

RETELELE PETRI COLORATE SI UTILITATEA LOR IN MODELAREA, VALIDAREA SI SIMULAREA SISTEMELOR

*M. Dragan, drd. Ing. Sys.
Universitatea din Bucuresti*

1. INTRODUCERE

Anul de naștere al teoriei rețelelor Petri poate fi considerat 1962, când marele matematician german Karl Adam Petri a prezentat teza sa de doctorat. El a dorit dezvoltarea unei teorii matematice adecvate pentru studierea sistemelor.

Petri a pornit de la observația că în sistemele de tranziție etichetate, ca și în sistemele distribuite, noțiunile de stare și tranziție joacă un rol central – tranziția reprezintă elementul care conduce la schimbarea stării sistemelor.

Încă de la prezentarea acestei teorii interesul studierii ei a fost imens, ceea ce a condus într-un timp scurt aceasta teorie să se dezvolte exponențial, prin apariția a numeroase clase de Rețele Petri.

În jurul anilor '80 apare în cadrul acestei teorii o nouă clasă a rețelelor Petri, clasa Rețelelor Petri Colorate – CPN – care derivă din teoria clasică, dar care se bucură de un mare avantaj, deoarece prezintă un limbaj de programare cu ajutorul căruia se pot modela și simula tot felul de sisteme din viața reală.

Sistemele din lumea reală, adesea, conțin multe părți, care sunt similare, dar nu identice. Folosind rețelele PT, aceste părți trebuie să fie reprezentate de subrețele disjuncte cu o structură aproape identică. Acest lucru înseamnă că per total rețelele PT devin foarte mari. Mai mult decât atât, devine foarte dificil pentru a vedea asemănările și diferențele între subrețele individuale, care reprezintă părți similare.

Utilizarea practică a rețelelor PT pentru a descrie sistemele din lumea reală, a demonstrat în mod clar că este nevoie acută de mai multe tipuri de rețele puternice, pentru a descrie sisteme complexe într-un mod ușor de administrat. Dezvoltarea rețelelor Petri de nivel înalt constituie o îmbunătățire foarte semnificativă în acest sens. Rețelele CP-apartțin clasei de rețele de nivel înalt.

2. PRELIMINARII

Modelarea [1, 3], validarea și simularea sunt necesare atunci când experimentarea directă pe

sistemul real este costisitoare din punct de vedere al cheltuielilor, nu este recomandată sau nu este posibilă.

Sistemul real poate fi utilizat în cadrul unui experiment direct, pe baza unui plan, care prevede diferite posibilități de decizie. După o evaluare corespunzătoare a consecințelor acestor experimente, se poate lua o decizie în privința adoptării celei mai convenabile variante a sistemului cu care se poate lucra.

Definiția 2.1 Modelearea reprezintă o metodă utilizată în știință și tehnică, constând în reproducerea schematică a unui obiect sau sistem sub forma unui sistem similar sau analog în scopul studierii proprietăților și transformărilor sistemului original.

Cel mai reprezentativ model pentru un sistem real poate fi obținut în urma unui proces iterativ, care are ca bază de pornire cel mai elementar model imaginat. Acest deziderat poate fi realizat prin atribuirea limitelor sistemului pe baza cărora să fie luate în considerare numai caracteristicile esențiale în funcție de scopul urmărit, iar pentru satisfacerea cerințelor de validare a modelului se poate încerca relaxarea limitelor sau a unor ipoteze simplificatoare.

În lumea reală, în general, pot fi definite trei tipuri de modele:

analogice (diagramele, graficele, hărțile, vitezometrul, etc.)

fizice (avioane, clădiri, machete de mașini, etc.),

simbolice – reprezintă modelele cele mai abstracte. Cu ajutorul acestor modele toate conceptele sunt exprimate prin variabile definite cantitativ și relațiile sunt reprezentate formal. Astfel modelele simbolice mai sunt cunoscute și sub denumirea de modele matematice sau modele cantitative.

În funcție de tipul datelor utilizate, modelele se pot clasifica în:

Modele deterministe - acele modele, ale căror date sunt considerate a fi cunoscute cu exactitate. În general, modelele deterministe sunt folosite pentru fundamentarea deciziilor interne ale unei organizații pentru alocarea lucrărilor pe diferite puncte de lucru, programarea în timp a unor activități etc.

Soluția unui model determinist va fi întotdeauna aceeași, pentru același set de date de intrare, care sunt cunoscute cu certitudine.

Modele stochastice (probabiliste) – sunt acele modele, ale căror date de intrare nu sunt cunoscute cu exactitate. De cele mai multe ori variabilitatea diferitelor date de intrare poate fi descrisă cu ajutorul unei distribuții de probabilitate. O distribuție de probabilitate poate fi prezentată fie sub forma unei funcții matematice fie sub forma unui tabel, care conține probabilitățile asociate valorilor posibile ale variabilei de intrare.

Analiza deciziilor prin modelare și simulare pleacă pe de o parte de la premisa acceptării limitelor umane de prelucrare a informațiilor și, pe de alta parte, de la considerarea necesității incorporării judecăților și intuițiilor, a rezultatelor imaginației și a creativității decidenților.

Validarea [3] necesită ca orice sistem informatic de calitate, proiectat și implementat profesional, trebuie să fie testat și validat înainte de a intra în faza de producție. Clientul trebuie să fie sigur că sistemul a fost dezvoltat și integrat în conformitate cu specificațiile proiectului. De asemenea, clientul trebuie să se asigure că funcționalitatea proiectului este corectă și fără erori.

După construirea modelului este necesar să se verifice dacă este corect. Procesul de verificare depinde de tipul modelului, dar în principal se verifică dacă modelul corespunde cerințelor pentru care a fost construit.

Validarea este diferită de verificare prin faptul că se stabilește dacă modelul produce rezultate, ale căror valori corespund celor observate în sistemul real. Dacă sistemul considerat nu există se pot utiliza sisteme similare sau se poate apela la experți pentru analiza rezultatelor.

Modelul verificat și validat poate fi apoi utilizat pentru realizarea analizelor de orice tip.

Simularea[1, 2, 3] indică scopul de a găsi „ceva” despre modul de funcționare al sistemului real.

Definiția 2.2 Simularea este procesul, prin care se construiește un model al unui sistem real și se realizează experimente cu acest model în scopul înțelegerii comportamentului sistemului și/sau evaluării diferitelor strategii pentru sistemul analizat.

Definiția 2.3 Simularea este imitarea modului de lucru al unui proces sau sistem real și include generarea unei istorii artificiale și observarea acesteia pentru a realiza inferențe privind caracteristicile funcționale ale sistemului real reprezentat.

Ea este folosită în analiza comportamentului unui sistem, în analize de tip :

„Ce se întâmplă dacă...?” sau în proiectarea unor sisteme reale.

Important este faptul că simularea este în special valoroasă pentru problemele care nu pot fi abordate prin metode matematice analitice sau de optimizare. Deși atât simularea cât și optimizarea sunt metode cantitative care se bazează pe modelele matematice, diferența fundamentală dintre cele două abordări constă în rolul variabilelor de decizie.

Din aceste definiții se pot deduce următoarele ipoteze care stau la baza simulării:

- În general este utilizat un calculator digital pentru realizarea simulării, ceea ce face ca sistemul să poată fi descris în termeni acceptabili pentru un sistem de calcul

- Se poate construi un model matematic și logic al sistemului analizat.

- Simularea reprezintă o tehnică de realizare a experimentelor.

- Sistemul analizat poate fi real sau teoretic.

Datele despre funcționarea sistemului analizat se obțin prin executarea unui model, generându-se astfel o istorie artificială.

3. CONSIDERATII TEORETICE

Explicația informală exprimă în esență, regulile de permisibilitate și de apariție pentru a se înțelege comportamentul unei rețele CP, și poate explica intuitiv rețeaua CP ce poate fi construită. Cu toate acestea, este foarte dificil (probabil imposibil) de a da o explicație informală completă și lipsită de ambiguitate, și, în aceste condiții, este extrem de important ca intuiția să fie completată de definiția formală.

Definiția 3.1[4] Se numește rețea Petri Colorată un tuplu $CPN = (P, T, A, \Sigma, V, C, G, E, I)$ unde :

1. S – reprezintă o mulțime finită, ale cărei elemente se numesc locații (place).

2. T – reprezintă o mulțime finită, ale cărei elemente se numesc tranziții (transition).

Cele două mulțimi trebuie să îndeplinească condițiile : $S \cup T \neq \emptyset$ și $S \cap T = \emptyset$.

3. $A \subseteq S \times T \cup T \times S$ – reprezintă mulțimea arcelor care leagă locațiile de tranziții și tranzițiile de locații.

4. Σ - reprezintă o mulțime finită de seturi de culori, diferită de mulțimea vidă.

5. V – reprezintă o mulțime finită de tipuri de variabile astfel încât $Type[v] \in \Sigma$ pentru toate variabilele $v \in V$.

6. $C : S \rightarrow \Sigma$ - reprezintă funcția setului de culoare care asignează un set de culori la fiecare locație.

7. $G : T \rightarrow EXPRV$ – reprezintă funcția garda care asignează o garda la fiecare tranziție t astfel încât $Type[G(t)] = Bool$.

8. $E : A \rightarrow EXPRV$ - reprezintă funcția expresiei de arc, care asignează o expresie de arc la fiecare arc astfel încât $Type[E(a)] = C(s)_{MS}$, unde s este locația conectată la arcul "a".

9. $I : P \rightarrow EXPR$ – reprezintă funcția de inițializare care asignează o expresie de inițializare la fiecare locație s astfel încât $Type[I(s)] = C(s)_{MS}$.

Definiția formală este cea care a făcut posibilă dezvoltarea metodelor formale de analiză prin care se poate dovedi dacă o anumită rețea CP are anumite proprietăți (de exemplu, absența unor blocaje).

Definiția 3.2[4, 5] Fie $S = \{s_1, s_2, s_3, \dots\}$ o mulțime de locații diferită de mulțimea vidă. Un **multiset** peste mulțimea S este o funcție $m : S \rightarrow \mathbb{N}$ care mapază fiecare element $s \in S$ într-un număr întreg nenegativ $m(s) \in \mathbb{N}$ și definește numărul de apariții al lui s în m .

De asemenea, modelarea, validarea și simularea sistemelor prin intermediul rețelelor Petri Colorate $CPN = (P, T, A, \Sigma, V, C, G, E, I)$, necesită definirea și cunoașterea următoarelor concepte: marcarea, marcarea inițială, variabilele ale tranzițiilor, legătura, pas.

Definiția 3.3[4] Numim **marcare** într-o rețea colorată o funcție $\mu : S \rightarrow \Sigma$ astfel încât pentru oricare locație $s \in S$, $\mu(s) \in C(s)_{MS}$, altfel spus, atribuie fiecărei locații $s \in S$ un multiset peste mulțimea culorilor lui s .

Se notează cu μ_0 marcarea inițială, marcarea ce se obține prin evaluarea expresiilor de inițializare ($\forall (s,c) \in J$ avem $\mu_0(s,c) = I(s)(c)$), $\forall s \in S$).

Notăm cu M mulțimea tuturor marcarilor unei rețele colorate.

Definiția 3.4[4] Se numește marcarea inițială, și se notează cu μ_0 , marcarea definită prin :

$$\mu_0(s) = I(s) \text{ pentru toate locațiile } p \in P.$$

Definiția 3.5[4] Se numesc variabilele ale tranziției $t \in T$, care se notează cu $Var(t) \in V$ și sunt compuse din variabilele care apar în «garda»

tranziției t și în expresiile arcelor conectate la tranziția t .

Definiția 3.6[4] Se numește **legătura** pentru o tranziție t o funcție b definită prin :

a) $\forall v \in Var(t): b(v) \in Type(v)$

b) $G(t) < b >$.

În general, se notează prin $B(t)$ setul tuturor legăturilor pentru tranziția $t \in T$.

Definiția 3.7[4, 5] Numim **pas** într-o rețea colorată un multiset finit și nevid peste mulțimea elementelor de legătură BE .

Notăm cu Y mulțimea tuturor pașilor unei rețele Petri colorate.

Definiția 3.8[4, 5] Se spune că pasul Y este **permis** la marcarea μ dacă și numai dacă pentru oricare locație $s \in S$ avem $\mu(s) \geq \sum_{(t,b) \in Y} E(s,t)(b)$,

unde $E(s,t)(b)$ reprezintă valoarea obținută prin evaluarea expresiei $E(s,t)$ în legătura b .

Dacă $(t,b) \in Y$ spunem că tranziția t este permisă la marcarea μ pentru legătura b .

Dacă $(t_1, b_1), (t_2, b_2) \in Y$ și $(t_1, b_1) \neq (t_2, b_2)$ spunem că (t_1, b_1) și (t_2, b_2) sunt concurenți permisi.

4. DESCRIREREA MODELULUI

Sistemul de modelat este preluat din viața cotidiană și prezintă o societate care are drept activitate efectuarea "Inspekțiilor Tehnice Periodice" pentru toate categoriile de autovehicule.

Pentru mersul corect al modelării și simulării rețelei CPN se compune din următoarele locații și tranziții cu semnificațiile corespunzătoare.

Rețeaua este compusă din 8 locații după cum urmează:

- AS – locație, în care se regăsesc autovehiculele ce urmează a fi verificate la punctul de control dacă au programare de a merge spre locația următoare.

- QA – reprezintă o locație unde autovehiculele formează o coadă de așteptare înainte de a ajunge în locația premergătoare verificării documentelor necesare pentru a putea fi testate.

- VA – este locația, în care autovehiculele sunt în poziția de a aștepta verificarea documentelor.

- AR – reprezintă locația, în care autovehiculele ajung dacă acestea nu au toate documentele corespunzătoare și sunt respinse.

- EI – locația, unde se găsește echipa de ingineri, care preia rând pe rând autovehiculele, care au trecut de verificarea documentelor și urmează să ajungă la standul de probe.

- TT – reprezintă o locație ajutătoare, în care se poate observa timpul aferent testării fiecărui autovehicul.
- AT – reprezintă o locație ajutoare, în care pot fi vizualizate autovehiculele ce au trecut de standul de probe.
- AF – reprezintă locația, în care se regăsesc toate autovehiculele, care au fost testate, precum și timpii petrecuți la standul de probe de fiecare autovehicul în parte.

Evoluția rețelei se bazează pe următoarele 3 tranziții :

- *pc* – este tranziția, prin care autovehiculele sunt verificate dacă au programare la verificări, sunt îndrumate spre coada de așteptare și tot prin această tranziție se realizează și trimiterea acestora în locația tampon premergătoare verificării documentelor.
- *vd* - prin această tranziție sunt îndeplinite acțiunile de verificare a documentelor necesare, dacă acestea corespund cerințelor atunci autovehiculele sunt îndrumate spre locația echipei de ingineri, dacă documentele nu sunt cele cerute atunci autovehiculele sunt respinse, ceea ce face ca aceste autovehicule să ajungă în locația AR.
- *ta* – tranziția, prin care se produce testarea autovehiculelor și are drept finalitate stocarea timpilor necesari pentru testarea fiecărui autovehicul în locația timpi testare - TT, depozitarea numelui fiecărui autovehicul testat în locația autovehicule testate (AT) și nu în ultimul rând formarea listei autovehiculelor testate și timpii petrecuți de acestea la standul de probe în locația autovehicule finalizate (AF).

Familia mulțimilor de culori F ale rețelei care modelează activitatea de testare a autovehiculelor are următoarea componență :

$F = \{ INT, DATA, INTxDATA, S, V, T \}$, în care:

- $INT = \{ int\ timed \}$, mulțimea de culori, din care iau valori toți întregii pentru sincronizare.
- $DATA = \{ string \}$, mulțimea de culori, în care se regăsesc valori de tip șir de date și cu care se colorează locațiile QA, AR și AT.
- $INTxDATA = \{ product\ INT*DATA \}$, mulțimea de culori rezultată prin produsul cartezian dintre mulțimea de culori INT și mulțimea de culori DATA. Multime de culori, cu care se colorează locațiile AS, VD și EI.
- $S = \{ intregi\ de\ la\ 0\ la\ 10 \}$, mulțime de culori, din care iau valori variabilele x .
- $V = \{ intregi\ de\ la\ 0\ la\ 10 \}$, mulțime de culori, din care iau valori variabilele y .
- $T = \{ intregi\ de\ la\ 0\ la\ 1000\ pentru\ sincronizare \}$, mulțime de culori, cu care se colorează locațiile TT și AF.

Expresiile de pe arce sunt multiseturi constante precum 1^n , variabile cum ar fi a, x, y ce iau valori în mulțimea de culori ce reprezintă întregii, șir de date sau condiții logice, prin care ponderăm arcele cu multiseturi peste mulțimile de culori prezentate mai sus sau cu multisetul vid "empty".

5. DESCRIEREA EVOLUȚIEI REȚELEI MODELATOARE

Întreaga evoluție a rețelei are la bază activitatea de revizie tehnică a autovehiculelor (ITP). Pentru a efectua aceasta activitate societatea efectuează alte demersuri necesare bunului mers al fluxului de lucru. Astfel pentru încadrarea în timpul de lucru și eliminarea timpilor morți se realizează o programare a autovehiculelor, în așa fel încât numărul acestora să nu fie mai mare de cel estimat a putea fi prelucrat și a se încadra în graficul orar al unității service. Programul de lucru este considerat a fi cuprins între orele 8.00 AM și 20.00 PM, activitate ce se desfășoară în două schimburi, câte 6 ore de lucru pentru fiecare schimb.

Autovehiculele programate apar într-o ordine aleatoare la punctul de control – tranziția *pc* – unde acestea sunt verificate dacă au programare, după care sunt trimise în locația (QA). De aici primul autovehicul programat merge prin realizarea tranziției *pc* în locația (VD), unde urmează să fie preluat pentru verificarea documentelor. Această acțiune se realizează prin producerea tranziției *vd* astfel încât acele autovehicule, ale căror acte nu corespund cerințelor impuse de lege sunt respinse și trimise în locația autovehicule respinse (AR), iar autovehiculele care corespund cerințelor sunt trecute spre locația, unde se află echipa de ingineri, de unde sunt preluate și prin producerea tranziției testare autovehicule *ta* acestea sunt supuse probelor conform normelor în vigoare și rezultatele producerii acestei tranziții sunt depuse în trei locații diferite:

- Timpii ceruți pentru testarea unui autovehicul sunt depuși în locația Timpi Testare (TT).
- Autovehiculele testate sunt trecute în ordinea testării lor în locația autovehicule testate (AT).
- În locația Autovehicule Finalizate (AF) sunt depuse rezultatele testelor finale, adică numărul autovehiculelor testate împreună cu timpii aferenți fiecărei testări.

Cum producerea fiecărei tranziții este calculată în mod exponențial și timpii de lucru diferă considerabil de la autovehicul la autovehicul.

Conform cu cele specificate până aici considerăm locațiile date în ordinea dată de următorul vector :

$$\{AS, QA, VD, AR, EI, TT, AT, AF\}$$

Începerea întregii activități a societății pornește de la următoarea marcăre inițială :

$$\mu_0 = \left(\left(22(I, "a1", @0+++), \dots, I, "a22"@0 \right), \left(I^{*****}, 0, I^{*****}, 0, 0, I^{*****}, I1@0 \right) \right),$$

în care 0 reprezintă setul vid, iar “****” reprezintă șirul vid, iar această marcăre specifică faptul că urmează să între la teste 22 de autovehicule, care se află în zona de așteptare pentru autovehicule.

La acest moment singura tranziție posibilă este tranziția *pc*, prin care autovehiculelor li se permite pătrunderea în coada de așteptare, adică în locația QA, așa cum se poate observa în următoarea fig. 1.

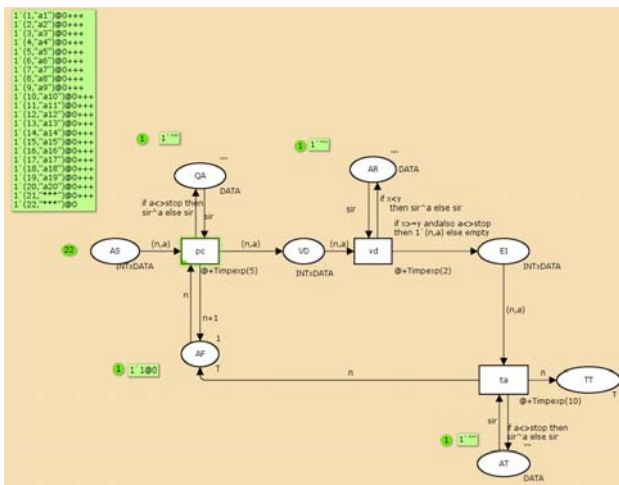


Figura 1.

Prin producerea tranziției punct control *pc* se obține următoarea marcăre:

$$\mu_1 = \left(\left(21(I, "a2", @0+++), \dots, I, "a22"@0 \right), \left(I^{*****}, I^{*****}, I^{*****}, 0, 0, I^{*****}, I1@0 \right) \right),$$

care indică faptul că la acest moment al evoluției rețelei se pot produce tranzițiile *pc* și *vd* – ducând rețeaua într-o nouă stare. Se observă că autovehiculul *a1* a trecut prin locația coadă autovehicule (QA) și a fost, de asemenea, deplasat la locația verificare documente (VD) (fig. 2).

La marcărea μ_2 sunt permise tranzițiile punct control *pc* și testare autovehicule *ta*, care se pot produce concurrent. Prin producerea tranziției testează autovehicule *ta* rețeaua trece în următoarea configurație (fig. 3):

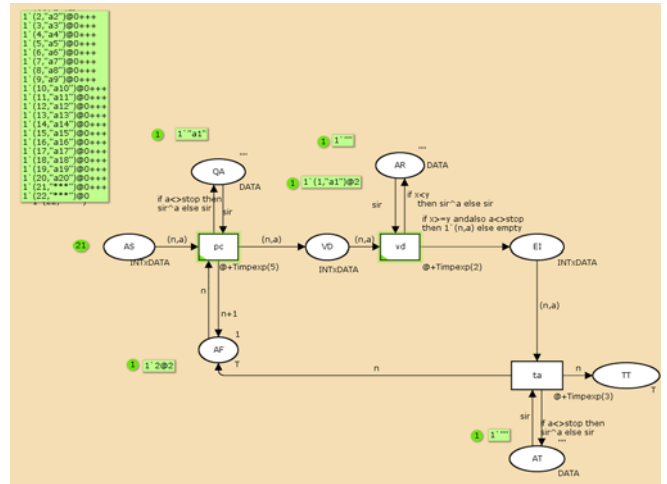


Figura 2.

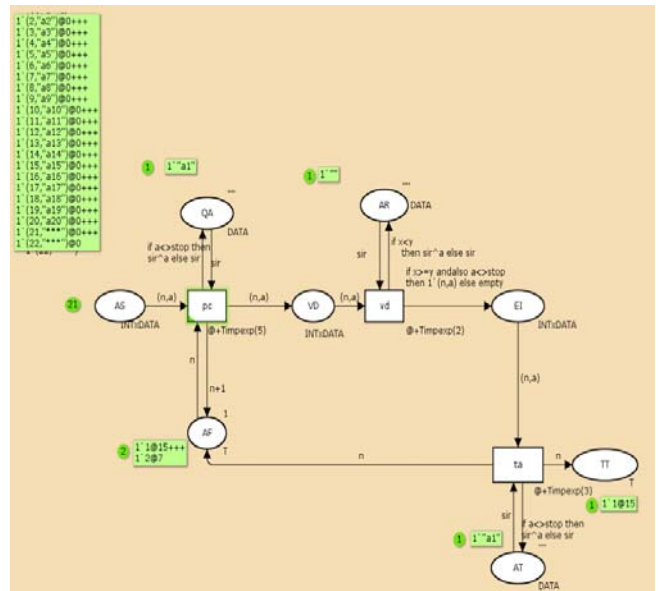


Figura 3.

Se continuă analog producându-se tranzițiile corespunzătoare acelor acțiuni ce indică starea reală din teren. Prin producerea acestor tranziții se obține de fiecare dată marcărea care precizează starea efectivă a activității. Se observă cu ușurință că la orice moment se pot obține toate datele despre evoluția rețelei corespunzătoare aceluși moment.

Cu instrumentul CPN-Tools componenta timed [3, 6] am obținut graful de acoperire ce are în noduri stările intermediare, prin care trece rețeaua în funcționarea ei, pornind de la marcărea inițială μ_0 , iar pe arce are tranziția cu legăturile sale pentru șir, *a*, *n*, *x* și *y*, care prin producere realizează o trecere de la nodul sursă la nodul destinație al fiecărui arc în parte.

Astfel prin producerea tranziției testează autovehicule *ta* se poate și că primul autovehicul a

fost testat, timpul necesar testării a fost de 15 minute și că urmează să între pe fluxul de lucru autovehiculul “a2”.

Rețeaua colorată, care modelează activitatea societății, a fost desenată prin intermediul instrumentului CPN-Tools [6], instrument furnizat și licențiat de grupul de lucru CPN, colectiv al Universității Aarhus din Danemarca.

Tot cu ajutorul instrumentului CPN-Tools a fost generat și graficul de acoperire, care conține în noduri marcările corespunzătoare tuturor stărilor prin care a trecut activitatea societății până la un moment dat. O primă parte a acestui graf este prezentată în fig.4.

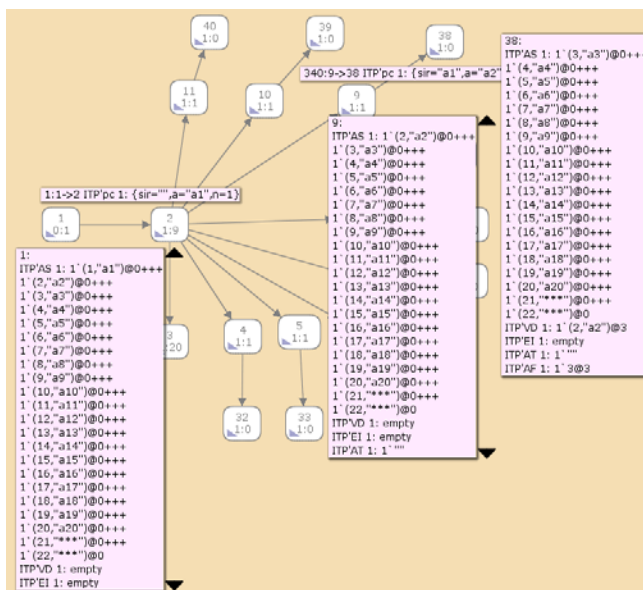


Figura 4.

În acest graf evidențiem marcarea inițială μ_0 , una din marcările terminale μ_{38} , precum și legăturile pe arce de la marcarea inițială la cea finală, care trec prin marcările:

$$\mu_0 \rightarrow \mu_2 \rightarrow \mu_9 \rightarrow \mu_{38} .$$

Pe acest arbore se poate vedea starea rețelei la fiecare moment prin analiza marcării corespunzătoare aceluși moment, așa cum ar fi, spre exemplu, stările rețelei corespunzătoare la marcarea intermediară μ_9 .

De asemenea, tot cu ajutorul instrumentului CPN am obținut și graficul din fig. 5:

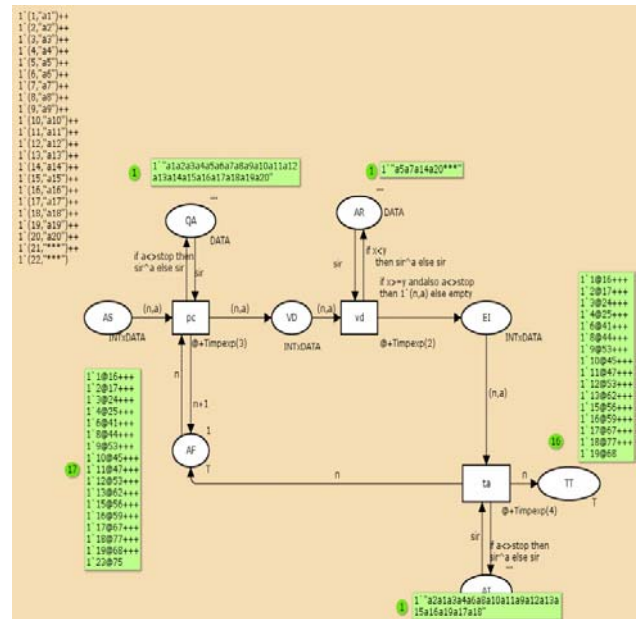


Figura 5.

Din această figură putem observa și interpreta întreaga activitate a rețelei, astfel:

- 1) Din diverse motive, acte necorespunzătoare spre exemplu, 4 autovehicule au fost respinse, urmând să revină la momentul la care au toate documentele în ordine.
- 2) Au fost testate 16 autovehicule, din care primele 4 au avut timpi de lucru mai reduși, aceasta deoarece făceau parte din categoria autoturismelor, iar restul de 12 autovehicule au necesitat un timp mai îndelungat pentru a putea fi testate, ele făcând parte din randul autovehiculelor de mare tonaj, ceea ce solicita teste mai complexe.

Folosind o asemenea modelare și simulare orice firmă poate acționa pe baza rezultatelor obținute în acele zone, în care pot aduce îmbunătățiri substanțiale întregii activități de producție.

Tot cu CPN-Tools am obținut și raportul din figura 6, pe care observăm informații despre spațiul stărilor rețelei, despre valorile minime și maxime atinse în locațiile rețelei în funcționarea ei, despre marcări capcană (home), despre numărul marcărilor moarte, despre tranziții viabile și tranziții moarte etc.

```

Boundedness Properties
-----
Best Integer Bounds
      Upper      Lower
ITP'AF 1      17      17
ITP'AR 1      1       1
ITP'AS 1      0       0
ITP'AT 1      1       1
ITP'EI 1      0       0
ITP'QA 1      1       1
ITP'TT 1     16      16
ITP'VD 1      0       0
Best Upper Multi-set Bounds
ITP'AF 1      1'1++
ITP'AR 1      1'"a5a7a14a20****"
ITP'AS 1      empty
ITP'AT 1      1'"a2a1a3a4a6a8a10a11a9a12a13a15a16a19a17a18"
ITP'EI 1      empty
ITP'QA 1      1'"a1a2a3a4a5a6a7a8a9a10a11a12a13a14a15a16a17a18a19a20"
ITP'TT 1      1'1++
ITP'VD 1      empty
Best Lower Multi-set Bounds
ITP'AF 1      1'1++
ITP'AR 1      1'"a5a7a14a20****"
ITP'AS 1      empty
ITP'AT 1      1'"a2a1a3a4a6a8a10a11a9a12a13a15a16a19a17a18"
ITP'EI 1      empty
ITP'QA 1      1'"a1a2a3a4a5a6a7a8a9a10a11a12a13a14a15a16a17a18a19a20"
ITP'TT 1      1'1++
ITP'VD 1      empty

Home Properties
-----
Home Markings
  Initial Marking is a home marking
Liveness Properties
-----
Dead Markings
  All
Dead Transition Instances
  All
Live Transition Instances
  None
Fairness Properties
-----
  No infinite occurrence sequences.
    
```

Figura 6.

6. CONCLUZII

Modelarea și simularea pot contribui la înțelegerea și îmbunătățirea unui sistem real. Cu toate că un sistem poate fi extrem de complex, este bine să se încerce să se construiască un model cât mai simplu posibil. Acesta se obține atât prin definirea limitelor sistemului analizat astfel încât să fie luate în considerare numai caracteristicile esențiale din punct de vedere al obiectivului analizei, cât și prin definirea unor ipoteze simplificatoare.

Modelul poate fi îmbunătățit prin redefinirea limitelor și prin relaxarea ipotezelor. Pe de alta parte, dacă se încearcă includerea în model a tuturor factorilor și relațiilor, modelul ar putea deveni prea complicat pentru a fi rezolvat. De aceea, este necesar să se realizeze un compromis între necesitatea de a construi un model simplu și ușor de rezolvat și necesitatea de a obține prin

model o reprezentare rezonabilă și plauzibilă a problemei reale.

Prin modelarea și simularea acestui sistem s-a dorit scoaterea în evidență a puterii de modelare a rețelelor Petri colorate stochastic temporizate. Acesta reprezintă modalitatea concretă, prin care pot fi studiate sisteme reale oricât de complexe.

Se observă că modelul poate fi extins cu ușurință la orice sistem cu cozi de așteptare oricât de complexă ar fi topologia aceluși sistem. În particular poate fi extins la rețele de calculatoare cu cozi de așteptare de tip JACKSON în serie, în paralel, cu feedback etc.

Bibliografie

1. Gutuleac, E. *Modelarea și Evaluarea Performanțelor Sistemelor de Calcul prin Rețele Petri*, Dep. Ed. Poligrafic al UTM, Chisinau, 1997.
2. Moore, J.H., Weatherford, L.R. *Decision Modeling with Microsoft Excel*, 6th ed. Prentice-Hall, Inc., Upper Saddle River, pag. 10-11, 2001
3. *Marele Dictionar Ortografic al Limbii Romane*, Editura Academiei Romane, 2008.
4. Jensen, K. *Coloured Petri Nets, Vol 1,2,3* Springer-Verlag Berlin Heidelberg, 1992.
5. Jensen, K., Kristensen, L. M. *Coloured Petri Nets-Modelling and Validation of Concurrent Systems*, Springer-Verlag Berlin Heidelberg, 2009.
6. <http://www.cs.au.dk/CPnets/CPN-Tools>, 2009.