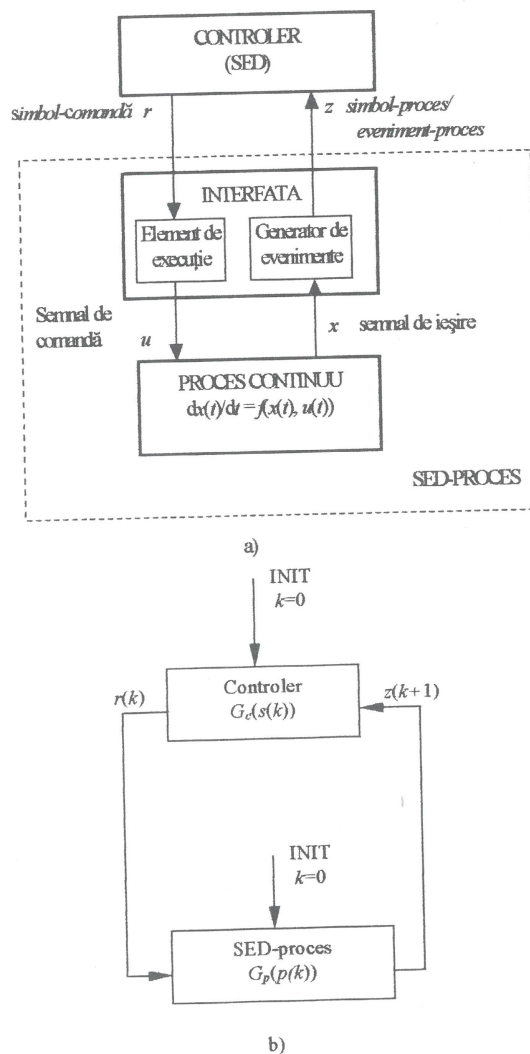


Figura 1. a) Structura unui SHC [1]; b) funcționarea în timp logic a SHC [4].

Conform [1], automatul SED-proces este *determinist* dacă orice stare discretă următoare poate fi univoc determinată pe baza stării curente și simbolului-comandă primit la momentul logic curent. Deoarece o mașină de stare finită este doar o aproximare a unei evoluții continue, automatul SED-proces este, în general, *nedeterminist*, în ciuda faptului că ecuațiile continue de stare reprezintă un model determinist. Nedeterminismul automatului SED-proces este cauza structurală a

necontrolabilității comportării dorite a SED-proces, care, la rândul său, conduce la dificultăți în sinteza controlerului [4], [5]. Conceptul, mai slab, de *cvasideterminism*, este introdus în [1].

Lucrarea de față prezintă relația dintre partiția spațiului continuu de stare, nedeterminism și necontrolabilitatea evoluțiilor discrete dorite ale SED-proces. Articolul este structurat după cum urmează. În secțiunea 2 este prezentată, succint, o variantă a formalismului Antsaklis, propusă inițial în [4]. Principalele proprietăți ale automatului SED-proces sunt analizate în secțiunea 3; sunt introduse conceptul de *evoluție continuă mascată*, conceptul de *stare discretă absorbantă*, o variantă a conceptului clasic de nedeterminism, specifică aproximărilor logice prin evenimente discrete ale proceselor continue, precum și conceptul de *urmărire deterministă stabilă* sau de *controlabilitate* a unei evoluții admisibile a automatului SED-proces [4]. În secțiunea 4 sunt prezentate, comparativ, trei modele SED-proces asociate sistemului continuu dublu integrator, cu partiții de stare, respectiv, distincte. Discuția este centrată pe transformarea, nealgoritmă, a partițiilor de stare, în scopul ameliorării gradului de determinism al automatului SED-proces respectiv.

2. Structura unui sistem hibrid de comandă – o scurtă prezentare

Structura SHC este prezentată în figura 1a.

Procesul continuu este modelat de sistemul diferențial

$$\dot{x}(t) = f(x(t), u(t)), \quad (1)$$

unde $x \in \mathbb{R}^n$ este vectorul de stare, $u \in \mathbb{R}^m$ este comanda, iar $t \in \mathbb{R}$ este variabila timp continuu. Setul valorilor admisibile ale comenzii este $U = \{u_1, \dots, u_M\} \subset \mathbb{R}^m$ și alfabetul simbolurilor-comandă este $R = \{r_1, \dots, r_M\}$.

Elementul de execuție implementează funcția bijectivă de selecție $\gamma: R \rightarrow U, \gamma(r_m) = u_m$, $m = 1, \dots, M$. Fie $k \in \{0, 1, \dots\}$ variabila timp logic. O secvență de simboluri-comandă $w_r = r(0), r(1), \dots, r(k), \dots$, $w_r \in R^*$, generează un semnal de comandă constant pe porțiuni $u(t) = \sum_{k \geq 0} \gamma(r(k)) \cdot I(t, t_c(k), t_c(k+1))$, unde $t_c(k) \in \mathbb{R}^+$ este momentul la care s-a primit $r(k)$, $t_c(k) < t_c(k+1)$, $\forall k \in \{0, 1, \dots\}$ și $I: \mathbb{R}^+ \times \mathbb{R}^+ \times \mathbb{R}^+ \rightarrow \{0, 1\}$ este funcția de extrapolare cu valorile: $I(t, t_1, t_2) = 1$, dacă $t_1 \leq t < t_2$ sau $I(t, t_1, t_2) = 0$, altfel.

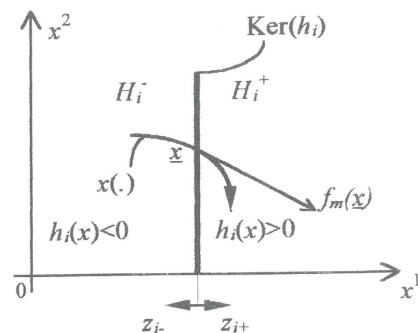


Figura 2. Exemplu: evenimentul-proces $(i+)$ se produce ori de câte ori traiectoria continuă de stare $x(\cdot)$ traversează $\text{Ker}(h_i)$ în sensul pozitiv, sub acțiunea unei comenzi

$$u_m = \gamma(r_m); f_m(x) = f(x, \gamma(r_m)), \text{ cu } r_m \in R \text{ și } x \in X = \mathbb{R}^2 \text{ [4], [6].}$$

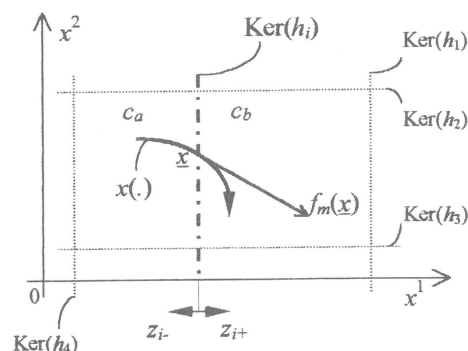


Figura 3. Exemplu: producerea tranziției discrete de stare $p_a \rightarrow p_b$ este observată prin intermediul simbolului-proces z_{i+} , iar celulele $c_a = et^{-1}(p_a)$ și $c_b = et^{-1}(p_b)$ sunt adiacente pe $\text{Ker}(h_i)$ [4], [6].

Generatorul de evenimente implementează o partiție a spațiului continuu de stare $X = \mathbb{R}^n$, definită cu ajutorul unui set de $N \geq 1$ funcționale netede $h_i: X \rightarrow \mathbb{R}$, $i = 1, \dots, N$. Hipersuprafețele nesingulare $\text{Ker}(h_i) = \{x \in X \mid h_i(x) = 0\}$, $i = 1, \dots, N$, definesc în X o acoperire de $Q \leq 2^N$ deschiși (disjuncti), etichetați biunivoc cu literele alfabetului $P = \{p_1, \dots, p_Q\}$. Un simbol-proces z_{i+} (sau z_{i-}) etichetează, univoc, un eveniment-proces $(i+)$ (respectiv $(i-)$), definit în raport cu $\text{Ker}(h_i)$ (figura 2). Simbolul-proces z_{i+} (sau z_{i-}) este trimis spre controler la momentul $t \in \mathbb{R} \Leftrightarrow h_i(x(t)) = 0$ și $dh_i(x(t))/dt > 0$ (respectiv $h_i(x(t)) = 0$ și $dh_i(x(t))/dt < 0$). Alfabetul simbolurilor-proces este $Z = \{z_{1+}, z_{1-}, \dots, z_{N+}, z_{N-}\}$.

SED-proces este modelul logic al tuturor comportărilor posibile ale procesului continuu, cuplat cu interfața și comandat de orice semnal generat de secvențe din alfabetul R . Cu alte cuvinte, se presupune că procesul poate fi comandat de orice secvență de simboluri-comandă din R și, în plus, starea inițială $x(0)$ poate fi plasată, arbitrar, în orice celulă a partiției, care devine, astfel, celula inițială. SED-proces este modelat de automatul nedeterminist $G_p = \{P, R, f_p, Z, g_p\}$, unde P este setul stărilor discrete, R este alfabetul de intrare, Z este alfabetul de ieșire, $f_p : P \times R \rightarrow 2^P$ este funcția de tranziție de stare și $g_p : P \times P \rightarrow Z$ este funcția de ieșire. Ecuațiile de dinamică sunt:

$$p(k+1) \in f_p(p(k), r(k)), \quad (2.1)$$

$$g_p(p(k), p(k+1)) = z(k+1), \quad (2.2)$$

unde $p(k) \in P$, $z(k+1) \in Z$, $r(k) \in R$ și $p(k) \neq p(k+1)$, $\forall k \in \{0, 1, \dots\}$ [4], [6].

Controlerul este o mașină Moore deterministă $G_c = \{S, Z, f_c, s_0, R, g_c\}$, unde S este setul finit al stărilor discrete, $s_0 \in S$ este starea inițială, Z este alfabetul de intrare, R este alfabetul de ieșire, $f_c : S \times Z \rightarrow S$ este funcția de tranziție de stare și $g_c : S \rightarrow R$ este funcția de ieșire. Ecuațiile de dinamică sunt:

$$f_c(s(k), z(k+1)) = s(k+1), \quad (3.1)$$

$$g_c(s(k+1)) = r(k+1), \quad (3.2)$$

cu $s(0) = s_0$, $g_c(s_0) = r(0)$ și $s(k) \in S$, $r(k) \in R$, $z(k+1) \in Z$, $\forall k \in \{0, 1, \dots\}$. La $k=0$ se produce evenimentul INIT, de punere în funcțiune (figura 1b).

Funcționarea în timp logic a SHC (figura 1b) este descrisă de evoluția controlerului G_c cuplat, dintr-o stare inițială specificată $p_0 \in P$, cu G_p , conform ecuațiilor

$$(G_c / G_p): \quad (4.1)$$

$$p(k+1) \in f_p(p(k), g_c(s(k))), \quad (4.1)$$

$$s(k+1) = f_c(s(k), g_p(p(k), p(k+1))), \quad (4.2)$$

$$g_c(s(k)) = r(k), \quad (4.3)$$

$$g_p(p(k), p(k+1)) = z(k+1), \quad (4.4)$$

unde $s(0) = s_0$, $p(0) = p_0$ și $k \in \{0, 1, \dots\}$ [4].

Observație. Ecuațiile (2), (3) și (4) arată, formal, că o evoluție a lui G_p , notată

$$w_p = p(0), p(1), \dots, p(k), \dots,$$

este comandată de o secvență de simboluri-comandă

$$w_r = r(0), r(1), \dots, r(k), \dots$$

și este observată ca o secvență de simboluri-proces

$$w_z = z(1), z(2), \dots, z(k+1), \dots,$$

cu $p(k) \in P$, $r(k) \in R$, $z(k+1) \in Z$ și $k \in \{0, 1, \dots\}$. $\square$

În sinteza SHC, se pornește de la un obiectiv primar de comandă, exprimat ca o secvență de deschiși din spațiul continuu de stare, fiecare deschis fiind adiacent cu precedentul și respectiv cu următorul. Se dorește ca traiectoria continuă $x(\cdot)$, inițializată arbitrar în primul deschis și observată doar prin impulsurile generate la trecerea de la un deschis la altul, să urmărească, în mod maximal permisiv, secvența specificată.

Pe baza a) obiectivului primar, b) a modelului procesului continuu și c) a unui set dat U de valori admisibile ale comenzii, în sinteza interfeței și a controlerului se parcurg următoarele etape principale:

I. se construiește partiția spațiului continuu de stare, i.e. se alege setul de funcționale h_i , $i = 1, \dots, N$;

II. se construiește modelul automat G_p al SED-proces;

III. se remodelează obiectivul primar ca un subautomat $G_{loop}^D \subseteq G_p$, reprezentând stările discrete dorite, împreună cu tranzițiile dorite între acestea și se sintetizează, dacă este posibil, un controler G_c , ca soluție maximal permisivă – i.e. optimală – a problemei stărilor și tranzițiilor dorite în SED-proces G_p , prin adaptarea teoriei Ramadge-Wonham [4], [7].

Eventuala necontrolabilitate a comportării discrete dorite a SED-proces este legată de nedeterminismul automatului G_p și, dacă problema, reformulată în cadrul teoriei SED, nu are soluție, atunci se încearcă să se construiască o nouă partiție de stare, cu o granularitate mai fină, după care sinteza se reia din etapa II. Dacă toate tranzițiile din G_{loop}^D sunt controlabile, atunci G_p , cuplat, din starea inițială specificată, cu G_c , evoluează ca G_{loop}^D (figura 1b).

SED-proces este modelul logic al tuturor comportărilor posibile ale procesului continuu, cuplat cu interfața și comandat de orice semnal generat de secvențe din alfabetul R . Cu alte cuvinte, se presupune că procesul poate fi comandat de orice secvență de simboluri-comandă din R și, în plus, starea inițială $x(0)$ poate fi plasată, arbitrar, în orice celulă a partiției, care devine, astfel, celula inițială. SED-proces este modelat de automatul nedeterminist $G_p = \{P, R, f_p, Z, g_p\}$, unde P este setul stărilor discrete, R este alfabetul de intrare, Z este alfabetul de ieșire, $f_p : P \times R \rightarrow 2^P$ este funcția de tranziție de stare și $g_p : P \times P \rightarrow Z$ este funcția de ieșire. Ecuațiile de dinamică sunt:

$$p(k+1) \in f_p(p(k), r(k)), \quad (2.1)$$

$$g_p(p(k), p(k+1)) = z(k+1), \quad (2.2)$$

unde $p(k) \in P$, $z(k+1) \in Z$, $r(k) \in R$ și $p(k) \neq p(k+1)$, $\forall k \in \{0, 1, \dots\}$ [4], [6].

Controlerul este o mașină Moore deterministă $G_c = \{S, Z, f_c, s_0, R, g_c\}$, unde S este setul finit al stărilor discrete, $s_0 \in S$ este starea inițială, Z este alfabetul de intrare, R este alfabetul de ieșire, $f_c : S \times Z \rightarrow S$ este funcția de tranziție de stare și $g_c : S \rightarrow R$ este funcția de ieșire. Ecuațiile de dinamică sunt:

$$f_c(s(k), z(k+1)) = s(k+1), \quad (3.1)$$

$$g_c(s(k+1)) = r(k+1), \quad (3.2)$$

cu $s(0) = s_0$, $g_c(s_0) = r(0)$ și $s(k) \in S$, $r(k) \in R$, $z(k+1) \in Z$, $\forall k \in \{0, 1, \dots\}$. La $k=0$ se produce evenimentul INIT, de punere în funcțiune (figura 1b).

Funcționarea în timp logic a SHC (figura 1b) este descrisă de evoluția controlerului G_c cuplat, dintr-o stare inițială specificată $p_0 \in P$, cu G_p , conform ecuațiilor

(G_c / G_p) :

$$p(k+1) \in f_p(p(k), g_c(s(k))), \quad (4.1)$$

$$s(k+1) = f_c(s(k), g_p(p(k), p(k+1))), \quad (4.2)$$

$$g_c(s(k)) = r(k), \quad (4.3)$$

$$g_p(p(k), p(k+1)) = z(k+1), \quad (4.4)$$

unde $s(0) = s_0$, $p(0) = p_0$ și $k \in \{0, 1, \dots\}$ [4].

Observație. Ecuațiile (2), (3) și (4) arată, formal, că o evoluție a lui G_p , notată

$$w_p = p(0), p(1), \dots, p(k), \dots,$$

este comandată de o secvență de simboluri-comandă

$$w_r = r(0), r(1), \dots, r(k), \dots$$

și este observată ca o secvență de simboluri-proces

$$w_z = z(1), z(2), \dots, z(k+1), \dots,$$

cu $p(k) \in P$, $r(k) \in R$, $z(k+1) \in Z$ și $k \in \{0, 1, \dots\}$. $\square$

În sinteza SHC, se pornește de la un obiectiv primar de comandă, exprimat ca o secvență de deschiși din spațiul continuu de stare, fiecare deschis fiind adiacent cu precedentul și respectiv cu următorul. Se dorește ca traiectoria continuă $x(\cdot)$, inițializată arbitrar în primul deschis și observată doar prin impulsurile generate la trecerea de la un deschis la altul, să urmărească, în mod maximal permisiv, secvența specificată.

Pe baza a) obiectivului primar, b) a modelului procesului continuu și c) a unui set dat U de valori admisibile ale comenzii, în sinteza interfeței și a controlerului se parcurg următoarele etape principale:

- I. se construiește partiția spațiului continuu de stare, i.e. se alege setul de funcționale h_i , $i = 1, \dots, N$;
- II. se construiește modelul automat G_p al SED-proces;
- III. se remodeleză obiectivul primar ca un subautomat $G_{loop}^D \subseteq G_p$, reprezentând stările discrete dorite, împreună cu tranzițiile dorite între acestea și se sintetizează, dacă este posibil, un controler G_c , ca soluție maximal permisivă – i.e. optimală – a problemei stărilor și tranzițiilor dorite în SED-proces G_p , prin adaptarea teoriei Ramadge-Wonham [4], [7].

Eventuala necontrolabilitate a comportării discrete dorite a SED-proces este legată de nedeterminismul automatului G_p și, dacă problema, reformulată în cadrul teoriei SED, nu are soluție, atunci se încearcă să se construiască o nouă partiție de stare, cu o granularitate mai fină, după care sinteza se reia din etapa II. Dacă toate tranzițiile din G_{loop}^D sunt controlabile, atunci G_p , cuplat, din starea inițială specificată, cu G_c , evoluează ca G_{loop}^D (figura 1b).

3. Proprietăți ale modelului automat SED-proces

3.1 Conținutul semantic al unei tranziții de stare discrete în G_p

Pentru a explica relația între evoluția continuă și evoluția discretă a procesului, sunt necesare câteva definiții suplimentare, introduse în [4] și în [6].

Fie $DX = X - Fr$, cu $Fr = \bigcup_{i=1}^N Ker(h_i)$.

Relația de echivalență construită de partiția de stare este $Rel \subset DX \times DX$, definită prin:
 $(x_1, x_2) \in Rel \Leftrightarrow h_i(x_1)h_i(x_2) > 0, \forall h_i, i = 1, N$.

Spațiul celular $DX_{/Rel} = C = \{c_1, \dots, c_Q\}$ este mulțimea claselor de echivalență ale relației Rel ; s-au notat $c_q \in C$ o celulă și $I_Q = \{1, \dots, Q\}$ setul de indecși ai mulțimii C . Aplicația bijectivă $et: C \rightarrow P$, $et(c_q) = p_q, \forall q \in I_Q$ este funcția de etichetare a mulțimii C . Surjecția canonică în raport cu C este aplicația $sur_C: DX \rightarrow C$, definită astfel încât $sur_C(x) = c_q \Leftrightarrow x \in c_q$ [4], [6].

Fie procesul continuu (1) și $k \in \{0, 1, \dots\}$ momentul logic prezent. Presupunem că, la momentul $t_c(k) \in \mathbb{R}$, starea continuă este $x(t_c(k)) = x_0 \in c(k) \in C$, automatul G_p este în starea discretă $p(k) = et^{-1}(c(k))$ și primește simbolul-comandă $r(k) \in R$. Prin integrarea sistemului diferențial (1), cu intrarea constantă $\gamma(r(k)) \in U$, rezultă starea continuă la momentul $t \geq t_c(k)$,

$$F_k(x_0, t) \equiv x(t) = x_0 + \int_{t_c(k)}^t f(x(\theta), \gamma(r(k))) d\theta. \quad (5)$$

Celula asociată stării discrete $p(k)$ este $c(k) = sur_C(x_0)$. La momentul

$$\underline{t} = \min\{t > t_c(k) \mid et^{-1}(p(k)) \neq sur_C(F_k(x_0, t)), F_k(x_0, t_c(k)) = x_0\} \quad (6)$$

traectoria continuă de stare pătrunde în celula $c(k+1) = sur_C(F_k(x_0, \underline{t})) \in C$. Presupunând că traectoria continuă de stare nu trece prin punctul de intersecție a două sau mai multor hipersuprafețe, rezultă că există o singură hipersuprafață, $Ker(h_i)$,

a partiției de stare, pe care sunt adiacente celulele $c(k)$ și $c(k+1)$. În concluzie, dându-se $p(k) \in P$ și $r(k) \in R$, dinamica discretă a lui G_p , de la momentul logic k la $k+1$, este descrisă de relațiile:

$$\begin{aligned} p(k+1) &\in f_p(p(k), r(k)), \\ p(k+1) &= et(sur_C(F_k(x_0, t))), x_0 \in et^{-1}(p(k)). \end{aligned} \quad (7)$$

În plus, $g_p(p(k), p(k+1)) = z(k+1)$ și $z(k+1) = z_{i+}$ (sau z_{i-}) dacă hipersuprafața $Ker(h_i)$ este traversată de traectoria continuă de stare $x(.) = F_k(x_0, .)$ în sens pozitiv (respectiv în sens negativ).

O definiție echivalentă a stării curente a automatului SED-proces pornește de la următoarea considerație, introdusă în [1]. La momentul

$t_e = \underline{t} - \varepsilon$, (la care $h_i(x(t_e)) = 0$ și $\dot{h}_i(x(t_e)) \neq 0$) starea continuă tocmai părăsește hipersuprafața de adiacență $Ker(h_i)$ și se produce tranziția de stare discretă $p(k) \rightarrow p(k+1)$, observată prin simbolul-proces $z(k+1)$ (figura 2 și figura 3). Automatul SED-proces G_p este în starea $p(k+1)$, dacă este îndeplinită condiția: $\lim_{\varepsilon \rightarrow 0} x(t_e + \varepsilon) \in et^{-1}(p(k+1))$.

3.2 Evoluții continue mascate și stări absorbante în automatul G_p

Conceptele de evoluție continuă mascată a procesului și, respectiv, de stare discretă absorbantă a automatului SED-proces au fost introduse, inițial, în [4]. Ele sunt specifice doar modelelor pur logice, cu evenimente discrete, ale sistemelor dinamice continue.

Presupunem că $p_s \in f_p(p_q, r_m)$, cu $p_q, p_s \in P$, $r_m \in R$ și, în plus, că $\exists x_0 \in c_q = et^{-1}(p_q)$ cu proprietatea

$F_m(x_0, t) \in c_q, \forall t \geq t_0$, unde t_0 este momentul de timp la care procesul a primit simbolul-comandă

r_m și $F_m(x_0, t) = x_0 + \int_{t_0}^t f(x(\theta), \gamma(r_m)) d\theta$. Atunci

traectoria $x(.) \equiv F_m(x_0, .)$ a procesului (1) este o evoluție continuă mascată, asociată perechii (p_q, r_m) . Prezența sau absența evoluțiilor continue mascate poate fi analizată doar pe baza portretelor de fază ale sistemului (1), trasate, respectiv, pentru fiecare comandă $u(.) \equiv \gamma(r_m)$, cu $r_m \in R$.

În modelul automat SED-proces G_p , o stare $p_q \in P$ este absorbantă sub acțiunea unui simbol-comandă $r_m \in R$, dacă $f_p(p_q, r_m)$ nu este definită, adică dacă nu se produce nici o tranziție discretă într-o stare $p_l \in P$, $p_l \neq p_q$, când G_p se află în starea p_q și primește simbolul-comandă r_m .

3.2 Comportări nedeterminate în automatul SED-proces G_p

Nedeterminismul automatului SED-proces G_p este principala sursă a dificultăților în sinteza controlerului G_c . Această proprietate a fost analizată în [1] și [2] și rafinată în [4].

Automatul SED-proces G_p este determinist dacă $\forall p_a \in P$ și $\forall r_m \in R$, fie 1) $f_p(p_a, r_m)$ nu este definită (adică starea p_a este absorbantă sub acțiunea simbolului-comandă r_m , astfel încât nu are loc nici o tranziție de stare discretă comandată, din p_a , de r_m), fie 2) r_m are o acțiune deterministă din p_a , adică 2.1) $f_p(p_a, r_m)$ este definită și $\text{card}(f_p(p_a, r_m)) = 1$ (astfel încât, în G_p , există doar o singură tranziție discretă de stare, din p_a , comandată de r_m) și 2.2) procesul continuu (1) nu are nici o evoluție continuă mascată, asociată perechii (p_a, r_m) .

Dacă $\exists p_a \in P$ și $\exists r_m \in R$ astfel încât a) $\text{card}(f_p(p_a, r_m)) > 1$ sau b) $\text{card}(f_p(p_a, r_m)) = 1$ și există cel puțin o evoluție continuă mascată asociată perechii (p_a, r_m) , atunci automatul SED-proces G_p este nedeterminist. Numim condiția a) *nedeterminism clasic* și condiția b) *nedeterminism specific hibrid*.

Reamintim că modelul (1) al procesului continuu este determinist. Conform conținutului semantic al automatului SED-proces, descris în relațiile (7), pentru un proces (1), o partiție de stare și un simbol-comandă fixate, determinismul automatului G_p depinde de condiția inițială x_0 , necunoscută pentru controlerul SED. Fie starea inițială discretă (sursă) $p(k) = p_a$, două inițializări posibile și distincte ale stării continue, $x_{01}, x_{02} \in c(k) = et^{-1}(p(k))$, situate, la momentul logic curent k , în celula sursă și simbolul-comandă $r_m = r(k) \in R$, primit la momentul $t_c(k) \in \mathbb{R}$. Fie,

de asemenea, momentele de timp $t_1, t_2 \in \mathbb{R}$, definite conform relației (6), la care traiectoriile continue de stare $F_k(x_{01}, \cdot)$, $F_k(x_{02}, \cdot)$ (5) trec, respectiv, în celula adiacentă următoare $c(k+1)$.

Intuitiv, dacă celula sursă $c(k)$ este "prea mare" și partiția de stare este prea grosieră, atunci este posibil ca:

$$\text{I) } p_b = p(k+1) = et(\text{sur}_C(F_k(x_{01}, t_1))),$$

$$\text{sau II) } p_d = p(k+1) = et(\text{sur}_C(F_k(x_{02}, t_2)))$$

$$\text{și III) } p_d \neq p_b,$$

adică r_m poate avea o acțiune nedeterministă din starea p_a (i.e. nedeterminism clasic).

O altă sursă de nedeterminism (i.e. nedeterminismul specific hibrid) poate să apară în situația în care: I') $t_1 < \infty$ (adică $p_b = p(k+1) = et(\text{sur}_C(F_k(x_{01}, t_1)))$) dar II') $t_2 = \infty$ (adică $F_k(x_{02}, \cdot)$ este o evoluție continuă mascată, asociată perechii (p_a, r_m)).

În consecință, se consideră, mai întâi, procesul continuu (1), o partiție de stare, un set U de valori admisibile ale comenzii și alfabetul R al simbolurilor-comandă. Dacă modelul automat al SED-proces, construit pe baza acestor date, este nedeterminist, atunci o posibilitate de a-l transforma într-un automat determinist constă în construcția unei noi partiții de stare, cu o granularitate mai fină decât a partiției inițiale. Motivația acestui efort este relația directă între determinism și controlabilitatea comportărilor discrete dorite, prezentată în subsecțiunea următoare.

3.4 Urmărirea deterministă stabilă și controlabilitatea unei evoluții admisibile în G_p

O evoluție $w_p = p(0), p(1), \dots, p(k), \dots$, cu $w_p \in P^*$, a automatului SED-proces G_p , este admisibilă dacă există o secvență de simboluri-comandă $w_r = r(0), r(1), \dots, r(k), \dots$, cu $w_r \in R^*$, astfel încât, pentru $\forall p(k), p(k+1) \in w_p$, simbolul-comandă $r(k) \in w_r$, asigură satisfacerea relației (2.1). Prin definiție, secvența de simboluri-comandă $w_r \in R^*$, cu proprietatea de mai sus, poate comanda (determinist sau nu) evoluția $w_p \in P^*$.

Intuitiv, proprietatea de urmărire deterministă stabilă a unei evoluții admisibile, $w_p \in P^*$, a automatului G_p cere ca, dându-se secvența de

celule, asociată evoluției $w_p \in P^*$, să existe o secvență de simboluri-comandă, $w_r \in R^*$, astfel încât traiectoria continuă de stare inițializată, arbitrar, în prima celulă și comandată cu legea generată de secvența respectivă de simboluri-comandă, să parcurgă, ordonat, toată secvența de celule; în plus, dacă secvența de celule nu este ciclică sau nu are ciclu final, atunci, odată ajunsă în celula finală, traiectoria continuă trebuie să nu o mai părăsească, sub acțiunea valorii de palier la care a comutat ultima dată comanda constantă pe porțiuni, i.e. în celula precedentă celulei finale.

Fie $w_p = p(0); p(1), \dots, p(k), \dots$, cu $w_p \in P^*$, o evoluție admisibilă a automatului SED-proces G_p și $w_r = r(0), r(1), \dots, r(k), \dots$, cu $w_r \in R^*$, o secvență de simboluri-comandă care poate comanda $w_p \in P^*$.

A) Dacă w_p este ciclică sau are un subciclu final, atunci secvența de simboluri-comandă $w_r \in R^*$ asigură urmărirea deterministă stabilă a lui w_p dacă, $\forall k \in \{0, 1, \dots\}$, simbolul-comandă $r(k) \in w_r$ are o acțiune deterministă din starea discretă $p(k) \in w_p$.

B) Dacă w_p este aciclică sau nu are un subciclu final, i.e. $w_p = p(0), \dots, p(n-1), p(n)$, cu $n \geq 1$ un întreg fixat, atunci secvența de simboluri-comandă $w_r = r(0), \dots, r(n-1)$ asigură urmărirea deterministă stabilă a lui w_p dacă:

- b1) $\forall k \in \{0, 1, \dots\}$, $0 \leq k \leq n-1$, $r(k) \in w_r$ are o acțiune deterministă din starea discretă $p(k) \in w_p$ și
b2) la momentul $k=n$, automatul G_p rămâne blocat în starea $p(n)$, i.e. nu există tranziție spontană de ieșire din $p(n)$, după ce, la momentul logic $n-1$, s-a primit simbolul-comandă $r(n-1) \in w_r$.

Proprietate. Condiția b2 din definiția precedentă este satisfăcută dacă și numai dacă starea discretă finală, $p_a = p(n) \in w_p$, este absorbantă sub acțiunea simbolului-comandă $r_m = r(n-1) \in w_r$. $\square$ Demonstrația este evidentă.

Dacă o evoluție admisibilă, $w_p \in P^*$, reprezintă comportarea dorită a automatului G_p , atunci ea este controlabilă dacă și numai dacă există o secvență de simboluri-comandă, $w_r \in R^*$, care asigură urmărirea sa deterministă stabilă [4]. Evident că, pornind de la secvența de simboluri-comandă $w_r \in R^*$, se poate construi mașina Moore deterministă G_c care, cuplată în buclă

închisă (figura 1.b), din starea inițială $p(0) \in w_p$, cu G_p , conform ecuațiilor (4), asigură evoluția dorită w_p a lui G_p : în (4.1), $p(k+1) \in w_p$, $\forall k \in \{0, 1, \dots\}$ [4].

În concluzie, determinismul modelului automat al SED-proces joacă în rol crucial în sinteza politicilor adecvate de conducere, prin evenimente, a procesului continuu.

4. Trei modele automate SED-proces ale sistemului continuu dublu integrator

4.1 Modelul continuu, elementul de execuție și obiectivul primar de comandă

Aproximarea, printr-un SED, a sistemului continuu dublu integrator este clasică în literatura dedicată sistemelor hibride [1], [2]. În continuare, se consideră un SHC cu structura din figura 1a, în care procesul este reprezentat de sistemul continuu dublu integrator. Al doilea model automat SED-proces, din cele trei descrise în subsecțiunile următoare, a fost studiat în [4] și [6].

Ecuațiile de dinamică ale sistemului dublu-integrator sunt

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t), \quad (8)$$

unde $x = [x_1 \ x_2]^T \in \mathbb{R}^2$ este vectorul de stare, $u \in \mathbb{R}$ este comanda și $t \in \mathbb{R}$ este variabila timp continuu. Elementul de execuție din structura SHC (figura 1.a) poate genera comenzi din mulțimea $U = \{-1, 0, 1\}$. Alfabetul simbolurilor-comandă, $R = \{r_1, r_2, r_3\}$, este asociat mulțimii U , conform bijecției $\gamma: R \rightarrow U$, cu valorile: $\gamma(r_1) = -1$, $\gamma(r_2) = 0$, $\gamma(r_3) = 1$. Traectoriile de stare corespunzătoare sunt definite de relațiile

$$x(t) = \begin{bmatrix} 1 & t \\ 0 & 1 \end{bmatrix} x_0 + \begin{bmatrix} 0.5t^2 \\ t \end{bmatrix} \gamma(r_m) \quad (9)$$

cu $r_m \in R$ și $x_0 = x(t_0) \in \mathbb{R}^2$. Ecuațiile curbelor integrale sunt:

- (C1): $x^1(t) + (x^2(t))^2 / 2 = K$, pentru $\gamma(r_1) = -1$;
(C2): $x^2(t) = K$, pentru $\gamma(r_2) = 0$;
(C3): $x^1(t) - (x^2(t))^2 / 2 = K$, pentru $\gamma(r_3) = 1$.

K este o constantă reală ce depinde de starea inițială.

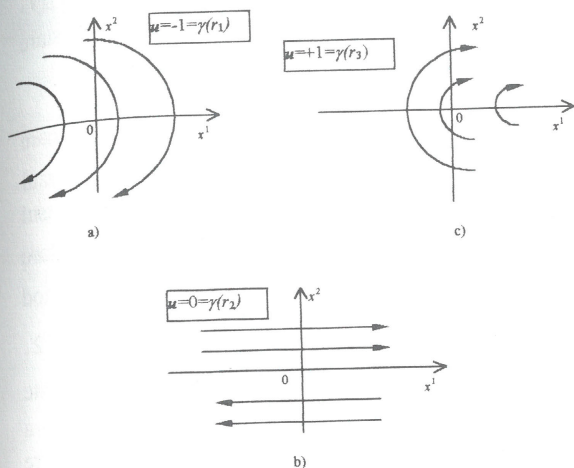


Figura 4. Traiectoriile de stare (9) ale sistemului dublu integrator (8).

Obiectivul primar de comandă cere să se asigure comanda, cu valori în U , a sistemului (8) astfel încât, pornind dintr-o inițializare arbitrară, situată în cadranul II, traiectoria de stare să parcurgă ciclul $II \rightarrow I \rightarrow IV \rightarrow III \rightarrow II \dots$

În continuare sunt descrise trei modele SED-proces, care au, în comun, procesul continuu (8), elementul de execuție specificat și obiectivul primar de comandă formulat mai sus.

4.2 Primul automat SED-proces G_{p0} : evoluții continue mascate și nedeterminism

Se consideră partiția de stare definită de $S_h^1 = \{h_1 : \mathbb{R}^2 \rightarrow \mathbb{R} \mid h_1(x) = x^1\}$. Hipersuprafața $Ker(h_1) = \{x \in \mathbb{R}^2 \mid h_1(x) = 0\}$ împarte planul fazelor în două celule, etichetate cu literele alfabetului $P_0 = \{p_{01}, p_{02}\}$, iar alfabetul simbolurilor-proces este $Z_0 = \{z_{1+}, z_{1-}\}$ (figura 5).

Pe baza portretelor de fază din figura 4 și a partiției de stare din figura 5, se deduce modelul automat al SED-proces $G_{p0} = \{P_0, R, f_{p0}, Z_0, g_{p0}\}$, reprezentat în figura 6. Funcția de tranziție de stare $f_{p0} : P_0 \times R \rightarrow 2^{P_0}$ este descrisă în tabelul 1.

Automatul G_{p0} este nedeterminist (specific hibrid), deoarece apar evoluții continue mascate. În figura 5, sunt reprezentate trei evoluții continue mascate, asociate perechii (p_{02}, r_1) , cu $\gamma(r_1) = -1$:

prima este traiectoria decelerată, inițializată în x_{02} , în interiorul parabolei de ecuație $x^1 + (x^2)^2 / 2 = 0$, cea de-a doua este traiectoria decelerată, inițializată în x_{03} , în zona situată în cadranul III, cuprinsă între parabola de ecuație $x^1 + (x^2)^2 / 2 = 0$ și $Ker(h_1)$, iar a treia este traiectoria decelerată ale cărei puncte satisfac chiar ecuația $x^1 + (x^2)^2 / 2 = 0$.

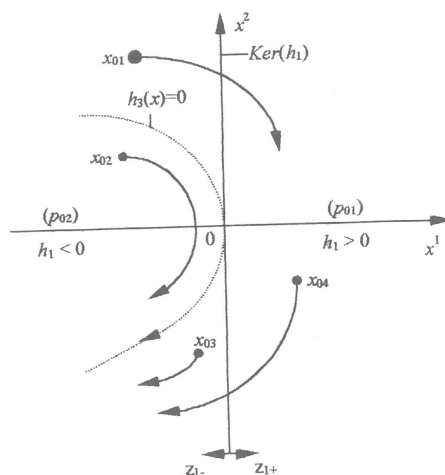


Figura 5. Partiția de stare definită de S_h^1 , cu $h_1(x) = x^1$; traiectoriile de stare inițializate în x_{02} și în x_{03} sunt evoluții continue mascate, asociată perechii (p_{02}, r_1) ; curba punctată este traiectoria mascată de ecuație $h_3(x) = 0$, cu $h_3(x) = x^1 + (x^2)^2 / 2$.

G_{p0} :

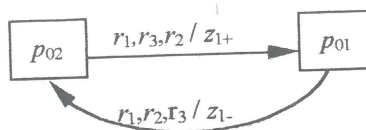


Figura 6. Modelul automat SED-proces G_{p0} , atașat sistemului (8), cu partiția de stare din figura 5.

Tabelul 1

Funcția de tranziție de stare f_{p0} în G_{p0}

$r(k) \setminus p(k)$	p_{01}	p_{02}
r_1	$\{p_{02}\}$	$\{p_{01}\}$
r_2	$\{p_{02}\}$	$\{p_{01}\}$
r_3	$\{p_{02}\}$	$\{p_{01}\}$

Observații. 1) Se constată că toate traiectoriile continue de stare ale sistemului (8), comandate de $\gamma(r_1) = -1$ și inițializate în zona

$\{x \in \mathbb{R}^2 \mid x^1 + (x^2)^2 / 2 \leq 0\} \cup$
 $\{x \in \mathbb{R}^2 \mid x^1 < 0, x^2 < 0\}$, sunt evoluții continue
 mascate, asociate perechii (p_{02}, r_1) . 2) Obiectivul
 primar de comandă este modelat de evoluția
 discretă dorită $w_p = p_{02}, p_{01}, p_{02}, p_{01}, \dots$,
 necontrolabilă. $\square$

4.3 Al doilea automat SED-proces G_{p1} : nedeterminism clasic

Se consideră un SHC cu procesul continuu (8) și
 elementul de execuție descris în subsecțiunea 4.1.
 Fie mulțimea de funcționale
 $S_h^2 = \{h_i : \mathbb{R}^2 \rightarrow \mathbb{R} \mid h_1(x) = x^1, h_2(x) = x^2\}$.

Fiecare funcțională $h_i \in S_h^2$ separă $\mathbb{R}^2$ în două
 semispații, $H_i^+ = \{x \in \mathbb{R}^2 \mid h_i(x) > 0\}$ și
 $H_i^- = \{x \in \mathbb{R}^2 \mid h_i(x) < 0\}$. Hipersuprafețele
 $\text{Ker}(h_i) = \{x \in \mathbb{R}^2 \mid h_i(x) = 0\}$, $i = 1, 2$, (figura 7)
 sunt drepte și sunt nesingulare. Alfabetul stărilor
 discrete este $P_1 = \{p_1, p_2, p_3, p_4\}$ iar alfabetul
 simbolurilor-proces este $Z_1 = \{z_{1+}, z_{1-}, z_{2+}, z_{2-}\}$, definit ca în figura 8.

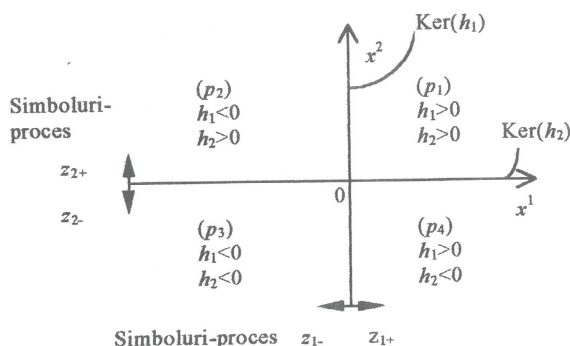


Figura 7. Partiția de stare definită de S_h^2 , cu
 $h_1(x) = x^1$ și $h_2(x) = x^2$, alfabetul P_1 al stărilor
 discrete și simbolurile-proces asociate.

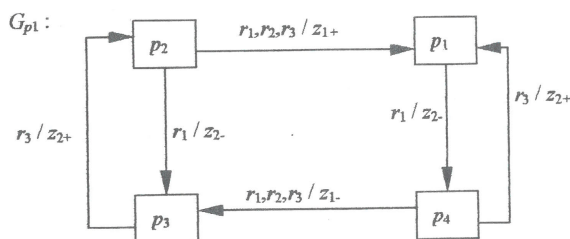


Figura 8. Modelul SED-proces G_{p1} al sistemului
 dublu integrator (8), cu partiția de stare din
 figura 7.

Pe baza portretelor de fază din figura 4 și a
 partiției de stare din figura 7, se deduce modelul
 automat al SED-proces $G_{p1} = \{P_1, R, f_{p1}, Z_1, g_{p1}\}$,
 reprezentat figura 8. Funcția de tranziție de stare
 $f_{p1} : P_1 \times R \rightarrow 2^{P_1}$ este definită în tabelul 2.

Se constată că, la automatul G_{p1} , nu mai apare
 problema evoluțiilor continue mascate din cazul
 precedent, dar G_{p1} este nedeterminist în sensul
 clasic. Cauza tranziției nedeterminate de stare
 $f_{p1}\{p_2, r_1\} = \{p_1, p_3\}$ este explicată, în mod
 intuitiv, în figura 9, unde $h_3(x) = x^1 + (x^2)^2 / 2$.
 De remarcat este faptul că $x^1 + (x^2)^2 / 2 = 0$ este
 ecuația satisfăcută de traiectoria continuă de stare a
 sistemului (8), comandată de $\gamma(r_1) = -1$ și care
 trece prin origine.

Tabelul 2

Funcția de tranziție de stare f_{p1} în G_{p1}

$r(k) \setminus p(k)$	p_1	p_2	p_3	p_4
r_1	$\{p_4\}$	$\{p_1, p_3\}$	-	$\{p_4\}$
r_2	-	$\{p_1\}$	-	$\{p_3\}$
r_3	-	$\{p_1\}$	$\{p_2\}$	$\{p_1, p_3\}$

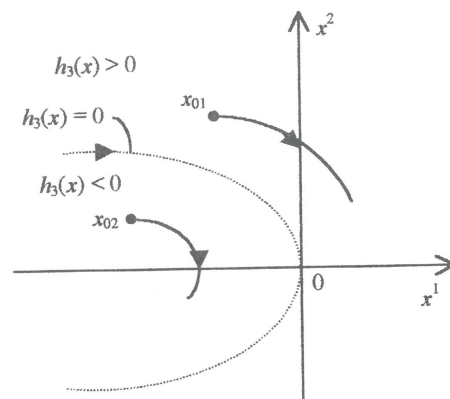


Figura 9. Cauza tranziției de stare
 nedeterminate $f_{p1}\{p_2, r_1\} = \{p_1, p_3\}$, în G_{p1} , cu
 $h_3(x) = x^1 + (x^2)^2 / 2$.

Observație. Obiectivul primar de comandă este
 modelat, în G_{p1} , de evoluția discretă dorită
 $w_p = p_2, p_1, p_4, p_3, p_2, \dots$, controlabilă și care
 poate fi urmărită în mod determinist stabil cu
 ajutorul secvenței de simboluri-comandă
 $w_r = r_3, r_1, r_3, r_1, r_3, \dots$ [4]. $\square$

4.4 Al treilea automat SED-proces G_{p2} : corecția parțială a nedeterminismului clasic

Fie acum, din nou, procesul continuu (8), cu elementul de execuție descris în subsecțiunea 4.1.

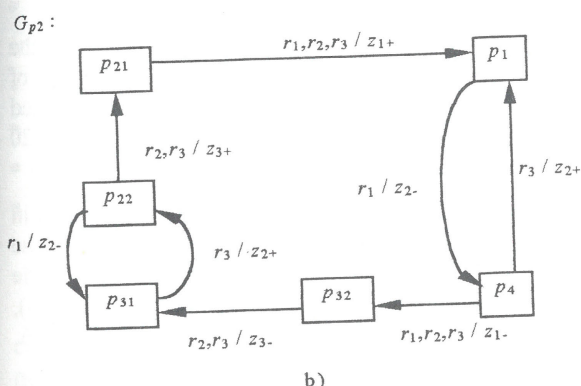
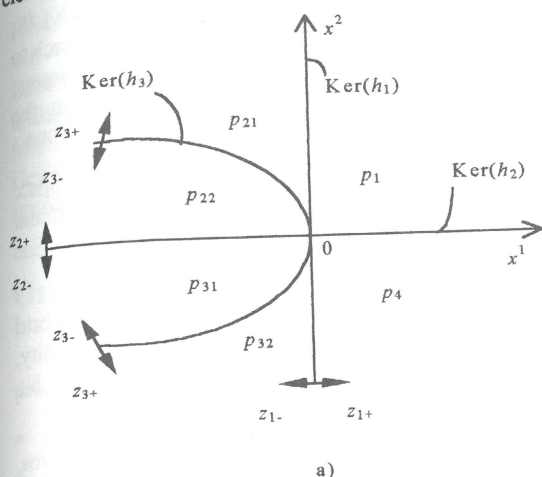


Figura 10. a) Partitia de stare definită S_h^3 , cu $h_1(x) = x^1$, $h_2(x) = x^2$ și $h_3(x) = x^1 + (x^2)^2 / 2$, alfabetul P_2 al stărilor discrete și simbolurile-proces asociate; b) automatul SED-proces G_{p2} .

Pe baza observației din figura 9, vechea partiție de stare, definită de mulțimea S_h^2 (figura 7), este înlocuită cu o nouă partiție, definită de mulțimea de funcționale

$$S_h^3 = S_h^2 \cup \{h_3 : \mathbb{R}^2 \rightarrow \mathbb{R} \mid h_3(x) = x^1 + (x^2)^2 / 2\}$$

(figura 10a).

Automatul SED-proces G_{p2} rămâne nedeterminist, din cauza tranziției discrete de stare $f_{p2}(p_4, r_3) = \{p_1, p_{32}\}$. Acest nedeterminism poate fi corectat doar prin adăugarea, în partiția de stare definită de S_h^3 (figura 10a), a unei noi hipersuprafețe. Noul alfabet al stărilor discrete este $P_2 = \{p_1, p_{21}, p_{22}, p_{31}, p_{32}, p_4\}$ și noul alfabet al simbolurilor-proces este $Z_{21} = \{z_{1+}, z_{1-}, z_{2+}, z_{2-}, z_{3+}, z_{3-}\}$, definit ca în figura 10a. Este evident faptul că, în cadranele II și III ale spațiului de stare $X = \mathbb{R}^2$, partiția definită de S_h^3 (figura 10a) are o granularitate mai fină decât partiția definită de S_h^2 (figura 7). În consecință, numărul stărilor discrete din alfabetul P_2 crește. De asemenea, și complexitatea noului automat SED-proces $G_{p2} = \{P_2, R, f_{p2}, Z_{21}, g_{p2}\}$ (figura 10b) crește, dar tranzițiile din stările p_{21} și respectiv p_{22} sunt acum deterministe (tabelul 3). De remarcat este faptul că celulele asociate stărilor discrete p_{21} și, respectiv, p_{22} acoperă celula corespunzătoare stării discrete p_2 în automatul G_{p1} (figura 7 și figura 10a).

5. Concluzii

În lucrarea de față a fost abordată proprietatea de nedeterminism a SED-proces și implicațiile acesteia, într-o clasă de SHC în care spațiul de stare al procesului, pur continuu, este partiționat de un set finit de hipersuprafețe netede, în celule, iar controlerul este un SED pur.

Dacă automatul SED-proces este determinist, atunci se poate sintetiza un controler SED care să asigure evoluția stării continue, conform oricărei secvențe admisibile de stări discrete dorite, în condițiile în care se cunosc doar celula inițială, ce conține inițializarea stării continue și simbolurile-proces (i.e. evenimentele-proces) generate la traversarea hipersuprafețelor partiției de stare. Aceasta provine din faptul că determinismul automatului SED-proces este direct legat de controlabilitatea evoluțiilor discrete dorite. Dacă automatul SED-proces nu este determinist, atunci este posibil ca unele evoluții admisibile și dorite ale sale să nu fie controlabile; pentru acestea, nu se va

Tabelul 3
Funcția de tranziție de stare f_{p2} în G_{p2}

$r(k) / p(k)$	p_1	p_{21}	p_{22}	p_{31}	p_{32}	p_4
r_1	$\{p_4\}$	$\{p_1\}$	$\{p_{31}\}$	-	-	$\{p_{32}\}$
r_2	-	$\{p_1\}$	$\{p_{21}\}$	-	$\{p_{31}\}$	$\{p_{32}\}$
r_3	-	$\{p_1\}$	$\{p_{21}\}$	$\{p_{22}\}$	$\{p_{31}\}$	$\{p_1, p_{32}\}$

putea sintetiza un controler SED corespunzător, deoarece, chiar dacă secvența de tranziții există, ca atare, totuși secvența de simboluri-comandă ce o controlează poate genera, în mod imprevizibil, și tranziții nedorite. La limită, doar prefixul controlabil al unei evoluții dorite va putea corespunde – dacă el există – comportării în buclă închisă a SED-proces [4].

Transformarea unei comportări nedeterminate într-una deterministă nu este algoritmică și implică: 1) examinarea portretelor de fază ale procesului continuu, trasate, respectiv, pentru fiecare valoare admisibilă și constantă a comenzii generate de elementul de execuție și, apoi, 2) îmbogățirea primei partiții de stare cu noi hipersuprafețe, în scopul reducerii volumului celulelor. Aceasta conduce la: a) o nouă partiție de stare, cu o granularitate mai fină și b) la un nou automat SED-proces, cu o complexitate mai mare decât primul, i.e. cu un număr mai mare de stări discrete. Acest ultim aspect poate deveni un dezavantaj.

SHC sunt, în general, dificil de analizat și de sintetizat, iar obiectivul de comandă are un caracter global, ce referă secvențe dorite de celule. Principalul dezavantaj al formalismelor de SHC bazate pe discretizarea spațiului continuu de stare constă în faptul că, pe de-o parte, traiectoriile continue de stare pot evolua în celule diferite, în funcție de condițiile inițiale, situate într-o celulă dată și, pe de altă parte, controlerul SED observă evoluția procesului doar prin impulsurile generate la traversarea, de către starea continuă, a hipersuprafețelor partiției de stare [6]. Deoarece modelul continuu este determinist, întreaga problemă de analiză a comportărilor dorite se poate, în principiu, rezolva, dacă se cunosc toate familiile de traiectorii de stare, atașate fiecărei valori admisibile a comenzii, în parte (i.e. dacă se pot integra ecuațiile de stare). Această dificultate este specifică și tehnicilor de verificare a proprietăților sistemelor hibride (cum ar fi accesarea unei celule sau locații de către o traiectorie de stare continuă, cu inițializare arbitrară în altă celulă), deoarece trebuie calculate nu o singură traiectorie continuă de stare – chestiune relativ simplă, dacă se utilizează simulatoarele numerice –, ci o întreagă familie de traiectorii, parametrizate cu condiția inițială. O tendință actuală este aceea de a aproxima

ecuațiile continue de stare cu clase de sisteme diferențiale, pentru care se pot calcula exact familiile de soluții [8].

Bibliografie

1. **STIVER, J.A., P.J. ANTSAKLIS, M.D. LEMMON:** A Logical DES Approach to the Design of Hybrid Control Systems. Technical Report of the ISIS Group at the University of Notre Dame, ISIS-94-011, 1994.
2. **STIVER, J.A., P.J. ANTSAKLIS:** State Space Partitioning for a Class of Hybrid Control Systems. În: Proc. of the American Contr. Conf., San Francisco, USA, F, 1993, pp. 2303-2304.
3. **NERODE, A., W. KOHN:** Models for hybrid systems: Automata topologies, controllability, observability. În: Hybrid Systems, LNCS, No 736, 1993, pp. 317-356.
4. **OLTEAN, V. E.:** Contribuții la modelarea, analiza și sinteza sistemelor hibride cu interfață continuu/discret, teză de doctorat, Universitatea "Dunărea de Jos" din Galați, 1998.
5. **OLTEAN, V. E.:** Procedural Aspects in the Design of Hybrid Control systems. În: Proc. of the 3rd int. Conf. on Automation of Mixed Processes ADPM'98, Reims, France, 12-20 March, 1998, pp. 278-286.
6. **OLTEAN, V.E.:** The Observability Concept in a Class of Hybrid Control systems. În: Revue Roumaine des Sciences Techniques, Série Électrotechnique et Énergétique, Tome 44(3), Editura Academiei Române, 1999, pp. 363-376.
7. **RAMADGE, P.J., W.M. WONHAM:** Supervisory control of a class of discrete event systems. În: SIAM Journal of Control Optimization, Vol. 25, Nr. 3, 1987, pp. 206-230.
8. **KROGH, B.H., A. CHUTINAN:** Hybrid Systems: Modeling and Supervisory Control. În: Advances in Control: highlights of ECC'99 (Paul M. Franck ed.), Springer Verlag, 1999, pp. 227-246.