

Modelarea și Verificarea Nodurilor Senzoriale Wireless prin Rețele Petri Hibride Stocastice

Guțuleac E., Gîrleanu I., Ababii V., Hlopanicov C.

Catedra Calculatoare
Universitatea Tehnică a Moldovei
Chișinău, Republica Moldova
egutuleac@mail.utm.md

Abstract — In this paper a Hybrid Stochastic Petri Net unifying framework is presented to provide a wide modeling and verifying of wireless sensor nodes, oriented for representation of the current state evolution of discrete-continuous process energy consumption, data transmission and reception over wireless channels into a time slot of CSMA - CA protocol.

Key Words — Modeling, wireless channels, energy consumption, sensor nodes, protocol, Hybrid Stochastic Petri Net.

I. INTRODUCERE

Progresele recente în domeniul sistemelor de calcul cu microprocesoare și a comunicațiilor fără fir (wireless) au permis dezvoltarea rețelelor cu noduri de senzori (ReNS) ce au mici costuri și diferite aplicații, mai ales, în situații emergente [4, 6]. Cu toate acestea, din cauza mediilor potențial dure, nesigure și dinamice, împreună cu o sursă de energie electrică și lățime de bandă limitată, proiectarea și realizarea ReNS se confruntă cu încă o mulțime de provocări tehnice de poziționare, durată de viață, consumului de energie, rutare a pachetelor de date, precum și cu probleme de securitate [3, 4, 6, 7].

Modelarea formală și tehnicile de verificare sunt utilizate pe scară largă la proiectarea și analiza mai multor tipuri de sisteme cu procese discret-continue, în baza verificării cărora se pot identifica lacunele de proiectare la stadiul incipient al ciclului lor de viață [1, 2]. Ca rezultat, aceste probleme pot fi eliminate mai devreme, iar costurile de depanare, întreținere și mentenanță pot fi reduse în mod semnificativ.

Unul dintre cele mai răspândite formalisme moderne, folosite pentru modelarea și analiza sistemelor paralele/distribuite cu procese discret-continue concurente, sunt rețelele Petri (RP) de diferite extensii, mai ales, RP hibride stocastice (RPHS) [1].

În literatura de specialitate sunt cunoscute doar câteva modele formale de verificare și analiză a rețelelor de senzori ce folosesc diferite abordări cu evenimente discrete [4, 6, 7].

În lucrarea dată este propus un cadru unificator și util pentru modelarea și analiza controlului funcționării corecte a nodurilor senzoriale ale ReNS complexe, orientate pe o descriere adecvată a stărilor discret-continue ale acestora și a regulilor de evoluție pentru actualizarea stărilor curente stabilite prin modele RPHS. Reprezentarea grafică a acestor tip de modele, precum și fundamentarea lor matematică sunt

bine potrivite pentru a reprezenta evoluția stărilor fiecărui nod senzorial (NSM) al ReNS și interacțiunea dintre aceștia prin canale wireless. Un avantaj important al demersului propus constă în faptul că modelul RPHS prezentat în această lucrare este foarte concis și flexibil, deoarece mai multe dintre variabilele sale sunt parametrizate cu atribute cantitative ce pot lua valori alternative dependente de marcajul curent.

II. REȚELE PETRI HIBRIDE STOCASTICE

Rețelele RPHS introduse în [1] permit modelarea, verificarea, simularea vizuală și evaluarea performanțelor sistemelor cu evenimente discret-continue.

Particularitățile RPHS constau în faptul că atât mulțimea locațiilor P constituie o partiție $P = P_d \cup P_c$, $P_d \cap P_c = \emptyset$, unde P_d este mulțimea locațiilor discrete și P_c este mulțimea locațiilor continue (buffere), cât și mulțimea tranzițiilor temporizate T constituie o partiție $T = T_d \cup T_c$, $T_d \cap T_c = \emptyset$, unde T_d este mulțimea tranzițiilor discrete și T_c este mulțimea tranzițiilor continue, care fiind validate ele își schimbă în mod continuu cu viteza $V_k(M, \vec{x})$ nivelul de fluid al bufferelor incidente cu tranziția continuă $u_k \in T_c$ respectivă. Locațiile discrete pot conține un număr întreg de jetoane, iar locațiile continue pot conține o cantitate (mărime reală) de fluid. Locațiile discrete sunt reprezentate grafic prin cerculețe simple, iar cele continue prin două cerculețe concentrice. Tranzițiile discrete temporizate T_τ (*imediate* T_0) sunt reprezentate prin bare negre *groase* (*subțiri*), $T_d = T_\tau \cup T_0$, $T_\tau \cap T_0 = \emptyset$, iar cele continue prin dreptunghiuri mici. Marcajul curent (M, \vec{x}) al RPHS este un vector coloană ce redă distribuția comună a jetoanelor în locațiile discrete și a nivelurilor de fluide în buffere. Partea discretă a acestui marcaj este reprezentată de vectorul $M = (m_i, \forall p_i \in P_d)$ ce conține un număr întreg pozitiv de jetoane $m_i = M(p_i) \in IN_+$ în locațiile discrete p_i , iar $\vec{x} = (x_k, \forall b_k \in P_c)$ este vectorul de marcare continuă a bufferelor în care x_k reprezintă nivelul de fluid în bufferul b_k , care este o mărime reală pozitivă sau negativă. Jetoanele discrete sunt deplasate în conformitate cu

regulile de declanșare ale tranzițiilor imediate $t_j \in T_0$ și a celor temporizate $q_k \in T_\tau$, prin arce discrete.

Reprezentarea grafică a tuturor primitivelor RPHS este dată în figura 1.

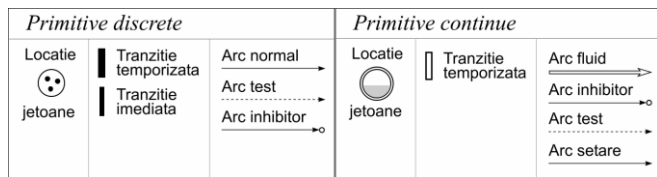


Fig.1. Primitivele unei rețele RPHS.

Duratele de declanșare ale tranzițiilor discrete temporizate sunt definite de *ratele* instantanee de declanșare $\lambda(q, M, \bar{x})$, care pot depinde de marcajul curent al rețelei. Nivelurile de fluid în buffere sunt schimbate fie prin tranziții continue u_j în funcție de viteza lor de declanșare $v(u, M, \bar{x})$, care poate depinde de marcajul curent al rețelei, fie pot fi stabilite direct de un arc de setare la o anumită valoare, atunci când o tranziție discretă se va declanșa. Arcele *inhibitoare* (arcele *test*) *dezactivează* (*activează*) o tranziție atunci când un anumit număr de jetoane sau o anumită cantitate de fluid este prezentă într-o locație discretă sau continuă.

Definirea detaliată a RPHS, principalele atribute cantitative ale RPHS, regulile de validare și declanșare ale tranzițiilor de către marcajul curent sunt prezentate în [1].

III. EXPRESII DESCRIPTIVE ALE RPHS

Pentru a reda proprietăți compoziționale modelelor de rețele RPHS în [1] este introdusă *noțiunea* de *dexel* (*descriptive expression element*) și un set de operații compoziționale care permit de a construi expresii descriptive (DE) ce sunt direct translate grafic în modelul specificat.

Pentru a facilita expunerea lucrării date, în continuare prezentăm succint doar unele operații compoziționale. Mai detaliat cititorul poate consulta lucrările [1, 2].

Un *dexel* hibrid *bHDE* al rețelei subiacente RPHS este redat de expresia:

$$bHDE = \langle \Pi_j |_{\bar{z}_j} m_{oi}^h k_i \bar{y}_i [W_i^+, W_i^-] \Pi_k |_{\bar{z}_k} \rangle,$$

unde $\bar{y} \in \{p, \bar{p}, \tilde{p}, b, \bar{b}, \tilde{b}\}$ este simbolul-locație care reprezintă locația discretă p sau continuă b incidentă la tranziția $\bar{z} \in \{t, u\}$, care este simbolul-tranziție discretă t sau continuă u , ce determină respectiv tipul de arc ($\{p, b$ – arc *normal*, \bar{p}, \bar{b} – arc *inhibitor*, \tilde{p}, \tilde{b} – arc *test* }) cu ponderea respectivă W_i^- incident înainte la tranziția $|_{\bar{z}_k}$, iar W_i^+ este ponderea arcului ce iese din tranziția $|_{\bar{z}_j}$ și intră în locația \bar{y}_i . Atributele locației \bar{y}_i sunt, respectiv: m_{oi} - marcajul inițial; k_i - capacitățile locației. Atributele tranzițiilor \bar{z}_j și \bar{z}_k sunt

respectiv: g_j și g_k - funcția de gardă; Π_j și Π_k - funcția de prioritate. Unele atribute pot fi omise, fiind considerate ca implicit date, de exemplu: $m_{oi} = 0$; $K_{p_i}^{\min} = 0$, iar $K_{p_i}^{\max}$ este considerată ca fiind nelimitată. În cazul în care $W_i^- = W_i^+ = 1$ paranteza pătrată se va omite.

Cu ajutorul diferitor derivate ale *bHDE* și operații *compoziționale*, folosind un raționament adecvat ce redă interacțiunea condițiilor și evenimentelor sistemului specificat, putem compune expresii descriptive ale modelelor RPHS (sub)sistemelor considerate.

O expresie descriptivă a unei rețele $\Sigma \Gamma \zeta$ tip RPHS este: $DE ::= bDE | DE_j * DE_j | \circ DE$, unde $*$ reprezintă operatorul unei *operații binare*, iar \circ - operatorul unei *operații unare*.

Implicit, la aplicarea acestor operații, locațiile și tranzițiile ce au același nume se vor *contopi* în mod respectiv. Într-o DE, orice simbol-locație sau simbol-tranziție poate fi folosit în orice ordine de mai multe ori. Astfel, se va subînțelege că în rețele RPHS respective, redade de expresia DE, ce au aceleași locații (tranziții), cu același simbol vor fi *contopite* respectiv într-un singur *simbol-locație* (*simbol-tranziție*).

Redăm în continuare unele operații compoziționale ce vor fi aplicate în această lucrare:

- *Operația Inhibiție*, redată de operatorul " \bar{p}_i ", este o operație unară. Ea descrie arcul *inhibitor*, redat de dexel-ul $DE1 = m_{oi}^{k_i} \bar{p}_i [W_i] \Pi_k |_{t_k}^{\alpha_k}$ ce descrie o rețea primitivă, constituită din locația p_i și arcul inhibitor ce duce din această locație în tranziția t_k cu atributele respective.

- *Operația Test* unară cu operatorul " \tilde{p}_i ", descrie arcul *test*, redat de $DE2 = m_{oi} \tilde{p}_i [W_i] |_{t_j}^{\alpha_j}$.

- *Operația Sincronizare*, redată de operatorul " \bullet " sau " \wedge ", este o operație binară *comutativă*, *asociativă* și *reflexivă* ce descrie *sincronizarea* pre-condițiilor legate cu p_i incidente înainte unui eveniment t_j , apariția căruia va avea loc numai atunci, când concomitent aceste pre-condiții sunt verificate, fiind descrise de DE3:

$$DE3 = (m_{o1}^{k_1} y_1 [W_1] \bullet \dots \bullet m_{on}^{k_n} y_n [W_n]) |_{t_j} \\ = (\bigwedge_{i=1}^n m_{oi}^{k_i} y_i [W_i]) |_{t_j}$$

- *Operația AND-Split sau Fork*, redată de operatorul " \diamond " sau " $;$ ", descrie faptul că la apariția unui eveniment specificat t_j se vor produce concomitent două sau mai multe post-condiții. Aceasta operație binară, fiind *comutativă*, *asociativă* și *reflexivă*, este redată de următoarea expresia DE4:

$$DE4 = |_{t_j}^{\alpha_j} (m_{o1} p_1 [W_1] \diamond \dots \diamond m_{on} p_n [W_n]) \\ = |_{t_j}^{\alpha_j} ((\diamond)_{k=1}^n m_{ok} p_k [W_k])$$

- *Operația Secvențialitate*, redată de operatorul " $||$ ", este o operație binară ce determină logica "cauză-consecință" a

relației dintre două stări locale p_i (pre-condiție) și p_k (post-condiție), determinată de acțiunea t_j . Această operație, exprimată de expresia $DE5$, este asociativă, reflexivă și tranzitivă, însă necomutativă:

$$DE5 = m_{0i} p_i [W_i] |_{t_j} m_{0k} p_k [W_k] \neq m_{0k} p_k [W_k] |_{t_j} m_{0i} p_i [W_i].$$

Modelarea operației "iterație" poate fi redată prin contopirea locațiilor de la începutul expresiei cu cea de la sfârșitul ei care au același nume (operația "închidere").

• *Operația Paralelism competitiv*, redată de operatorul " ∇ " sau " \mp ", descrie relațiile logice de paralelism competitiv ale condițiilor și evenimentelor între două sau mai multe procese concurente. Ea este aplicată pentru a efectua compunerea modelului din mai multe submodele de subrețele RPHS, ce descriu funcționarea subsistemelor respective, într-un model rezultat al sistemului considerat.

Fie două subrețele N_A și N_B sunt redade de expresiile respective $DE_A = A$ și $DE_B = B$, atunci la compunerea lor prin aplicarea operatorului " ∇ ", relativ la aceste două expresii descriptive, obținem o rețea rezultantă N_R redată de $DE_R = C = A \nabla B$ în care locațiile și tranzițiile ce au același nume vor fi contopite în mod respectiv. Nodurile contopite vor păstra atributele și incidența arcelor din fiecare subrețea. Această operație este comutativă, asociativă și reflexivă.

• *Operația redenumire funcțională*, redată de operatorul " $(Rename(DE_k \setminus \Psi))$ ", permite de a redenumi unele locații, tranziții și/sau atribute ale acestora din DE_k ce aparțin submulțimii de noduri, arce sau marcaje:

$$\Psi = \{t_i \mapsto t_j, p_n \mapsto p_i, x_v \mapsto y_m\},$$

astfel încât acestea pot fi modificate și folosite la descrierea proceselor ce cooperează prin Ψ .

La evaluarea DE se va lua în considerație următoarele priorități de folosire a operațiilor compoziționale: o operație unară leagă mai puternic ca cele binare; • operația " \bullet " și " \diamond " este superioară operației " $|$ ", care, la rândul ei, este superioară operației redenumire funcțională.

În [1] este arătat cum se poate efectua maparea DE a unei RPHS în reprezentare grafică a acestui tip de rețele și invers.

IV. MODELUL RPHS AL UNUI NOD SENZORIAL

O rețea generică de senzori constă dintr-un număr mare de noduri senzoriale (NSM_i , $i = 1, \dots, n_s$) distribuite într-o zonă largă de explorare, care cooperează împreună pentru a recepționa de la senzori săi sau nodurile adiacente și transmite datele către un nod receptor principal ce va produce informații semnificative în conformitate cu protocolul CSMA/CA [4, 6]. De obicei, NSM_i sunt identice care comunică între ele prin canale fără fir (wireless); această caracteristică facilitează proiectarea și analiza funcționării lor. Un NSM_i este compus din următoarele componente principale: o unitate de procesare a datelor relevante; o memorie pentru a stoca programele de sistem și datele intermediare; senzori și actuatori, care sunt o interfață reală cu mediul fizic:

dispozitive care pot observa sau controla parametri fizici ai mediului; un dispozitiv de comunicare pentru transmiterea sau recepția informației printr-un canal wireless, care funcționează într-un mod de excludere reciprocă; o sursă de alimentare disponibilă, adică o anumită formă de baterie electrică este necesară pentru a furniza energie, iar consumul de energie trebuie să fie bine controlat.

La analiza funcționării elementelor fiecărui nod senzorial, capacităților sale (stocare, transmitere date, moduri de operare), limitărilor (baterie, memorie) și a sarcinilor ce trebuie realizate (achiziție, transmitere informații la nodul principal), apar unele probleme de control, verificare și evaluare a performanțelor necesare pentru a fi studiate, de exemplu, controlul alternării între modurile de funcționare.

În continuare funcționarea unui NSM_i este descrisă în mod formal prin RPHS. Acest model este elaborat pentru a studia în viitor, unele probleme menționate mai sus.

Pentru a determina DE respectivă a lui $SG\zeta_i$, ce descrie funcționarea unui NSM_i vom reprezenta stările locale prin simboluri-locații, iar activităților sale prin simboluri-tranziții.

Specificațiile locațiilor discrete și celor continue sunt:

• *locații discrete*: $p_1, (p_6)$ - activarea regimului de recepție (transmitere) a datelor; $p_2, (p_7)$ - recepția (transmiterea) datelor este activată; $p_3, (p_8)$ - recepție (transmitere) date; $p_4, (p_9)$ - recepția (transmiterea) datelor întreruptă; $p_5, (p_{10})$ - permiterea recepției (transmiterii); p_{11} - recepție date de la senzori; p_{12} - noduri senzoriale adiacente libere; p_{13} - cereri recepție (transmitere) date; p_{14} - decizie alocare canal recepție (transmitere) date; p_{15} - cereri anulate; p_{16} - ascultarea canalului; p_{17} - decizie recepție (transmitere); p_{18} - canalul este alocat; p_{19} - resursele canalului sunt disponibile; p_{20} - pachete anulate; p_{21} - succes recepție (transmitere) pachet; p_{22} - anularea recepției (transmiterii) și eliberarea canalului.

• *locații continue*: b_1 - nivelul curent de energie al bateriei; b_2 - volumul curent de date al memoriei.

Specificațiile tranzițiilor discrete și celor continue sunt:

• *tranziții discrete temporizate*: $t_1, (t_7)$ - costul descărcării bateriei la comutarea recepției (transmiterii) datelor; $t_2, (t_8)$ - selectarea regimului de comutarea a recepției (transmiterii) datelor; $t_3, (t_9)$ - permiterea recepției (transmiterii) datelor; $t_4, (t_{10})$ - suspendarea recepției (transmiterii) datelor; $t_5, (t_{11})$ - schimbarea regimului de recepție (transmitere) a datelor; $t_6, (t_{12})$ - permiterea alocării memorie la transmiterea (recepția) datelor; t_{13} - formarea cererilor acces la canal;

t_{18} – ascultarea canalului; t_{23} – recepția (transmiterea) cu succes a pachetului de date; t_{27} – durata slot-ului de timp.

- *tranziții discrete imediate:* t_{14} - inițializarea slot-ului de timp și a cererilor de acces la canal; t_{15} , (t_{16}) – anularea (continuarea) ascultării canalului; t_{19} , (t_{20}) - anularea (continuarea) alocării canalului; t_{21} - anularea transmiterii (recepției) în runde inutile, în care durata lor este mai mare decât cea a slot-ului de timp; t_{22} - detectarea coliziunilor și anularea transmiterii (recepției) datelor; t_{24} , t_{25} , t_{26} , t_{28} - eliminarea jetoanelor din sistem până la terminarea slot-ului de timp. Un slot de timp se termină atunci când se va declanșa tranziția t_{27} .

- *tranziții continue:* u_1 , (u_2) – descărcarea bateriei la comutarea recepției (transmiterii) datelor; u_3 , (u_4) – descărcarea bateriei în stare inactivă (la recepția datelor de la senzori); u_5 , (u_6) – recepția (transmiterea) datelor în (din) memorie; u_7 – încărcarea bateriei.

Controlul recepției datelor este redat de expresia descriptivă $Z^r = \overset{\leftrightarrow}{\bigvee}_{j=1}^3 Z_j^r$, unde:

$$\begin{aligned} Z_1^r &= (b_1[\alpha_1] \cdot p_1) |_{l_1} p_2 \check{\vee} p_{11} |_{l_2} p_1, \\ Z_2^r &= (\tilde{p}_2 \cdot p_5) |_{l_3} (p_3 \diamond p_4) \check{\vee} (p_3 \cdot p_4) |_{l_4} p_5, \\ Z_3^r &= (p_2 \cdot p_3 \cdot p_4) |_{l_5} (p_5 \diamond p_{11}). \end{aligned}$$

Expresia descriptivă Z^r , ce redă controlul transmiterii datelor, este obținută din Z^r folosind operația:

$$Z^r = \text{Rename}(Z^r \setminus \Psi_1), \text{ unde:}$$

$$\Psi_1 = \{t_j \mapsto t_{6+j}, p_j \mapsto p_{5+j}, j = \overline{1,5}\}.$$

Încărcarea și descărcarea bateriei este redată de expresia descriptivă $Z_i^B = \overset{\leftrightarrow}{\bigvee}_{j=1}^2 Z_j^B$, unde:

$$\begin{aligned} Z_1^B &= |_{u_7} b_1[\alpha_2] \check{\vee} (b_1[\alpha_3] \cdot \tilde{p}_2) |_{u_1} \check{\vee} (b_1[\alpha_4] \cdot \tilde{p}_7) |_{u_2}, \\ Z_2^B &= (b_1[\alpha_5] \cdot \tilde{p}_{11}) |_{u_3} \check{\vee} (b_1[\alpha_6] \cdot \tilde{p}_2 \cdot \tilde{p}_7) |_{u_4} b_2[\alpha_7]. \end{aligned}$$

Controlul accesului la memorie și a recepției (transmiterii) datelor în (din) memorie este redat de $Z^M = \overset{\leftrightarrow}{\bigvee}_{j=1}^2 Z_j^M$, unde:

$$\begin{aligned} Z_1^M &= (\tilde{p}_2 \cdot p_5 \cdot \tilde{b}_2) |_{l_6} p_{10} \check{\vee} (\tilde{p}_7 \cdot p_{10}) |_{l_{12}} p_5, \\ Z_2^M &= \tilde{p}_3 |_{u_5} b_2[\alpha_8] \check{\vee} (b_2[\alpha_8] \cdot \tilde{p}_8) |_{u_6}. \end{aligned}$$

Controlul accesului la canal, recepției și transmiterii datelor prin canalul fără fir este redat de $Z^C = \overset{\leftrightarrow}{\bigvee}_{j=1}^5 Z_j^C$, unde:

$$\begin{aligned} Z_1^C &= (p_{13}[m_{13}] \cdot p_{24}) |_{l_{14}} (p_{14}[m_{13}] \diamond p_{19} \diamond p_{23} |_{l_{27}} p_{24}), \\ Z_2^C &= p_{14} |_{l_{16}} p_{16} |_{l_{18}} p_{17} |_{l_{20}} p_{18} |_{l_{21}} p_{20} |_{l_{26}} p_{22}, \\ Z_3^C &= (p_{18}[m_{18}] \cdot p_{19}[m_{19}]) |_{l_{22}} p_{20}[m_{18}] \check{\vee} p_{19} |_{l_{25}} p_{22}, \end{aligned}$$

$$Z_4^C = (p_{18} \cdot p_{19}) |_{l_{23}} p_{21}[2] |_{l_{24}} p_{22} \check{\vee} p_{17} |_{l_{19}} p_{22},$$

$$Z_5^C = p_{14} |_{l_{15}} p_{15} |_{l_{17}} p_{22}[m_{22}] |_{l_{28}} p_{12}[m_{22}] |_{l_{13}} p_{13}.$$

Modelul $\text{SG}\zeta 1$ de rețea RPHS ce descrie funcționarea nodului senzorial *NSM* este redat de expresia descriptivă Z :

$$Z = Z^r \check{\vee} Z^r \check{\vee} Z^B \check{\vee} Z^M \check{\vee} Z^C.$$

Maparea lui Z în $\text{SG}\zeta 1$ este prezentată în figura 2, atributele cantitative explicite ale căreia sunt:

- *Funcțiile de gardă:* $g(u_7) = (x_7 < \gamma)$,
 $g(u_5) = g(u_6) = (m_{18} = 1) \& (m_{18} = 1)$,
 $g(t_{14}) = (m_3 > 0) \& (m_{19} = 0) \& (m_{20} = 0)$,
 $g(t_{22}) = (m_{18} > 1) \& (m_{23} = 1)$, $g(t_2) = (m_{23} = 0)$,
 $g(t_{17}) = g(t_{24}) = (m_{24} = 1)$,
 $g(t_{19}) = (m_{21} = 2) \& (m_{24} = 1)$,
 $g(t_{26}) = (m_{24} = 1) \& (m_{20} > 0)$, $g(t_{28}) = (m_{22} > 0)$,
 • Ratele de declanșare ale tranzițiilor:
 $V(u_i) = v_i, i = 1, \dots, 7$, $\lambda(t_l) = \lambda_l, l = 1, \dots, 12$;
 $\lambda(t_{13}) = m_{12} \cdot \lambda_{13}$, $\lambda(t_{18}) = m_{16} \cdot \lambda_{16}$,
 $\lambda(t_{23}) = \lambda_{23}$, $w(t_j) = \beta_j, j = 15, 16, 19, 20$.

- Capacitățile bufferilor b_1, b_2 sunt h_1, h_2 respectiv.

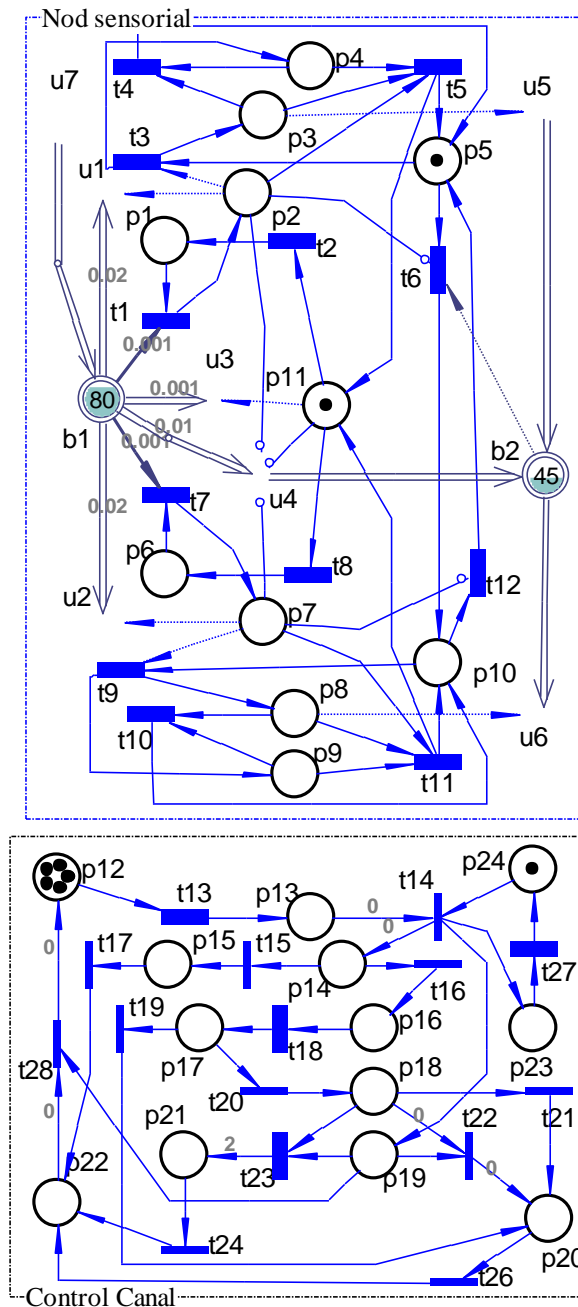
În figura 2 este prezentat modelul $\text{SG}\zeta 1$ de rețea RPHS, obținut în baza expresiei Z , cu marcajul inițial:

$$M_0 = (p_5, p_{11}, 5p_{12}, p_{24}), \bar{x}_0 = (80x_1, 45x_2)$$

Modelul $\text{SG}\zeta 1$ a fost validat și analizat folosind *sistemul software instrumental VHPN* de simulare vizuală a modelelor de tipul RPHS [5]. Această rețea $\text{SG}\zeta 1$ este mărginită, viabilă și reinițializabilă și astfel există un regim staționar de funcționare.

Acest model descrie funcționarea unui nod senzorial, fiind compus din următoarele părți de bază: *senzor* (u_4), *comunicație* ($p_3 - p_5, p_8 - p_{10}, t_3 - t_5, t_9 - t_{11}, u_5 - u_6$), *memorie* (b_2, t_6, t_{12}), *baterie de energie* ($b_1, u_1, u_2, u_7, t_1, t_7, p_1, p_6$) și *control al canalului wireless* ($p_{12} - p_{24}, t_{13} - t_{28}$). Există, de asemenea, trei moduri operaționale cu stările lor locale respective (p_2, p_7, p_{11}). Datele (informațiile), stocate în memoria b_2 , pot sosi de la senzorul însuși u_4 sau de la alți senzori u_5 . Această parte, de asemenea, are un mecanism care poate transmite date doar atunci când în memorie există suficiente informații (p_{10}).

Recepția datelor este permisă doar în cazul în care există suficient spațiu în memorie (p_5).

Fig. 2. Modelul nodului senzorial $S\Gamma C1$ de rețea RPHS.

Partea de comunicație este modelată prin excluderea reciprocă a locațiilor p_2 și p_7 . Atunci când există suficiente informații pentru a fi transmise (redate de ponderea arcului test (b_2, t_6)) nodul senzorial poate să transmită datele sale (p_{10}). Când senzorul are suficient spațiu în memorie, se va marca locația (p_5), indicând faptul că memoria este alocată pentru a primi și stoca datele; tranzițiile continue (u_5) și (u_6) reprezintă transferul de informații de la un nod senzorial la altul. Pentru a descrie acest fapt sunt considerate trei moduri operaționale de bază: *inactiv*, *recepție* și *transmitere*.

Senzorul poate trece liber de la un mod de *funcționare* la altul, dar acest lucru are un cost de comutare în timp și de energie (p_1, p_6 și t_1, t_7). Ratele $v_i, i = 1, \dots, 4$ ale descărcării bateriei b_1 depind de modul curent de funcționare (tranzițiile $u_1 - u_3$) și a duratei de funcționare a senzorului (u_4).

În submodelul ce descrie *controlul canalului wireless*, un slot de timp este reprezentat de tranziția determinist-temporizată t_{27} . Declanșarea tranziției imediate t_{14} indică începutul unui slot de timp, iar ca rezultat avem: (1) sunt depuse în locația p_{14} un număr de jetoane egal cu marcajul locației p_{13} , care denotă faptul că m_{13} noduri pot lua decizii; (2) un jeton în locația p_{19} , care implică faptul că resursele canalului fără fir sunt disponibile; (3) un jeton în locația p_{23} pentru a inițializa mecanismul time-out (t_{14}); La începutul unui slot de timp, nodurile au două opțiuni t_{15} – *anulare* sau t_{16} – *continuare* de a asculta canalul. În cazul în care un nod alege opțiunea *continuare* (un jeton în p_{16}), el va asculta canalul (t_{18}), pentru o durată aleatorie τ_{18} . Vom presupune că τ_{18} are o distribuție exponențială cu parametrul λ_{18} . Această ipoteză este justificată, deoarece ne vom concentra pe o analiză staționară. Parametrul λ_{18} este fixat și el este dat în specificația protocolului și nu poate fi modificat de către utilizatorii obișnuiți. Nodul nu va alege o perioadă de ascultare mai mare decât durata unui slot, adică $\tau_{18} < \tau_{27}$. De asemenea, ascultarea canalului induce un consum de energie, redat de rata tranziției continue u_4 . Dacă canalul este detectat ca fiind inactiv la terminarea τ_{18} , nodul începe să trimită pachete (declanșarea lui t_{20}) și va produce un jeton în locația p_{18} și astfel va valida tranziția temporizată t_{23} – *transmisie cu succes* a pachetului. Durata lui t_{23} este $\tau_{23} = 1/\lambda_{23}$. Declanșarea tranziției t_{23} va produce un jeton obișnuit în locația p_{21} , indicând faptul că un pachet a fost transmis cu succes. În acest moment, dacă un alt nod j a alocat deja canalul, el va renunța (t_{19} se va declanșa) deoarece el poate observa că în acest moment canalul este ocupat. Declanșarea tranziției imediate t_{22} reprezintă situația în care a apărut o coliziune, adică un alt nod trimite un pachet atunci când canalul este deja ocupat și deci în acest caz toate transmisiile de pachete vor fi compromise. Coliziunea de pachete se întâmplă în următorul caz: la terminarea lui τ_{18} , nodul i ascultă canalul dacă el este inactiv și atunci el va trimite pachetele sale. De asemenea, un alt nod j poate detecta în acest moment că canalul este inactiv, adică nodul j nu observă că nodul i trimite deja pachetele sale. Când apare o coliziune, locația p_{18} va conține jetoanele ce aparțin nodurilor în curs de a transmite pachetele lor. Astfel,

va fi validată tranziția imediată t_{22} , care are o prioritate mai mare față de tranziția temporizată t_{21} , iar declanșarea ei va produce în locația p_{20} un număr de jetoane egal cu cel ce se află în locațiile p_{18} și p_{19} . Declanșarea tranzițiilor imediate $t_{24} - t_{26}$, t_{17} , t_{19} și t_{28} elimină jetoanele din sistem și reinițializează transmiterea sau recepția datelor în următorul slot de timp. Aceste tranziții declanșează la terminarea fiecărui slot de timp, rezultând cu depunerea a unui număr respectiv de jetoane în locațiile p_{12} și p_{24} . Un slot de timp se încheie atunci când se va declanșa tranziția temporizată t_{19} .

Funcțiile de gardă ale tranzițiilor sunt utilizate pentru a specifica regulile de declanșare ale tranzițiilor în dependență de starea curentă. Astfel, funcția de gardă asociată cu tranziția t_{19} indică terminarea perioadei de ascultare a canalului în cazul în care p_{17} conține jetoane. În acest caz nodul i trebuie să verifice dacă canalul este ocupat atunci când este terminată ascultarea canalului. Această condiție exprimă faptul existenței a două jetoane în locația p_{21} , ceea ce indică că semnalul de transmisie de la nodul j a sosit și el a fost detectat. În acest caz nodul i trebuie să oprească transmiterea pachetului său și să îl elimine (declanșarea tranziției t_{19}).

Faptul condiționat de $m_{21} = 0$ nu înseamnă neapărat că canalul este inactiv, sunt șanse ca un alt nod j să fie în starea de a emite pachete, însă semnalul de transmisie a acestuia încă nu a ajuns la nodul i . Funcția de gardă a tranziției t_{21} este utilizată pentru a condiționa eliminarea jetoanelor în runde inutile în care durata de declanșare a tranziției t_{18} sau durata de declanșare cumulativă a tranzițiilor t_{18} și t_{23} este mai mare decât a slot-ului de timp τ_{27} .

Menționăm că în acest caz marcajele în p_{20} nu sunt stabile, ele sunt eliminate la declanșarea tranziției imediate p_{26} . Acest fapt face ca la analiza funcționării protocolului să fie omise runde inutile. Funcția de gardă a tranziției t_{22} condiționează declanșarea sa doar înaintea expirării duratei slot-ului, adică când $m_{23} = 1$, altfel jetoanele din p_{18} trebuie să treacă prin tranziția t_{21} .

În figura 3 este prezentată o realizare de simulare a funcționării $ST\zeta 1$ cu următoarele atribute cantitative :

$$\begin{aligned} M_0 &= (p_5, p_{11}, 5p_{12}, p_{24}), \vec{x}_0 = (80x_1, 45x_2); \\ \lambda_i &= \lambda_{6+i} = 10, i = 1, \dots, 4; \lambda_{13} = 0.1; \lambda_{18} = 5; \lambda_{23} = 0.2; \\ \tau_{27} &= 30; \alpha_1 = \alpha_5 = 0,001; \alpha_2 = \alpha_7 = \alpha_8 = 1; \\ \alpha_3 &= \alpha_4 = 0,002; \alpha_6 = 0,02; u_i = 1, i = 1, \dots, 7. \end{aligned}$$

Folosind acest model $ST\zeta_i$ la etapa de proiectare, se poate de determinat parametrii de control ai NSM_i astfel încât să fie asigurată funcționarea dorită în timp real în conformitate cu specificațiile definite de către utilizator.

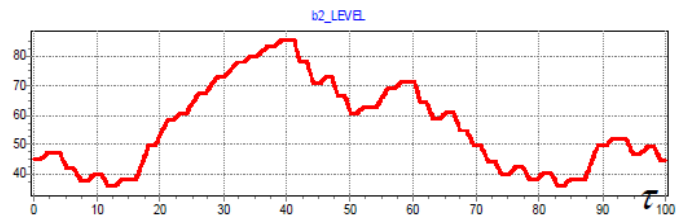


Fig. 3. Evoluția schimbării volumului de date în memoria b_2 a $ST\zeta 1$.

CONCLUZII

În lucrarea dată este prezentat un cadru unificator pentru descrierea funcționării nodurilor senzoriale prin RPHS, orientate pe reprezentarea stărilor curente și a regulilor de evoluție ale procesului discret-continuu de consum al energiei, transmisie- recepție a datelor prin canale fără fir într-un slot al protocolului CSMA/CA. Reprezentarea grafică a modelelor de RPHS și fundamentarea lor matematică sunt bine potrivite pentru a reprezenta evoluția stărilor fiecărui nod senzorial și, de asemenea, interacțiunea dintre aceștia. În plus, cu destul de puține modificări și completări, abordarea descrisă în lucrare poate fi în continuare generalizată pentru a studia sisteme din domenii cu caracteristici de funcționare dinamic reconfigurabile ce sunt condiționate de evenimente specificate la etapa de proiectare.

Lucrarea dată a fost efectuată în cadrul proiectului național de cercetări științifice aplicative 14.820.18.02.03/U.

REFERINȚE

- [1] E. Guțuleac, "Descriptive compositional HSPN modeling of computer systems," Annals of the University of Craiova, România, Series: Automation, Computers, Electronics and Mechatronics, Vol. 3(30), No.2, pp. 82-87, 2006.
- [2] E. Guțuleac, "Descriptive Timed Membrane Petri Nets for Modeling of Parallel Computing," International Journal of Computers, Communications & Control, No. 3, Vol. I, Agora University Editing House, Oradea, România, pp. 33-39, 2006.
- [3] B. Han, D. Z. Zhang and T. Yang, "Energy Consumption Analysis and Energy Management Strategy for Sensor Node," International Conference on Information and Automation, Proceedings of the 2008 IEEE, Vol. 6, pp. 211-214, 2008.
- [4] J. Martyna, "Simulation Study of the Mobility Models for the Wireless Mobile Ad Hoc and Sensor Networks," In A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2012, CCIS 291, Springer-Verlag, pp. 324-333, 2012.
- [5] Iu. Țurcanu, E. Guțuleac, A. Cordonenu, "Sistem Software de Simulare Animată a Rețelelor Petri Diferențiale Reconfigurabile," In Proceedings of the 6-th International Conference ICMCS-2009, vol. 1, 1-3 October, pp. 307-311, 2009.
- [6] J. Yick, B. Mukherjee, D. Ghosal, "Wireless sensor network survey," Computer Networks, 52(12), pp. 2292-2330, 2008.
- [7] H. Zhou, D. Luo, Y. Gao, and D. Zuo, "Modeling of Node Energy Consumption for Wireless Sensor Networks," Wireless Sensor Network, Vol. 3, No. 1, pp. 18-23, 2011.