საქართველოს ტექნიკური უნივერსიტეტი

გ. გოგიჩაიშვილი, გ. სურგულაძე,
დ. გულუა, მ. კაშიბაძე

ავტომატიზებული მართვის მოდელები: პეტრის ქსელები

დამტკიცებულია:
სტუ-ს სარედაქციო-
საგამომცემლო
საბჭოს მიერ

თბილისი – 2005

თავი I

პეტრის ქსელები – მიმოხილვა და ახალი ამოცანები

1.1. პეტრის ქსელების თეორიული საფუძვლები

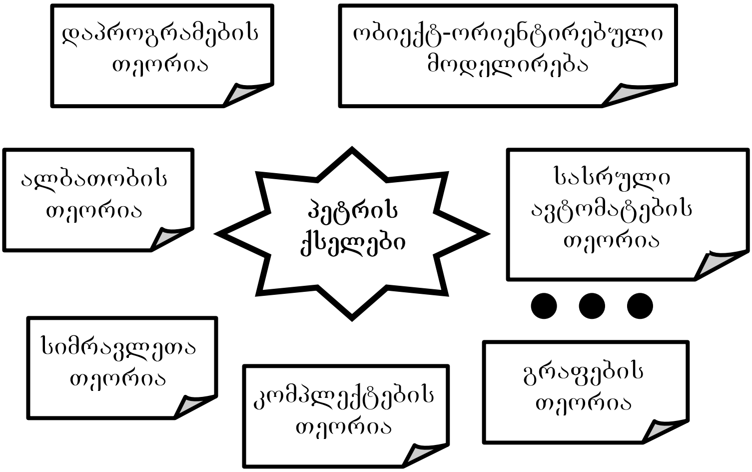
პეტრის ქსელების ისტორია 1962 წლიდან იწყება, როცა გერმანელმა ინჟინერმა, კარლ ადამ პეტრიმ დარმშტადტის ტექნიკურ უნივერსიტეტში დაიცვა სადოქტორო დისერტაცია თემაზე "კომუნიკაცია ავტომატებით".

ამ ნაშრომში მან პირველად ჩამოაყალიბა და დაასაბუთა იდეა ორი განსხვავებული ტიპის კვანძებითა და მათი დამაკავშირებელი მიმართული რკალებით აგებული მუშა ქსელების შესახებ, რომლებიც ერთი მოდელის ფარგლებში გააერთიანებდა კონკრეტულ და აბსტრაქტულ პროცესებს და მონაცემებს.

სახელი "პეტრის ქსელები" თეორიამ მოგვიანებით მიიღო ავტორის პატივსაცემად. თავისი თეორიის საფუძვლად პეტრიმ სასრული ავტომატების, სიმრავლეთა და გრაფების თეორიის ელემენტები გამოიყენა.

არსებობის 44-წლიანი ისტორიის მანძილზე გაიბა კავშირები პეტრის ქსელებსა და სხვა მრავალ მათემატიკურ თუ არამათემატიკურ დისციპლინებთან.

ერთის მხრივ პეტრის ქსელები ფართოვდება თეორიულად, სულ უფრო მძლავრი ხდება მისი მათემატიკური აპარატი, იქმნება ახალი თეორიული კლასები, ხოლო მეორეს მხრივ მატულობს პეტრის ქსელების პრაქტიკული გამოყენების სიხშირე ინფორმატიკის მოწინავე მიმართულებებთან მჭიდრო თანამშრომლობის შედეგად, რაც სქემატურად 1.1 ნახაზზეა გამოსახული.



ნახ.1.1. პეტრის ქსელები და მისი გარემოცვა

1.1.1. სიმრავლეები

სიმრავლეთა თეორია პეტრის ქსელების ერთერთი ბაზისია. განვიხილოთ მოკლედ მისი ძირითადი ელემენტები.

საწყისი აღნიშვნები:

N = {0,1,...} – ნატურალურ რიცხვთა სიმრავლე;

Z = {…,-1,0,1,…} – მთელ რიცხვთა სიმრავლე

Boolean = {true,false} – ბულის სიმრავლე

სიმრავლე არაერთგვაროვან ობიექტთა ერთობლიობაა. მათ სიმრავლის **ელემენტები** ეწოდება.

a არის **A**-სიმრავლის ელემენტი, თუკი ფლობს თვისებას **a∈A** („მიეკუთვნება“). სიმრავლე მოიცემა ელემენტთა ჩამონათვალით $\mathbf{A = \{a_1, a_2 ,..., a_n\}}$, ან გარკვეულ **p(a)** ფუნქციაზე დაყრდნობით, რომლის შედეგი სიმრავლის ელემენტისთვის აუცილებელ პირობას აკმაყოფილებს:

$$\mathbf{A = \{a \mid p(a)\}.}$$

ცარიელი სიმრავლე **Ø**-სიმბოლოთი აღინიშნება და გამოისახება პირობით $\mathbf{Ø = \{a \mid a \neq a\}}$, რადგან პირობა $\mathbf{a \neq a}$ ყოველთვის მცდარია.

სიმრავლეთა თეორიაში განისაზღვრება შემდეგი ძირითადი დამოკიდებულებები და ოპერაციები: **ქვესიმრავლე ($A \subseteq B$), ჭეშმარიტი ქვესიმრავლე ($A \subset B$), გაერთიანება ($A \cup B$), თანაკვეთა ($A \cap B$), სხვაობა ($A \setminus B$),** სადაც:

$\mathbf{A \subseteq B}$, როცა ნებისმიერი $\mathbf{a \in A}$–თვის მართებულია $\mathbf{a \in B}$

$\mathbf{A \subset B}$, როცა $\mathbf{A \subseteq B}$ და $\mathbf{A \neq B}$

$\mathbf{A \cup B = \{ a \mid a \in A}$ ან $\mathbf{a \in B \}}$

$\mathbf{A \cap B = \{ a \mid a \in A}$ და $\mathbf{a \in B \}}$

$\mathbf{A \setminus B = \{ a \mid a \in A}$ და $\mathbf{a \notin B \}}$

ცარიელი სიმრავლე ნებისმიერი არაცარიელი სიმრავლის ქვესიმრავლეა: $\mathbf{Ø \subset A}$. **A** და **B** სიმრავლეებს **განცალკევებული სიმრავლეები** ეწოდება, თუ $\mathbf{A \cap B = Ø}$.

სიმრავლე შეიძლება შეიცავდეს ელემეტებს, რომლებიც თავადაა სიმრავლეები. **A**-სიმრავლის ყველა შესაძლო ქვესიმრავლეთა სიმრავლე $\mathcal{P}(\mathbf{A})$-თი აღინიშნება, ნატურალურ რიცხვთა სიმრავლე 0-ის ჩათვლით – **N**-ით, ლოგიკურ მნიშვნელობათა (ჭეშმარიტი ან მცდარი) სიმრავლე – **B**-თი.

$\mathbf{A_1, A_2, \ldots, A_n}$ ($\mathbf{n \in N}$) სასრულ სიმრავლეთა **პროდუქტი (დეკარტული ნამრავლი)** განისაზღვრება შემდეგნაირად:

$$\mathbf{A_1 x A_2 x \ldots x A_n = \{ (a_1, a_2, \ldots, a_n) \mid a_i \in A_i \; i=1,\ldots,N \}}$$

პროდუქტის ცალკეულ ელემენტს **n**-კორტეჟი ეწოდება. ყოველი **i**-სთვის, სადაც $\mathbf{1<i<n}$, $\mathbf{a_i}$-ს კორტეჟის **i**-ური ელემენტი ეწოდება $(\mathbf{a_1, \ldots, a_n})$. **წყვილი** განისაზღვრება, როგორც **n**-კორტეჟის კერძო შემთხვევა, 2-კორტეჟი (ბინარული).

თუ ყველა სიმრავლე $\mathbf{A_1 = A_2 = \ldots A_n = A}$ მსგავსია, პროდუქტი ჩაიწერება $\mathbf{A^n}$ სახით. გარდა ამისა, $\mathbf{A^1 = A}$ და $\mathbf{A^0 = Ø}$.

1.1.2. მულტისიმრავლეები (კომპლექტები)

მულტისიმრავლე განისაზღვრება სიმრავლეში ერთი და იმავე ელემენტის რამდენჯერმე ასახვისთვის. მაგალითად, პეტრის ქსელის პოზიციაში რამდენიმე მსგავსი მარკერის აღსაწერად.

მულტისიმრავლე B არაცარიელ საბაზო A სიმრავლეზე, ეწოდება ფუნქციას:

$$\mathbf{B:A \to N}$$

სადაც საბაზო **A** სიმრავლის ყოველი **a∈A** ელემენტის **სიხშირე B** მულტისიმრავლეში აისახება ფორმატით **B(a).** სიხშირის სიდიდე შეიძლება 0-ის ტოლიც იყოს. სიმრავლე მულტისიმრავლის სპეციალური შემთხვევაა, სადაც სიხშირის მნიშვნელობებია 0 ან 1.

მულტისიმრავლის ასახვის გაფართოებული ფორმა შემდეგია: **[a,a,...,a,b,...,b,...],** სადაც ყოველი ელემენტი თავისი სიხშირის მიხედვით მეორდება.

მულტისიმრავლეთა სიმრავლე საბაზო **A** სიმრავლეზე აღინიშნება **μA**-თი. მულტისიმრავლე ასევე შეიძლება გამოისახოს სიმბოლური ჯამის სახით, რომელიც **a∈A** ელემენტის სიხშირეს და სახელს შეიცავს:

$$B = \sum_{a \in A} B(a)a$$

თუ **B(a) = 1,** მაშინ ჯამურ ასახვაში იგი საერთოდ გამოიტოვება და იწერება მხოლოდ **a**.

B∈μA მულტისიმრავლეში**, a∈A** ელემენტს ეწოდება **B-**ს წევრი და ჩაიწერება **a∈B,** თუ **B(a)>0** და პირიქით, თუ **B(a)<0,** მაშინ **a∉B.** ცარიელი მულტისიმრავლე **∅** წევრებს არ შეიცავს: **∀a∈A, ∅(a)=0.**

მულტისიმრავლის **სიმძლავრე (კარდინალურობა)** მისი ყველა ელემენტის **სიხშირეთა ჯამს** ეწოდება და განისაზღვრება შემდეგნაირად:

$$|B| = \sum_{a \in A} B(a)$$

თუ **|B|** სასრულია, მაშინ მულტისიმრავლე **B**-ს **სასრული მულტისიმრავლე** ეწოდება.

ორი მულტისიმრავლე, $\mathbf{B_1}$ და $\mathbf{B_2}$ **ტოლია** $(\mathbf{B_1=B_2})$, თუ $\forall \mathbf{a \in A, B_1(a)= B_2(a)}$.

$\mathbf{B_1}$ **ნაკლებია ან ტოლია** $\mathbf{B_2}$-ის (ანუ $\mathbf{B_2}$ მოიცავს $\mathbf{B_1}$-ს) თუ $\forall \mathbf{a \in A, B_1(a) \leq B_2(a)}$.

მულტისიმრავლეებზე ძირითად ოპერაციებს წარმოადგენს:

შეკრება: $\mathbf{B=B_1+B_2}$, თუ $\forall \mathbf{a \in A, B(a) = B_1(a) + B_2(a)}$

გამოკლება**:** $\mathbf{B=B_1\text{-}B_2}$, თუ $\forall \mathbf{a \in A, ((B_1(a) \geq B_2(a)) \wedge ((B(a) = B_1(a) - B_2(a))}$

სკალარული ნამრავლი: მულტისიმრავლის $\mathbf{B_1 \in \mu A}$ და ნატურალური რიცხვის $\mathbf{n \in N}$ სკალარული ნამრავლი განისაზღვრება როგორც $\mathbf{B=n*B_1}$, თუ $\forall \mathbf{a \in A, B(a) = n*B_1(a)}$, სადაც **"*"** არითმეტიკული გამრავლების ოპერაციაა**.**

1.1.3. პეტრის ქსელების ძირითადი ცნებები

პეტრის ქსელების ნოტაციას საბოლოო სახე ჯერ არ მიუღია. მისი მრავალმხრივი განვითარების პროცესში წარმოიშვა შეუსაბამისობების მთელი რიგი, რაც ნოტაციის სხვადასხვანაირ ინტერპრეტირებას უკავშირდებოდა.

პეტრის ქსელების სტანდარტიზაციის პროცედურა ეგრეთ წოდებული **პროექტ 15909-**ის ფარგლებში 1995 წლიდან მიმდინარეობს. მასში მონაწილეობენ ისეთი ავტორიტეტული ორგანიზაციები, როგორიცაა **სტანდარტიზაციის საერთაშორისო ორგანიზაცია (ISO)**, **ინფორმაციულ ტექნოლოგიათა ერთიანი ტექნიკური კომიტეტი (JTC1) და საერთაშორისო ელექტროტექნიკური კომისია (IEC)**. პროექტი 3 ნაწილისგან შედგება:

1. მაღალი დონის პეტრის ქსელები – კონცეფცია, განსაზღვრებები და გრაფიკული ნოტაცია;

2. პეტრის ქსელების გაცვლითი ფორმატი, სავარაუდოდ **XML**-ის ბაზაზე;

3. მოდულური პეტრის ქსელების კონსტრუქციები – იერარქიულობა, დროითი და სტოქასტური გაფართოებები.

ქვემოთ მოცემულია პეტრის ქსელის ნოტაცია უკვე ფაქტობრივ სტანდარტად დამკვიდრებული „პროექტ 15909"-ს პირველი ნაწილის საფუძველზე [5].

ძირითადი ცნებები:

საბაზო სიმრავლე (Basis Set). ობიექტების საწყისი სიმრავლე მულტი-სიმრავლეების (კომპლექტების) შესაქმნელად.

მულტისიმრავლე ანუ კომპლექტი (Multiset). ობიექტების ნაკრები, სადაც ერთგვაროვანი ობიექტების განმეორება შესაძლებელია.

მულტისიმრავლის კარდინალურობა (Cardinality). მულტისიმრავლის ელემენტების საერთო რაოდენობა.

პოზიცია (Place). ქსელის ტიპიზებული კვანძი. ქსელის გრაფში წრით ან ელიფსით გამოისახება.

გადასასვლელი (Transition). ქსელის არატიპიზებული კვანძი, რომელიც მართკუთხედით გამოისახება.

რკალი (Arc). ქსელის მიმართული კავშირის ხაზი, რომელიც აერთებს პოზიციებს გადასასვლელებთან (შემავალი რკალი) ან პირიქით (გამომავალი რკალი).

შემავალი პოზიცია (Input Place). გადასასვლელთან შემავალი რკალით შეერთებული პოზიცია.

გამომავალი პოზიცია (Output Place). გადასასვლელთან გამომავალი რკალით შეერთებული პოზიცია.

პოზიციის ტიპი (Place Type). პოზიციასთან დაკავშირებულ მონაცემთა ელემენტების არაცარიელი სიმრავლე.

მარკერი (Marker). პოზიციასთან დაკავშირებული და შესაბამისი პოზიციის ტიპის მონაცემთა ელემენტი.

მარკირება (Marking). ყველა პოზიციაში შემავალ მარკერთა ერთობლიობა.

საწყისი მარკირება (Initial Marking). ყველა პოზიციაში შემავალ მარკერთა ერთობლიობა ქსელის მუშაობის დასაწყისში.

პოზიციის მარკირება (Place Marking). პოზიციაში მოთავსებულ მარკერთა მულტისიმრავლე.

გადასასვლელის გახსნის პირობა (Transition Condition). გადასასვლელთან დაკავშირებული ლოგიკური (ბულის) ტიპის გამოსახულება.

გადასასვლელის გახსნის რეჟიმი (Transition Mode). ცვლადების დაკავშირება გადასასვლელის გახსნის პირობასთან ისე, რომ გადასასვლელის გახსნა ნებადართული გახდეს.

გადასასვლელის გახსნის ნებართვა (Enabling a Transition). გადასასვლელი რომ გაიხსნას, ყოველი პოზიციის მარკირება უნდა აკმაყოფილებდეს მისი და გადასასვლელის დამაკავშირებელი რკალის მოთხოვნას (რკალის გამოსახულება), რაც ნიშნავს, რომ მარკირება შეიცავს მარკერების მინიმუმ იმავე მულტისიმრავლეს, რაც რკალის გამოსახულებაზეა ასახული.

გადასასვლელის გახსნა (Transition Occurence). თუ გადასასვლელის გახსნა ნებადართულია, იგი შეიძლება გაიხსნას. ამ დროს გადასასვლელის ყოველი შემავალი პოზიციიდან მოიხსნება მარკერები გახსნის რეჟიმის შესაბამისად, ხოლო ყოველ გამომავალ პოზიციაში გამომავალი რკალების გამოსახულებათა შესაბამისი მარკერები ჩაემატება. პოზიცია შეიძლება ერთდროულად შემავალი და გამომავალი იყოს (**მარყუჟი**)

გადასასვლელის ცვლადები (Transition Variables). რკალებისა და გადასასვლელის გახსნის პირობაში შემავალი ცვლადების ერთობლიობა.

რკალის ანოტაცია (Arc Annotation). გამოსახულება, რომელიც შეიძლება შეიცავდეს კონსტანტებს, ცვლადებს და ოპერატორებს რკალთან დაკავშირებული პოზიციის ტიპის მულტისიმრავლიდან.

მიღწევადი მარკირება (Reachable Marking). მარკირება, რომელიც მიიღება ქსელის საწყისი მარკირებიდან გადასასვლელთა გარკვეული მიმდევრობის გახსნის შემდეგ.

მიღწევად მარკირებათა სიმრავლე (Reachability Set). საწყისი მარკირებიდან მიღწევად მარკირებათა სიმრავლე თვით საწყისი მარკირების ჩათვლით.

ალგებრა (Algebra). მათემატიკური სტრუქტურა, რომელიც შეიცავს სიმრავლეთა სიმრავლეს და ფუნქციათა სიმრავლეს, რომლებიც ამ სიმრავლეთა დომენებსა და ქვედომენებზე მოქმედებს.

ტიპი (Sort). მონაცემთა სტრუქტურის სახელი.

არგუმენტის ტიპი (Argument Sort). ოპერატორის არგუმენტის ტიპი.

გამომავალი ტიპი (Output Sort). ოპერატორის შედეგის ტიპი.

არულობა (Arity) – ფუნქციაში შემავალი (არგუმენტები) და გამომავალი (შედეგი) ტიპები (მაგ., ბინარული, **n**-არული).

ტიპიზაცია (Typisation). ტიპის დაკავშირება პოზიციასთან.

აღწერები (Declarations). გამოსახულებათა სიმრავლე სიმრავლეთა, კონსტანტების, პარამეტრების მნიშვნელობათა, ტიპიზებული ცვლადებისა და ფუნქციების განსაზღვრისათვის, რომლებიც მაღალი დონის პეტრის ქსელებზე აისახება.

ოპერატორი (Operator). სიმბოლოთა ერთობლიობა (აბრევიატურა) ფუნქციის სახელის წარმოსადგენად.

პარამეტრი (Parameter). მუდმივა (კონსტანტა), რომელიც სიმრავლეში განსაზღვრულ სიდიდეთა არეს შეიცავს.

მინიჭება (Assignment). მნიშვნელობის მინიჭება ცვლადების სიმრავლის კონკრეტული ცვლადისათვის.

სიგნატურა (Signature). ალგებრული სტრუქტურა, რომელიც ტიპების და ოპერატორების სიმრავლეებისგან შედგება.

ბულის სიგნატურა (Bool Signature). სიგნატურა, რომელიც ბულის (ლოგიკურ) ტიპს შეიცავს.

მრავალტიპური სიგნატურა (Many-sorted Signature). სიგნატურა, სადაც ტიპების სიმრავლის კარდინალურობა ერთზე მეტია.

ცვლადიანი სიგნატურა (Signature with Variables). სიგნატურა, რომელიც შეიცავს ცვლადების სახელებს, ტიპებს და ოპერატორებს.

თერმი (Term). სიგნატურის საფუძველზე შედგენილი გამოსახულება, რომელიც შეიცავს მუდმივებს, ცვლადებს და ოპერატორებს.

დახურული თერმი (Closed Term). თერმი, რომელიც შეიცავს კონსტანტებს და ოპერატორებს, მაგრამ არა ცვლადებს.

თერმის მნიშვნელობა (Term Evaluation). შედეგი, რომელიც მიიღება თერმის ცვლადებითვის მნიშვნელობების მინიჭებისა და ფუნქციათა შედეგების გამოთვლის შემდეგ.

მაღალი დონის პეტრის ქსელი (High Level Petri Net). ალგებრული სტრუქტურა, რომელიც შეიცავს: პოზიციების სიმრავლეს; გადასასვლელთა სიმრავლეს; ტიპების სიმრავლეს; ტიპების პოზიციებზე და ტიპების გადასასვლელებზე დამაკავშირებელ ფუნქციებს; პრეფუნქციებს შემავალი და პოსტფუნქციებს გამომავალი მარკირებების განსაზღვრისათვის; საწყის მარკირებას.

პეტრის ქსელის გრაფი (Petri Net Graph). მიმართული გრაფი ორი ტიპის კვანძებითა (პოზიციები და გადასასვლელები) და მათი დამაკავშირებელი რკალებით. დაშვებულია კავშირები „პოზიცია-გადასასვლელი“ ან

„გადასასვლელი-პოზიცია“, მაგრამ არა „პოზიცია-პოზიცია“ ან „გადასასვლელი-გადასასვლელი“.

მაღალი დონის პეტრის ქსელის გრაფი (High Level Petri Net Graph). ქსელის გრაფისა და ანოტაციების (წარწერების) ერთობლიობა, რომელიც შეიცავს პოზიციათა ტიპებს, რკალების ანოტაციებს, გადასასვლელთა გახსნის პირობებს, შესაბამის განსაზღვრებებს განსაზღვრებათა სიაში და ქსელის საწყის მარკირებას.

მიღწევადობის გრაფი (Reachibility Graph). მიმართული გრაფი, სადაც კვანძები მიღწევად მარკირებებს შეესაბამება, რკალები – გადასასვლელთა გახსნის ოპერაციას.

პარამეტრიზებული მაღალი დონის პეტრის ქსელის გრაფი (Parameterized High Level Petri Net Graph). მაღალი დონის პეტრის ქსელის გრაფი, რომელშიც პარამეტრები განისაზღვრება.

1.1.4. პეტრის ქსელების სემანტიკური მოდელი

პეტრის ქსელის სემანტიკური მოდელის აღწერის პროცესში გამოიყენება შემდეგი აბრევიატურები: **HLPN** – მაღალი დონის პეტრის ქსელი და **HLPNG** – მაღალი დონის პეტრის ქსელის გრაფი.

HLPN წარმოადგენს სტრუქტურას

$$\mathbf{HLPN = (P, T, D; Type, Pre, Post, M_0),}$$

სადაც:

- **P - პოზიციად** წოდებული ელემენტების სასრული სიმრავლეა;
- **T - გადასასვლელებად** წოდებული ელემენტების სასრული სიმრავლე, ისე, რომ $\mathbf{P \cap T = \emptyset}$;
- **D –** არაცარიელი **დომენების** სასრული სიმრავლე, რომლის ყოველ ელემენტს **ტიპი** ეწოდება;

- **Type : PUT$\in$D** წარმოადგენს ტიპების პოზიციებზე დაკავშირებისა და გადასასვლელის გახსნის რეჟიმის განსაზღვრის ფუნქციას;
- **Pre, Post : TRANS->μPLACE** წარმოადგენენ **წინასწარ** (გადასასვლელის გახსნამდე) და **შედეგის** (გადასასვლელის გახსნის შემდგომ) ასახვებს, სადაც
  - **TRANS = { (t,m) | t⊆T, m⊆Type(t) }**
  - **PLACE = { (p,g) | p⊆P, g⊆Type(p) }**
- **M_0⊆μPLACE** მულტისიმრავლეა, რომელსაც **ქსელის საწყისი მარკირება** ეწოდება;
- **M⊆μPLACE** მულტისიმრავლეა, რომელსაც **ქსელის მარკირება** ეწოდება;

გადასასვლელის გაშვების რეჟიმების სასრული სიმრავლე, **T_μ∈μTRANS** ნებადართულია **M**-მარკირებაში**,** თუ **Pre(T_μ)≤M,** სადაც **Pre-**ს წრფივი გაფართოება შემდეგი სახისაა:

$$\mathbf{Pre(T_\mu) = \Sigma_{tr\in TRANS}\, mult(tr,T_\mu)Pre(tr)}$$

თუ გადასასვლელის გახსნის რეჟიმთა მულტისიმრავლე **T_μ** ნებადართულია **M** მარკირებაში, მაშინ გადასასვლელის გახსნის პროცედურას **ბიჯი** ეწოდება და მისი შესრულების შედეგად მიღებული ახალი მარკირება გამოისახება ფორმულით:

$$\mathbf{M' = M - Pre(T_\mu) + Post(T_\mu)}$$

ბიჯის ფორმალური ასახვა შეიცავს საწყის და შედეგის მარკირებებს, აგრეთვე გადასასვლელთა გახსნის დაშვებული რეჟიმების მულტისიმრავლეს:

$$M \xrightarrow{T_\mu} M'$$

1.1.5. მაღალი დონის პეტრის ქსელის გრაფი – **HLPNG**

მაღალი დონის პეტრის ქსელის გრაფი წარმოადგენს სტრუქტურას:

$$\mathbf{HLPNG = (NG, Sig, H; Type, AN, M_0),}$$

სადაც

- **NG = (P,T;F) ქსელის გრაფად** იწოდება, რომელშიც
  - **P** კვანძების სასრული სიმრავლეა (**პოზიციები**);
  - **T -** კვანძების სასრული სიმრავლე (**გადასასვლე-ლები**) და **P⋂T=∅;**
  - **F⊆(PxT)⋃(TxP) - რკალებად** წოდებული მიმართული მონაკვეთების სიმრავლე**;**
- **Sig = (S, O, V)** წარმოადგენს გრაფის ნატურალურ-ლოგიკურ **სიგნატურას.**
- **H = (S_H, O_H)** სიგნატურისთვის განსაზღვრული მრავალსორტიანი **ალგებრაა;**
- **Type : P->S_H** ტიპების პოზიციებზე **დანიშვნის ფუნქციაა;**
- **AN = (A, TC)** ქსელის **ანოტაციათა** წყვილია, სადაც:
- **TC:T->TERM(O⋃V)$_{Bool}$** წარმოადგენს **ფუნქციას,** რომელიც გადასასვლელებს ლოგიკური გამოსახუ-ლების ტიპის ანოტაციით აფართოებს;
- **M_0:P->$U_{p\in P}$μType(p),** ისე, რომ **∀p∈P, M_0(p)∈μType(p)** **საწყისი მარკირების ფუნქციაა**, რომელიც მარკერთა მულტისიმრავლეს ყოველი პოზიციის **ტიპთან** კორექტულად აკავშირებს.

გრაფიკულად **პოზიცია** წრეებით ან ელიფსებით გამოისახება. პოზიციის ანოტაცია (წარწერები) შედგება მინიმუმ პოზიციის სახელის, პოზიციასთან დაკავშირებული ტიპის სახელისა და საწყისი მარკირებისგან. თუ საწყისი მარკირება ცარიელია, იგი შეიძლება არ გამოისახოს.

გადასასვლელს მართკუთხედი ან შავი ხაზი გამოსახავს; გადასასვლელის ანოტაცია შედგება მინიმუმ მისი სახელისგან; თუ **გადასასვლელის გაშვების პირობა** მოცემულია, იგი მართკუთხედის შიგნით აისახება და გამოიტოვება მხოლოდ მაშინ, როცა ყოველთვის ჭეშმარიტია.

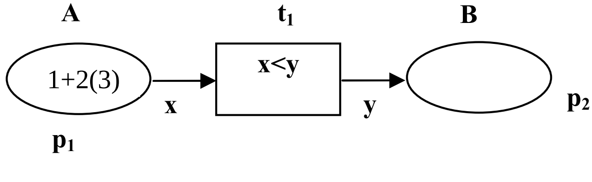
რკალს შეიძლება ჰქონდეს როგორც ერთ- ან ორმხრივმიმართული მონაკვეთის, ასევე მიმართული მრუდის სახე.

თუ ერთი და იგივე პოზიციისა და გადასასვლელის დამაკავშირებელ, შემავალ და გამომავალ რკალებს მსგავსი ანოტაციები აქვთ, ორმხრივმიმართული რკალების გამოსახვა ამ შემთხვევაში ერთი ორისრიანი რკალის სახითაც ნებადართულია. რკალის ანოტაციად ზემოთ აღწერილი **თერმების მულტისიმრავლეები** გამოიყენება.

მარკერი პეტრის ქსელში პოზიციათა მარკირების ელემენტს წარმოადგენს და შესაბამისად, პოზიციის გვერდზე (ან შიგნით) გამოისახება მისი მარკირების ფარგლებში **სიმბოლური ჯამის** სახით.

ფრჩხილები გამოიყენება მიმდინარე მარკირებაში მარკერის **სიხშირისა** (ფრჩხილებს გარეთ) და მარკერის **მნიშვნელობის** (ფრჩხილებს შიგნით) გამოსაყოფად.

პეტრის ქსელების ზემოთმოყვანილი ნოტაციის მაგალითი მოცემულია 1.2 ნახაზზე.



აღწერები

A = {1,2,3,4};

B = {3,4,5,6};

<: ZxZ→Boolean არითმეტიკული ნაკლებობა

ნახ.1.2. HLPN-გრაფი

მოცემულია 2 **პოზიცია** (**p_1** და **p_2**), 1 **გადასასვლელი** (**t_1**) და **დამაკავშირებელი რკალები**. აღიწერება 2 **ტიპი**, **A** და **B**, რომლებიც **საბაზო სიმრავლეებს** წარმოადგენს და ნატურალურ რიცხვთა სხვადასხვა ქვესიმრავლეებს შეიცავს. ცვლადი **x** **A**-ტიპისაა, **y** – **B**-ტიპის. გადასასვლელი შეიცავს **გახსნის პირობას** **$x<y$**, რისთვისაც აღწერების სიაში „ნაკლებობის" **ოპერატორი** განისაზღვრება. რკალზე (**p_1,t_1**) დართულია **ანოტაცია** – ცვლადი **x**, ხოლო რკალზე (**t_1,p_2**) **ანოტაცია** – ცვლადი **y**.

პოზიცია **p_1** **ტიპიზებულია** **A**-ტიპით და გააჩნია **საწყისი მარკირება 1+2(3),** რომელიც წარმოადგენს მულტისიმრავლეს **$M_0(p_1)=\{(1,1),(2,0),(3,2),(4,0)\}$,** სადაც ყოველი წყვილის პირველი ელემენტი ნატურალური რიცხვია **$A=\{1,2,3,4\}$** საბაზო სიმრავლიდან, ხოლო მეორე – მისი **სიხშირე** მულტისიმრავლეში. პოზიცია **p_2** **ტიპიზებულია** **B**-ტიპით და მისი **საწყისი მარკირება** ცარიელ მულტისიმრავლეს წარმოადგენს **$M_0(p_2)=\{\emptyset\}$.**

საწყისი მარკირებიდან გამომდინარე, გადასასვლელი **t_1** **ნებადართულია** გადასასვლელის გახსნის შემდეგ **რეჟიმებში: {(1,3), (1,4), (1,5), (1,6), (3,4), (3,5), (3,5)}**, სადაც ყოველი წყვილის პირველი ელემენტი **x**-ცვლადის მნიშვნელობაა, ხოლო მეორე – **y**-ცვლადისა, ისე, რომ ყოველი წყვილისთვის **$x<y$**.

მოცემულ ქსელში გაშვების რეჟიმების **პარალელიზმიც** ფიქსირდება. მაგალითად, რეჟიმების მულტისიმრავლე **(1,3)+2(3,5)** **t_1** გადასასვლელს პარალელურად ნებადართულს ხდის. შეიძლება ასეთი მაგალითის განხილვაც: **(1,5)+(3,4); (1,6)+(3,5)+(3,6)**.

თუ გაიხსნება **ნებადართული მარკირება**, მაგალითად, **(1,3) რეჟიმთა მულტისიმრავლეში** (რომელიც ამ შემთხვევაში მხოლოდ 1 ტიპის 1 ელემენტისგან შედგება), მაშინ **შედეგის მარკირებებს** შემდეგი სახე ექნება:

$M(p_1)=\{(1,0),(2,0),(3,2),(4,0)\}$;

$M(p_2)=\{(3,1),(4,0),(5,0),(6,0)\}$;

(1,3)+2(3,5) რეჟიმთა მულტისიმრავლეში გადასასვლელის გახსნა შემდეგ **შედეგის მარკირებებს** ჩამოაყალიბებს:

$M(p_1)=\{\emptyset\}$;

$M(p_2)=\{(3,1),(4,0),(5,2),(6,0)\}$;

1.2. პეტრის ქსელების კლასიფიკაცია

1.2.1. პეტრის ქსელების ქვეკლასები

პეტრის ქსელების ევოლუციის პირველ ეტაპზე მათი 3 თეორიული კლასი განისაზღვრა (I, II და III დონის პეტრის ქსელები), მაგრამ დღეისათვის მათი რაოდენობა ორამდე ჩამოვიდა - განისაზღვრება **დაბალი** და **მაღალი** დონის პეტრის ქსელები, ამასთან მეორე კლასი პირველს მოიცავს.

ძველი კლასიფიკაცია პოზიციებზე, გადასასვლელებ-სა და რკალებზე იყო ორიენტირებული და განასხვავებდა მათ ისეთ მახასიათებლებს, როგორიცაა მარკერთა მაქსიმალური რაოდენობა პოზიციაში, რკალების ჯერადობა და სხვა.

ახალ კლასიფიკაციაში ყურადღება უშუალოდ მარკერთა სემანტიკაზეა გამახვილებული.

კერძოდ, დაბალი დონის პეტრის ქსელებში დაიშვება მხოლოდ **"შავი"** მარკერები ყოველგვარი შინაგანი სტრუქტურის გარეშე, ხოლო მაღალი დონის პეტრის ქსელები დამატებით წინასწარ განსაზღვრული სტრუქტურის **"ფერად"** მარკერებსაც შეიცავს, თუმცა უნდა აღინიშნოს, რომ ტერმინები "შავი" და "ფერადი" სიმბოლურია და ლიტერატურაში მათ ხშირად განსხვავებული სახელებით მოიხსენიებენ.

მარკერთა არსი განსაზღვრავს შემდგომ პოზიციებისა და გადასასვლელების, აგრეთვე რკალების

ანოტაციის შიგთავსს. “შავმარკერიან” ქსელებში მხოლოდ არატიპიზებული, ერთგვაროვანი პოზიციები და გადასასვლელებია დაშვებული, “ფერად” ქსელებში ყველა პოზიციისთვის საკუთარი ტიპი განისაზღვრება შესაბამისი ტიპის მარკერთა დომენით. ერთი ტიპის პოზიციაში მეორე ტიპის მარკერის არსებობა დაუშვებელია.

გადასასვლელები ფართოვდება გადასასვლელის გაშვების პირობებით, რომელიც ლოგიკურ გამოსახულებას წარმოადგენს და შეიძლება ჭეშმარიტი ან მცდარი იყოს.

რკალის ანოტაცია დაბალი დონის პეტრის ქსელში ან საერთოდ გამოიტოვება (რკალში ერთ გაშვებაზე მხოლოდ ერთი “შავი” მარკერი გადაადგილდება) ან ნატურალურ რიცხვს წარმოადგენს, რომელიც გადასაადგილებელ მარკერთა რაოდენობას ასახავს (რკალის ჯერადობა), მაშინ, როცა მაღალი დონის პეტრის ქსელში რკალის ანოტაცია შეიძლება შეიცავდეს უფრო რთულ მონაცემებსაც, რომლებიც ქვემოთ განიხილება.

დაბალი დონის პეტრის ქსელების ქვეკლასებიდან შეიძლება დავასახელოთ ავტომატური პეტრის ქსელები, მარკირებული გრაფები, პეტრის ქსელები თავისუფალი არჩევანით, ელემენტარული სისტემური ქსელები, C/E-ქსელები, უსაფრთხო S/T ქსელები, S/T (კლასიკური) ქსელები და სხვა, ხოლო მაღალი დონის პეტრის ქსელების ყველაზე კარგად გამოკვლეულ და განსაზღვრულ ქვეკლასს **სისტემური პეტრის ქსელები** წარმოადგენს.

ავტომატურ პეტრის ქსელებში ანუ მდგომარეობათა მანქანებში **(State Machines)** ყოველ გადასასვლელს შეიძლება ჰქონდეს მაქსიმუმ 1 შესასვლელი და 1 გამოსასვლელი. იგი **მკაცრად შენახვადი** პეტრის ქსელია (მარკერების საერთო რაოდენობა მასში არასდროს იცვლება). ავტომატური პეტრის ქსელებით შეიძლება

კონფლიქტების, მაგრამ არა **პარალელიზმის** მოდელირება.

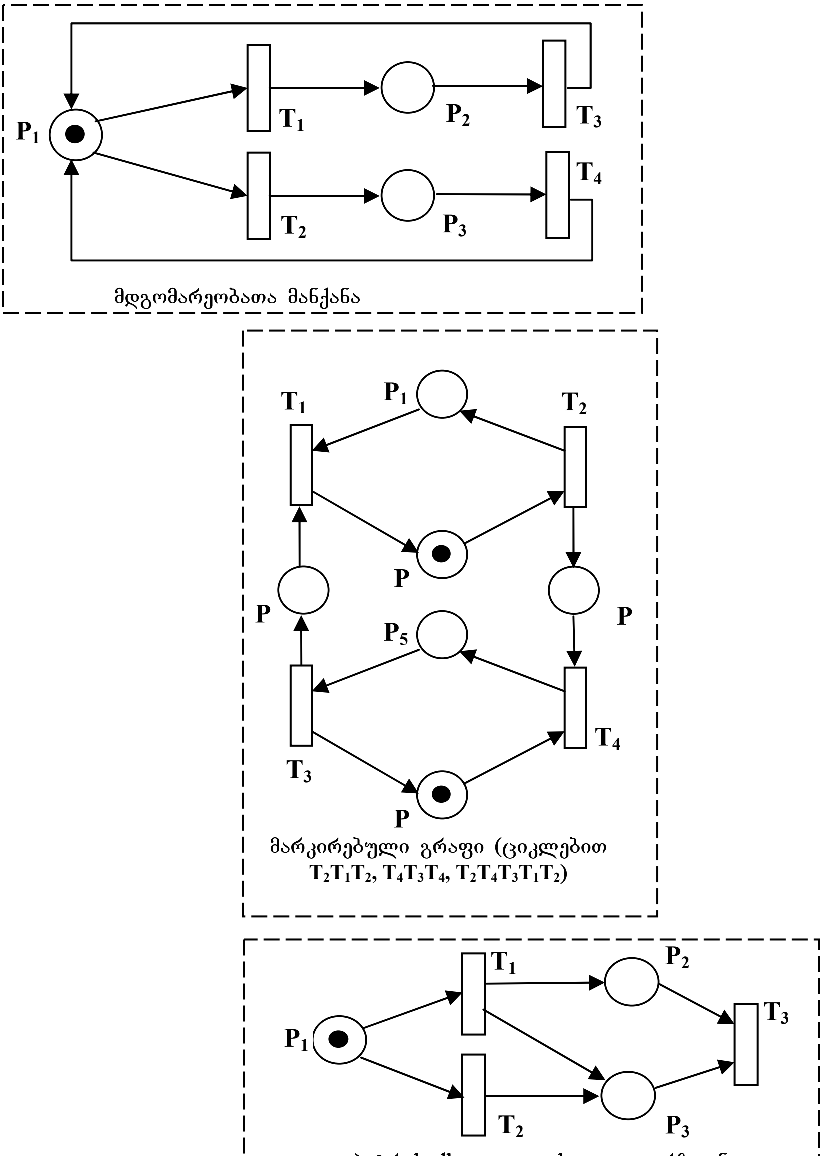
მარკირებულ გრაფებში (Marked Graphs) ყოველი პოზიცია ზუსტად 1 გადასასვლელის შესასვლელს და ზუსტად 1 გადასასვლელის გამოსასვლელს წარმოადგენს. იგი თეორიულად ავტომატური პეტრის ქსელების ორეულია, ამოდელირებს პარალელიზმს, მაგრამ კონფლიქტებს - ვერა.

მარკირებულ გრაფებში არსებობს **ციკლები** – შეკრული (ჩაკეტილი) გზა რომელიმე გადასასვლელიდან იმავე გადასასვლელამდე, რომელიც გადასასვლელთა გარკვეული მიმდევრობის გახსნით მიიღება. ციკლის გაშვების შედეგად მარკირებულ გრაფში მარკერების საერთო რაოდენობა არ იცვლება, თუმცა, ზოგადად, მარკირებული გრაფი შენახვადი არ არის (მასში მერკერების მთლიანი რაოდენობა შეიძლება იცვლებოდეს).

პეტრის ქსელებში თავისუფალი არჩევანით (Free Choise Petri Nets) მართვადი კონფლიქტის ცნება შემოდის: თუ რამდენიმე გადასასვლელს შემავალი პოზიციისთვის **კონფლიქტი** აქვს, პეტრის ქსელში თავისუფალი არჩევანით ისინი ყველა ნებადართული უნდა იყოს, ანუ საკონფლიქტე პოზიცია ერთადერთი შემავალი პოზიცია უნდა იყოს ყველა მოკონფლიქტე გადასასვლელისთვის.

ზემოთ აღწერილი 3 ქვეკლასის პეტრის ქსელების ნიმუშები მოცემულია 1.3 ნახაზზე.

S/T-ქსელები (State/Transition Nets) კლასიკური პეტრის ქსელების წარმომადგენელია. იგი შედგება მსგავსი მარკერებისგან, რომელთა გრაფიკული ფორმა პატარა შავი წრეა პოზიციის ფარგლებში.



მდგომარეობათა მანქანა

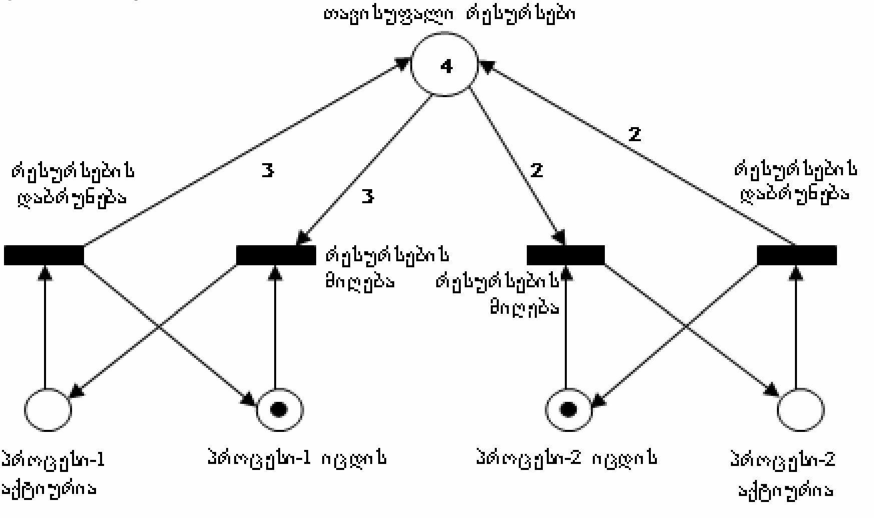
მარკირებული გრაფი (ციკლებით $T_2T_1T_2$, $T_4T_3T_4$, $T_2T_4T_3T_1T_2$)

პეტრის ქსელი თავისუფალი არჩევანით

ნახ.1.3. დაბალი დონის პეტრის ქსელების 3 ქვეკლასი

S/T-ქსელებში პოზიცია შეიძლება ერთზე მეტ მარკერს შეიცავდეს, ხოლო მარკერების დიდი ოდენობის შემთხვევაში პოზიციაში მათი რაოდენობა რიცხობრივად ჩაიწერება.

გადასასვლელის გაშვების აუცილებელი პირობაა ყველა შემავალ პოზიციაში დამაკავშირებელი რკალის ჯერადობაზე არანაკლები ოდენობის მარკერების მოგროვება. რკალების ჯერადობა ნატურალური რიცხვით გამოისახება (ნახ.1.4).



ნახ.1.4. P/T (კლასიკური) პეტრის ქსელი „რესურსების განაწილების" ამოცანისთვის

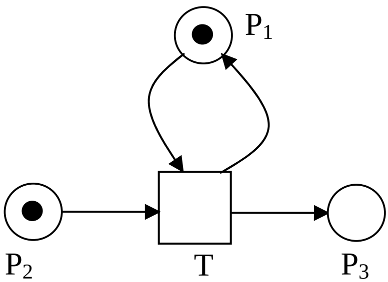
უსაფრთხო S/T-ქსელების პოზიციებში მარკერთა ოდენობა 1-ს არ უნდა აღემატებოდეს.

ელემენტარული სისტემური ქსელები (Elementary System Nets) S/T-ქსელების მსგავსად დაბალი დონის პეტრის ქსელების კლასში შედის, როგორც ყველა სხვა ქსელი, რომლებშიც მხოლოდ ერთგვაროვანი მარკერებია დასაშვები. სხვაობა ისაა, რომ ელემენტარულ სისტემურ ქსელებში ერთი პოზიცია შეიძლება არაუმეტეს ერთ

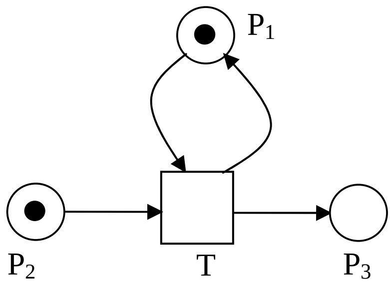
მარკერს შეიცავდეს. შესაბამისად, მასში ჯერადი რკალები არ არსებობს და რკალი ჭდეს (წარწერას) არ საჭიროებს. ამასთან, ელემენტარულ ქსელურ სისტემებში აკრძალულია გადასასვლელის გაშვება, თუ ერთი მაინც გამომავალი პოზიცია მარკერს უკვე შეიცავს [15].

მისგან განსხვავებით უსაფრთხო **S/T**-ქსელებისთვის მნიშვნელობა არა აქვს გამომავალი პოზიციების მდგომარებას, მათში გადასასვლელი გაიშვება, თუ ყველა შემავალ პოზიციაში მარკერი იქნება. ამასთან, უსაფრთხო **S/T**-ქსელები კრძალავენ პოზიციაში ერთზე მეტი მარკერის არსებობას [18]. აქედან გამოდინარე, უსაფრთხო პეტრის ქსელში მარყუჟის არსებობა გადასასვლელის გახსნას ხელს ვერ უშლის, ელემენტარულ სისტემურ ქსელში კი პირიქით.

1.5 ნახაზზე გრაფიკულად სრულებით მსგავსი ელემეტარული სისტემური ქსელი და **S/T**-ქსელი განსხვავდება შესრულების მანერით: პირველში გადასასვლელი **T** ვერ გაიშვება, მეორეში იგი ნებადართულია.



ელემენტარული სისტემური ქსელი



უსაფრთხო **S/T**-ქსელი

ნახ.1.5. გრაფიკულად მსგავსი და შესრულების წესებით განსხვავებული პეტრის ქსელები

სისტემური პეტრის ქსელები (**System Petri Nets**) ყველა ზემოაღწერილი ქვეკლასისგან განსხვავებით მაღალი დონის პეტრის ქსელის ქვეკლასს წარმოადგენს.

მაღალი დონის პეტრის ქსელების სტანდარტული ნოტაციის თანახმად, სისტემური პეტრის ქსელებისთვის განისაზღვრება **კონსტანტები, ცვლადები** და **ფუნქციები**, რომელთა ერთობლიობას სისტემური პეტრის ქსელის **სტრუქტურა** ეწოდება, ხოლო გადასასვლელებისთვის განისაზღვრება გახსნის პირობა, რომელსაც "**გადასასვლელის დამცავი ფუნქცია**" ეწოდება [**15**]. სისტემური პეტრის ქსელების ძირითადი განსაზღვრებები შემდეგია:

განსაზღვრება–1. ვთქვათ **Σ** ქსელია. **Σ**-ს **A** უნივერსუმი თითოეული $\mathbf{p}\in\mathbf{P}_{\Sigma}$ პოზიციისთვის აფიქსირებს მდგომარეობათა $\mathbf{A_p}$ სიმრავლეს, რომელსაც **A**-ში **p**-ს **დომენი** ეწოდება.

განსაზღვრება–2. ვთქვათ **Σ** ქსელია **A** უნივერსუმით, მაშინ

1. **Σ**-ს **მდგომარეობა a** თითოეული პოზიციისთვის განსაზღვრავს სიმრავლეს $\mathbf{a(p)\subseteq A_p}$;
2. ვთქვათ $\mathbf{t\in T_{\Sigma}}$. **ქმედება m** თითოეული მოსაზღვრე **f=(p,t)** ან **f=(t,p)** რკალისთვის განსაზღვრავს სიმრავლეს $\mathbf{m(f)\subseteq A_p}$.

ეს ნიშნავს, რომ სისტემური ქსელების პოზიციებში ცვლადის მნიშვნელობათა სიმრავლის მოთავსება შეიძლება, რკალსაც ელემენტარული პეტრის ქსელებისგან განსხვავებით ერთზე მეტი მნიშვნელობის გატარება შეუძლია, რაც მაღალი დონის პეტერის ქსელებს ახასიათებს.

სისტემური პეტრის ქსელების შინაარსი სტრუქტურების ცნებაზეა დაფუძნებული. განვსაზღვროთ თავიდან კონსტანტისა და ფუნქციის ცნებები.

განსაზღვრება–3. დავუშვათ $\mathbf{A_1,\ldots,A_k}$ არის ქვესიმრავლეთა სიმრავლე.

1. დავუშვათ $\mathbf{a \in A_i}$, რომელიღაც $\mathbf{1<i<k}$ - თვის. მაშინ **a**-ს ეწოდება **კონსტანტა** სიმრავლეში $\mathbf{A_1,\ldots,A_k}$ და $\mathbf{A_i}$-ს ეწოდება **a**-ს **კლასი**.

2. $\mathbf{i=1,\ldots,n+1}$ - თვის დავუშვათ $\mathbf{B_i \in \{A_1,\ldots,A_k\}}$, და ვთქვათ, $\mathbf{f{:}B_1 \times \ldots \times B_n \rightarrow B_{n+1}}$ არის ფუნქცია. მაშინ **f**-ს ეწოდება ფუნქცია $\mathbf{A_1,\ldots,A_k}$ სიმრავლეებზე. სიმრავლეები $\mathbf{B_1,\ldots,B_n}$ წარმოადგენს **f**-ის არგუმენტების ტიპებს, ხოლო $\mathbf{B_{n+1}}$ შედეგის ტიპს. **n+1** კორტეჟი $\mathbf{(B_1,\ldots,B_{n+1})}$ წარმოადგენს **f**-ის **არეს** და შემდეგნაირად ჩაიწერება $\mathbf{B_1 \times \ldots \times B_n \rightarrow B_{n+1}}$.

სისტემური პეტრის ქსელის **სტრუქტურა** კონსტანტებისა და ფუნქციების სიმრავლეს ეწოდება.

განსაზღვრება–4. დავუშვათ $\mathbf{A_1,\ldots,A_k}$ არის ქვესიმრავლეთა სიმრავლე, $\mathbf{a_1,\ldots,a_l}$ კონსტანტები $\mathbf{A_1,\ldots,A_k}$-ში და $\mathbf{f_1,\ldots,f_m}$ ფუნქციები $\mathbf{A_1,\ldots,A_k}$-ზე. მაშინ $\mathbf{A=(A_1,\ldots,A_k;a_1,\ldots,a_l;f_1,\ldots,f_m)}$ წარმოადგენს **სტრუქტურას**. $\mathbf{A_1,\ldots,A_k}$ **მატარებელი სიმრავლეებია**, $\mathbf{a_1,\ldots,a_l}$ - **კონსტანტები**, ხოლო $\mathbf{f_1,\ldots,f_m}$ - **ფუნქციები**.

სტრუქტურების ფუნქციათა შემადგენლობა **თერმების** ცნებით აღიწერება. თერმები შეიცავს **ცვლადებს** და **კონსტანტებს** და ისევე როგორც ცალკეული ცვლადები, კონკრეტულ მომენტში კონკრეტულ მნიშვნელობას ღებულობს.

თერმებისა და ცვლადების უკეთ წარმოსადგენად განვიხილოთ "მოთხოვნათა მომსახურების" სისტემა. იგი შედგება მონაცემთა 3 მომხმარებლისაგან **(u,v,w),** რომელთა ციკლურ მომსახურებას აწარმოებს მონაცემთა 2 მენეჯერი (**m** და **n**).

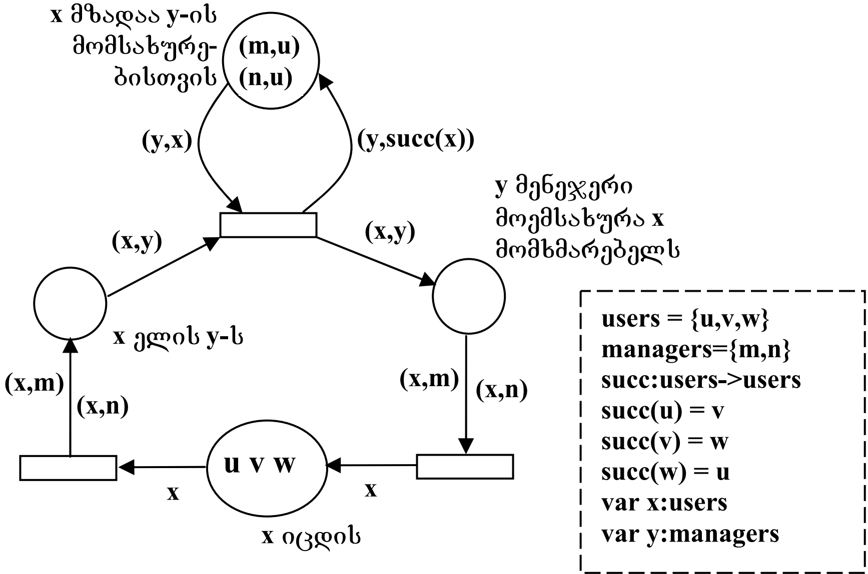
საწყის მდგომარეობაში თითოეული მომხმარებელი ლოკალურ მდგომარეობაშია, ხოლო გარკვეულ მომენტში მომხმარებელი მოითხოვს მონაცემებს ორივე მენეჯერისგან და მხოლოდ მას შემდეგ, რაც ორივესგან დაკმაყოფილდება, უბრუნდება ლოკალურ მდგომარეობას.

მომსახურებას პირველად **u** ღებულობს, მერე **v** და ბოლოს **w**. 1.6 ნახაზზე წარმოდგენილია სისტემური პეტრის ქსელი მოცემული სისტემისათვის.

ამ შემთხვევაში პეტრის ქსელის სტრუქტურას შემდეგი სახე ექნება:

"მოთხოვნათა მომსახურება"=(users,managers, users×managers, managers×users,u,v,w,m,n,succ)

სადაც **succ** (ინგლისური სიტყვიდან **success** - "წარმატება") მოთხოვნის წარმატებით შესრულების მაუწყებელი ფუნქციაა, მისი არგუმენტი **x** ცვლადია, რომელიც **users** ტიპისაა და შესაბამისად, მნიშვნელობებს მხოლოდ **{u,v,w}** სიმრავლიდან იღებს, ხოლო თავად **succ**-ის მნიშვნელობა რიგში მომდევნო ადგილას მდგარი მომხმარებელია, რომელიც ფუნქციის მეშვეობით მენეჯერთან მიმართვის უფლებას ღებულობს.



ნახ.1.6. სისტემური პეტრის ქსელი „მოთხოვნათა მომსახურების ამოცანისთვის“

სისტემურ ქსელზე რკალთა ზოგიერთი წარწერა თერმია, მაგალითად, **(x,m)** და იგი შეიცავს ცვლადსაც **(x)** და კონსტანტასაც **(m)**, როგორც ზემოთ აღვნიშნეთ.

სისტემური ქსელის მუშაობის პირველ ბიჯზე, ამოცანის პირობის თანახმად, მომსახურებას ღებულობს **u** მომხმარებელი, დავუშვათ **m** მენეჯერისგან. მაშინ თერმი **(y,x)=(m,u),** თერმი **(x,m)=(u,m)** და ცვლადი **x=u**.

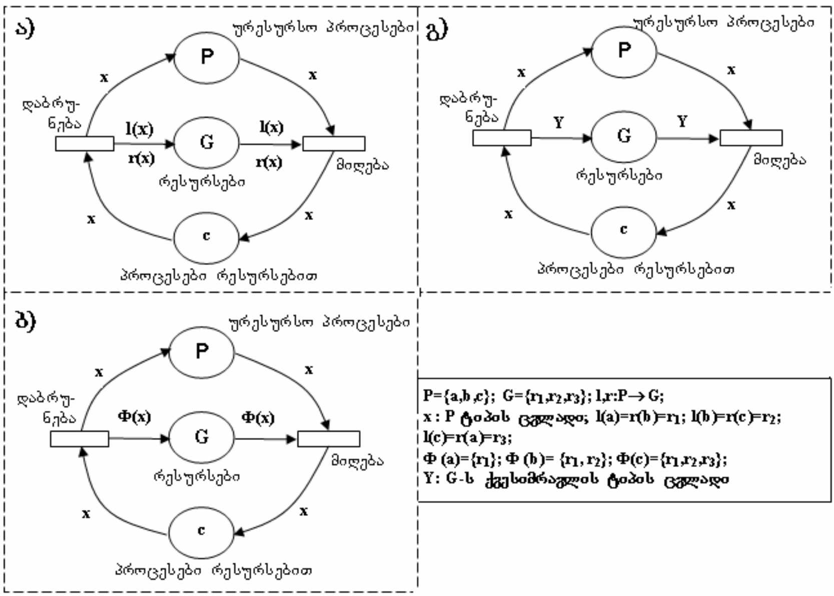
თუ **x**-ს სხვა მნიშვნელობას მივანიჭებთ, მაგალითად **v**-ს, გადასასვლელი „მომსახურების მოთხოვნა" კი გაიხსნება, მაგრამ თავად „მომსახურება" - ვერა, რადგან ამ დროს **(y,x)** თერმის მნიშვნელობა **(m,v)** უნდა გახდეს, რის საშუალებასაც პოზიციის („**y** მზადაა **x**-ის მომსახურებისთვის") საწყისი შემცველობა არ იძლევა და შესაბამისად, ქსელი არ იმუშავებს.

უფრო რთული შემთხვევებისთვის თერმების სიმრავლე შემოიტანება (**სიმრავლე-თერმები**), რომელიც სამი ტიპისაა: **კონსტანტა-თერმი, ფუნქცია-თერმი** და **ცვლადი-თერმი**.

კონსტანტა-თერმი ქსელში არა ცალკეული კონსტანტების, არამედ კონსტანტების (მაგალითად, ნატურალური რიცხვების) სიმრავლეთა ასახვისთვის გამოიყენება და ამ შემთხვევისთვის პოზიციის აღმნიშვნელი **ჭდე** არა კონსტანტების სიმრავლეს, არამედ სიმრავლის აღმნიშვნელ სიმბოლოს (დიდ ლათინურ ასოს) წარმოადგენს, ხოლო თავად სიმრავლე ქსელის სტრუქტურის განსაზღვრათა ბლოკში აღიწერება.

ფუნქცია-თერმი რკალის ანოტაციაა, აღინიშნება **Φ-**სიმბოლოთი და პეტრის ქსელში კონსტანტების ცვლადი რაოდენობის ტრანსპორტირებას ემსახურება არგუმენტ-ცვლადის მნიშვნელობის შესაბამისად.

ცვლადი-თერმი ასევე რკალის ანოტაციას წარმოადგენს და ქსელში მთლიანად სიმრავლის ან მის ქვესიმრავლეთა გადასატანად გამოიყენება. თერმების სიმრავლეთა ტიპების მაგალითები მოცემულია 1.7 ნახაზზე.



ნახ.1.7. სიმრავლე-თერმები „რესურსების განაწილების ამოცანიდან": ა) კონსტანტა-თერმები **P** და **G**; ბ) ფუნქცია-თერმი **Φ(x)**; გ) ცვლადი-თერმი **Y**

1.2.2. მაღალი დონის პეტრის ქსელების ტიპები

პეტრის ქსელების ქვეკლასიდან განისაზღვრება **პეტრის ქსელის ტიპი**. ახალი ტიპის განსაზღვრის არე შეიძლება მრავალნაირი იყოს. ყოველ პეტრის ქსელს გააჩნია შემდეგი საერთო ელემენტები:

- **პოზიციები,**
- **გადასასვლელები**
- **რკალები,**

რომლებითაც პეტრის ქსელის გრაფი იქმნება.

პეტრის ქსელის ახალი **ტიპის** განსაზღვრისას საფუძვლად სწორედ პეტრის ქსელის გრაფია აღებული და იგი შემდგომი ასახვებითა და ფუნქციებით პეტრის ქსელის კონკრეტულ ტიპამდე ფართოვდება.

პეტრის ქსელის სხვადასხვა ტიპები ერთმანეთისგან შეიძლება განსხვავდებოდეს **მარკერთა ტიპებით და მათგან გამომდინარე ერთიანი მარკირების სისტემით, ქსელის ელემენტების აღწერით (ჭდეები) ან/და გადასასვლელთა გაშვების წესებით**.

ჭდეები პეტრის ქსელის ელემენტებზე, ძირითადად მხოლოდ წარწერებია და შეიცავს **ელემენტის სინტაქსს, მაგრამ არა სემანტიკას**. შესაბამისად, ისინი ქსელის შესრულების პროცესში ვერაფერს ცვლის. ჭდეების დანიშნულება პეტრის ქსელის სინტაქსური კონტროლია.

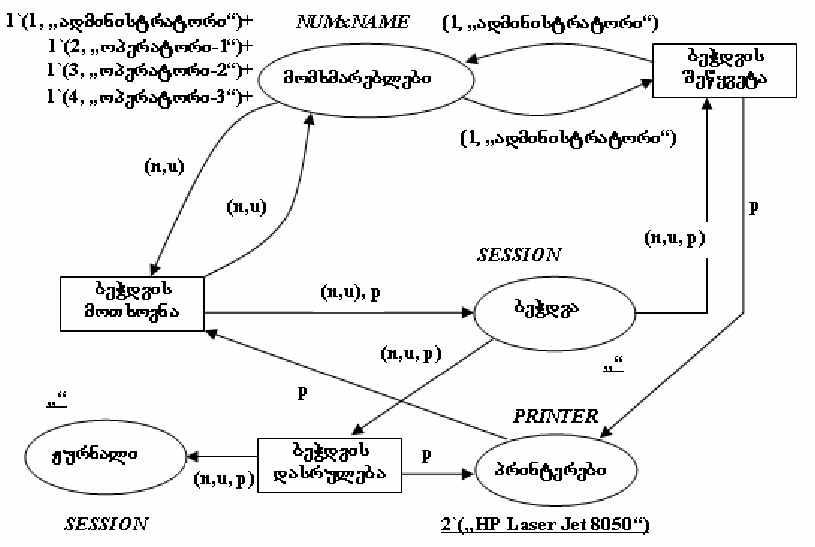
ამის მიუხედავად, ახალი ტიპის ჭდის განსაზღვრა უკვე საკმარისია იმისთვის, რომ ახალი პეტრის ქსელის ტიპი იქნეს განსაზღვრული.

პოზიციებზე, გადასასვლელებზე ან/და რკალებზე დროითი დაყოვნების განსაზღვრას დროითი პეტრის ქსელის ტიპი შემოაქვს, დაყოვნების დროთა ალბათურ განაწილებას – სტოქასტური პეტრის ქსელის ტიპი და ასე შემდეგ.

პეტრის ქსელების ტიპების განსაზღვრის უფრო ფართო გარემო მაღალი დონის პეტრის ქსელების ქვეკლასებია (მაგალითად, სისტემური პეტრის ქსელები).

ქვემოთ მოკლედ აღვწერთ ყველაზე კარგად დამუშავებულ და გავრცელებულ პეტრის ქსელის ტიპებს.

ფერადი პეტრის ქსელები (Coloured Petri Nets) მაღალი დონის პეტრის ქსელების კლასში შედის და სხვადასხვა ტიპის ანუ ფერის მარკერებს შეიცავს [7]. ტერმინი „ფერადი“ მხოლოდ ტრადიციისთვის გამოიყენება და ქსელში განსხვავებული მარკერების არსებობაზე მიანიშნებს - ამგვარი ქსელების დაბალი დონის პეტრის ქსელებისგან გამოსარჩევად, რომლებიც ერთგვაროვან, „შავ“ მარკერებს შეიცავს. ფერადი პეტრის ქსელის სტრუქტურა საკმაოდ რთულია და იგი მრავალი სახეობის ჭდეებს შეიცავს (ნახ.1.8).



განსაზღვრებები:
Type *NUM*: **Integer**; Type *NAME*: **String**; Type *NUMxNAME*: product *NUM*NAME*;
Type ***PRINTER***: **String**; Type ***SESSION***: product *NUMxNAME*PRINTER*;
var **n**: *NUM*; var **u**: *NAME*; var **p**: ***PRINTER***;

ნახ.1.8. ფერადი პეტრის ქსელი „ბეჭდვის მოთხოვნათა მომსახურების“ ამოცანისთვის

ფერადი პეტრის ქსელის ყოველ პოზიციას გააჩნია მინიმუმ ორი ჭდე: **სახელი**, რომელიც აღმნიშვნელი წრის ან ელიფსის შიგნით იწერება და მარტივი ან შედგენილი **ტიპი** (პოზიციის გვერდით, კურსივით, გასაღებური სიტყვა **Type** ან **Color**). მაგალითად, პოზიცია **„მომხმარებლები“** ***NUMxNAME*** ტიპისაა, რომელიც წინასწარგანსაზღვრული ***NUM*** და ***NAME*** ტიპების დეკარტული ნამრავლით წარმოიქმნება.

ფერადი პეტრის ქსელი შეიცავს „ფერად“ მარკერებს, რომლებიც კონკრეტული ტიპის შესაძლო მნიშვნელობათა სიმრავლეს ან მულტისიმრავლეს (კომპლექტს) წარმოადგენს.

განისაზღვრება კონსტანტები (გასაღებური სიტყვა **val**), ცვლადები (**var**) და ფუნქციები (**fun**). სხვადასხვა ტიპის მონაცემთა შორის კავშირების ასახვისთვის გამოიყენება სიმრავლეთა და კომპლექტების თეორიის ელემენტები.

გარდა მონაცემთა ტიპისა, ყოველი პოზიციის გვერდით შეიძლება აისახოს პოზიციაში მოცემულ მომენტში შემავალი ფერადი მარკერები.

საინიციალიზაციო მარკირება ხაზგასმული ტექსტის სახით გამოიტანება.

მაგალითად, საწყის მდგომარეობაში პოზიცია „მომხმარებლები“ შეიცავს ***NUMxNAME*** ტიპის ფერად მარკერთა შემდეგ სიმრავლეს (**საინიციალიზაციო მარკირება**): **{ 1`(1, „ადმინისტრატორი“), 1`(2, „ოპერატორი-1“), 1`(3, „ოპერატორი-2“), 1`(4, „ოპერატორი-3“) }**

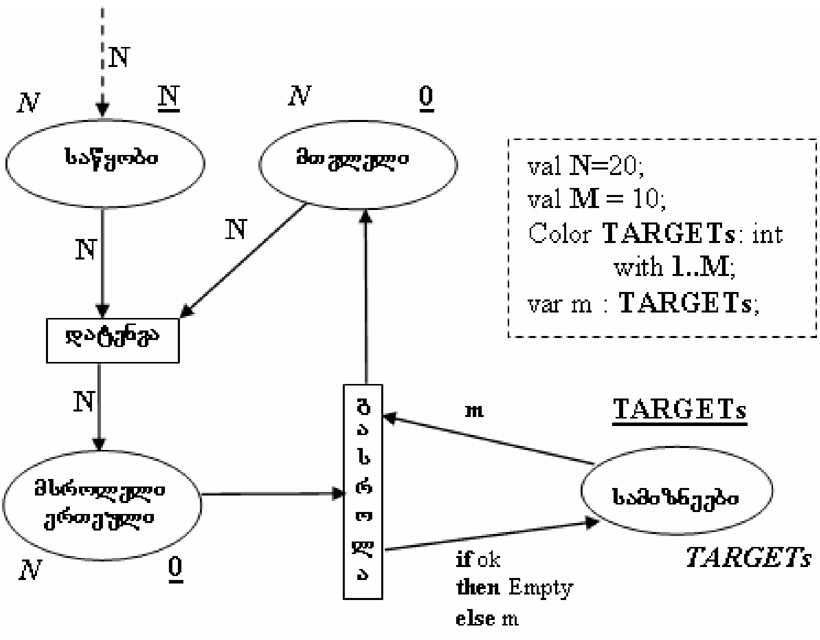
საყურადღებოა საწყისი „1“-იანი ყოველი ელემენტის დასაწყისში (მას **კოეფიციენტი** ეწოდება), რომელიც უთითებს, რომ პოზიცია არაუმეტეს 1 მოცემულ მონაცემს შეიცავს (ანუ არსებობს მხოლოდ ერთი მომხმარებელი სახელით „ადმინისტრატორი“, რომლის რიგითი ნომერია 1). ამ შემთხვევაში გვაქვს მონაცემთა ელემენტების სიმრავლე, ხოლო **2`(„HP Laser Jet 8050“)** ნიშნავს, რომ

სისტემა შეიცავს 2 ერთგვაროვან პრინტერს – ეს უკვე მულტისიმრავლე, ანუ კომპლექტია.

თუ საინიციალიზაციო მარკირების ტექსტი ძალიან გრძელია (მაგალითად მონაცემთა ბაზის ათასობით ჩანაწერს გამოსახავს), მარკერს ფსევდონიმი განესაზღვრება, რომელიც ტრადიციულად თანხვდება მარკერის ტიპის სახელს, მაგრამ იწერება ხაზგასმით.

1.9 ნახაზზე ფსევდონიმი **TARGETs** ასახავს სამიზნეთა **1..M** სიმრავლეს, ანუ ამ შემთხვევაში ნატურალურ რიცხვთა (სამიზნეთა ნომრები) სიმრავლეს 1-დან 10-მდე.

TARGETs–ის გაშლილი ვარიანტი იქნება სიმრავლე **{1`1, 1`2, 1`3, 1`4, 1`5, 1`6, 1`7, 1`8, 1`9, 1`10}**



ნახ.1.9. არადეტერმინირებული ფერადი პეტრის ქსელი „მიზანში სროლის“ ამოცანისთვის

ამავე ნახაზზეა ასახული **არადეტერმინირებული ლოგიკური გამოსახულება (პირობის ბლოკი)** ფერადი პეტრის ქსელის რკალებზე, რომელიც გადასასვლელთა გაშვების სხვადასხვა პირობებს და შედეგებს ასახავს, ანუ ლოგიკური პირობის ჭეშმარიტებისას გადასავლელს გასხვავებული მნიშვნელობა მიეწოდება (ან გადასასვლელიდან განსხვავებული მნიშვნელობა გამოვა), მცდარობისას – განსხვავებული.

ლოგიკური პირობის მნიშვნელობა სხვადასხვა შემთხვევებში სხვადასხვანაირად განისაზღვრება.

ინტერაქტიულ სიმულატორებში ჭეშმარიტება-მცდარობას თავად მომხმარებელი განსაზღვრავს, ავტომატური სიმულაციისას – შემთხვევით სიდიდეთა გენერატორი და ასე შემდეგ.

ფერადი პეტრის ქსელებში ყველაზე კარგადაა შერწყმული პეტრის ქსელებისა და დაპროგრამების (იერარქიულობა, მოდულურობა – დიდი სისტემების მოდელირებისთვის) თეორია, რაც თეორიულთან ერთად მის დიდ პრაქტიკულ ღირებულებასაც განაპირობებს თანამედროვე ინფორმაციულ ტექნოლოგიათა მრავალ სფეროში, როგორებიცაა საკომუნიკაციო პროტოკოლები, აუდიო/ვიდეო- და ოპერაციული სისტემები, აპარატურის და პროგრამების დაპროექტება, ბიზნეს-პროცესები და სხვა.

დღეისათვის ფერადი პეტრის ქსელები და მისი კომპიუტერული რეალიზაცია **„CPN-Tools“** მსოფლიოს 40 ქვეყნის 400-ზე მეტ ორგანიზაციაში გამოიყენება სისტემების მოდელირების ინსტრუმენტად (მათგან 100-მდე კომერციულ კომპანიაში) [9]. წინამდებარე სადისერტაციო სამუშაოს რიგი ექსპერიმენტები სწორედ ამ ინსტრუმენტის ლიცენზირებული ვერსიით იქნა შესრულებული.

დროითი პეტრის ქსელები ფაქტობრივად ყოველი ტიპის პეტრის ქსელისთვის დროითი გაფართოების დამატებით მიიღება. დროითი გაფართოება აუცილებელია რეალური საპრობლემო სფეროს მოდელირებისთვის, მის

გარეშე პეტრის ქსელი მხოლოდ სისტემის რაოდენობრივი ანალიზისთვის გამოდგება.

დროითი პეტრის ქსელი 4 ტიპის არსებობს: **პოზიციურ-დროითი** (**Timed Places Petri Nets - TPPNs**), **ტრანზაქციულ-დროითი** (**Timed Transition Petri Nets - TTPNs**), **რკალურ-დროითი** და **მარკერულ-დროითი** [11].

პოზიციურ-დროითი ტიპისთვის განისაზღვრება დაყოვნების ერთი და იგივე დრო პოზიციაში მოთავსებული ყველა მარკერისთვის და დროის ათვლა იწყება შესაბამისი გადასასვლელის გააქტიურებისთანავე (როცა მისი გახსნა ნებადართული ხდება). ყველა შემავალი პოზიციის დაყოვნების დროის გასვლის შემდეგ გადასასვლელი გაიხსნება.

ტრანზაქციულ-დროით პეტრის ქსელებში დაყოვნების დრო გადასასვლელისთვის (ტრანზაქციის-თვის) განისაზღვრება. პეტრის ქსელების ეს ტიპი 2 ქვეტიპს შეიცავს: **წინასწარი არჩევანისა** და **შეჯიბრების** მოდელებს.

წინასწარი არჩევანის შემთხვევაში გადასასვლელი გააქტიურებისთანავე იღებს მონოპოლურ უფლებას ყველა შემავალ პოზიციაში მოთავსებულ მარკერების იმ ოდენობაზე, რაც მისი გახსნისთვის აუცილებელია (სხვა პოზიციებთან კონფლიქტში იმარჯვებს). ამის შემდგომ იწყება დაყოვნების დროის ათვლა. მისი გასვლისთანავე გადასასვლელი გაიშვება პეტრის ქსელის წესების მიხედვით, ანუ გადასასვლელის გააქტიურებას აუცილებლად მისი გახსნა მოჰყვება.

შეჯიბრის მოდელში მთავარი დროითი ფაქტორია, მარკერები ყველა აქტიურ გადასასვლელს ეკუთვნის და გაივლის მას, რომლის დაყოვნების დროც უფრო მალე გავა.

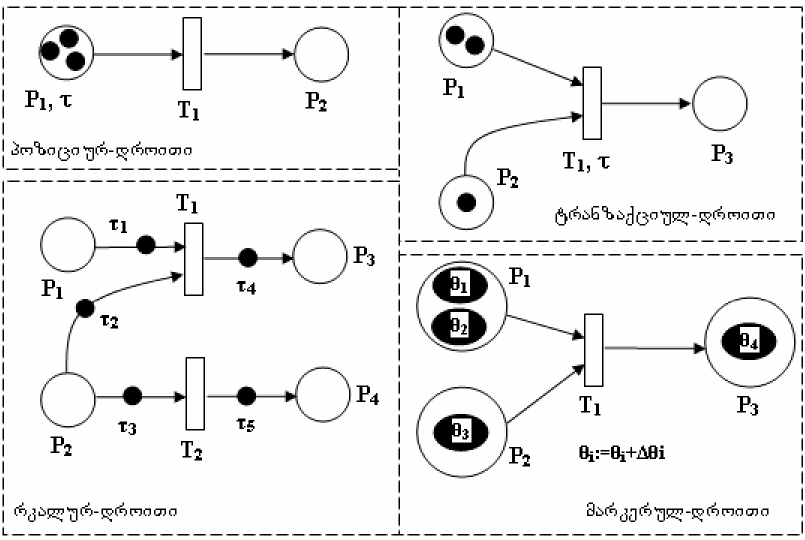
რკალურ-დროით პეტრის ქსელში დროითი დაყოვნების სიდიდეები რკალებს ენიჭება, განისაზღვრება მარკერის რკალში „მოგზაურობის“ დრო და გადასასვლელის გახსნა შესაძლებელია მხოლოდ მაშინ,

როცა ყველა შემავალ რკალში მოძრავი მარკერი უწევს გადასასვლელს.

მარკერთა „მოგზაურობა“ გადასასვლელისკენ იწყება მხოლოდ მაშინ, როცა გადასასვლელის გახსნა ნებადართული ხდება. გახსნის შემდგომაც ყოველ რკალს ენიშნება მასში მარკერის „მოგზაურობის“ დრო, სანამ იგი გამომავალ პოზიციას მიაღწევს.

მარკერულ-დროითი პეტრის ქსელი ყოველი მარკერისათვის ცალკე დაყოვნების დროის განსაზღვრას მოითხოვს. ამგვარი ტიპი მოხერხებულია დროითი პრიორიტეტების მოდელირებისთვის.

დროითი პეტრის ქსელის სხვადასხვა ტიპები 1.10 ნახაზზეა მოცემული.



ნახ.1.10. დროითი პეტრის ქსელის ტიპები

ცხადია, პეტრის ქსელების დროითი გაფართოების შემოტანა მოდელირებისას ახალ პრობლემებს წარმოშობს. მაგალითად, ტრანზაქციულ-დროით პეტრის

ქსელებში გასარკვევია, თუ როგორ უნდა ვმართოთ იმ გადასასვლელთა დაყოვნების დროები, რომლებმაც „შეჯიბრის“ შედეგად მარკერი დაკარგა და ხელახალ გააქტიურებას ელოდება გასახსნელად.

არსებობს ახალი დროითი დაყოვნების განსაზღვრის 2 ვარიანტი: დაფიქსირდება მარკერის დაკარგვისას დარჩენილი დრო (**Continue**) და გადასასვლელის შემდგომი გააქტიურებისას დროის „ჩამოყრა“ დარჩენილი დროიდან გაგრძელდება ან დროითი დაყოვნების საწყისი მნიშვნელობა ხელახლა განისაზღვრება (**Restart**). ამ ვარიანტებზე დაყრდნობით მთლიანად ტრანზაქციულ-დროითი პეტრის ქსელებითვის დროით გაფართოებათა მოდიფიცირების 3 სტრატეგია განისაზღვრება:

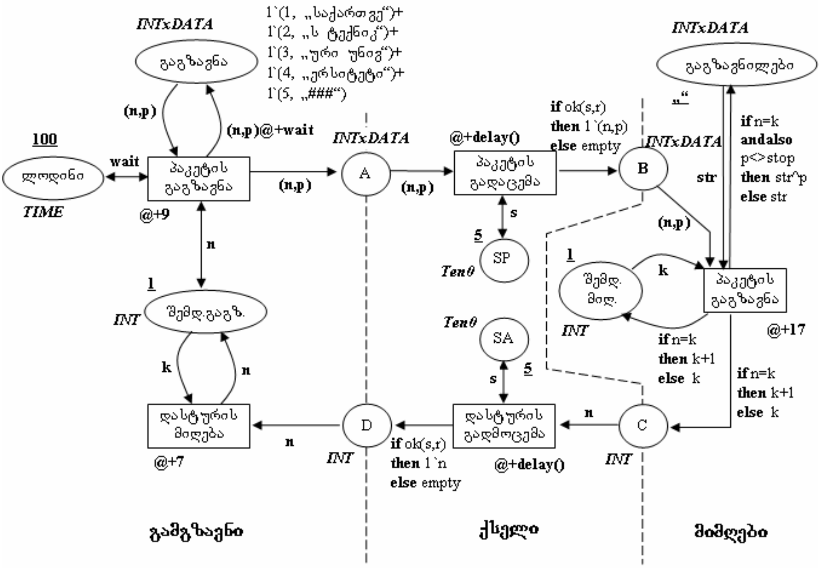
• **სრული რესტარტი (Resampling)** – ნებისმიერი გადასასვლელის გახსნისთანავე ქსელის ყველა გადასასვლელის დაყოვნების დრო თავიდან განისაზღვრება, არანაირი ინფორმაცია არ ინახება

• **ნაწილობრივი რესტარტი (Enabling Memory)** – მარკერწართმეული გადასასვლელების დაყოვნების დრო თავიდან განისაზღვრება (რესტარტი), ხოლო დანარჩენები (რომლებიც გააქტიურებულია) ჩვეულებრივად აგრძელებს დროის „ჩამოყრას“.

• **დროის შენახვა (Age Memory)** – გადასასვლელის გაშვებისას ყველა გადასასვლელის მიმდინარე დრო ინახება და გადასასვლელის შემდგომი გააქტიურებისას დროითი დაყოვნება შენახვის დროითი პუნქტიდან აგრძელებს შემცირებას.

უნდა აღინიშნოს, რომ დასაშვებია **ჰიბრიდული** პეტრის ქსელების აგება დროითი და არადროითი ელემენტებით, რაც ხშირად სისტემების მოდელირების და ანალიზის ყველაზე ეფექტურ ინსტრუმენტს წარმოადგენს.

დროითგაფართოებიანი, ჰიბრიდული, ფერადი პეტრის ქსელის კომპლექსური მაგალითი 1.11 ნახაზზეა გამოსახული.




```
color INT = int timed; color DATA = String; color INTxDATA = Product INT*DATA;
var n,k : INT; var p,str : DATA; val stop="###";
color Ten0 = int with 1..10; color Ten1 = int with 1..10; var s:Ten0; var r:Ten1;
fun Ok(s:Ten0, r:Ten1) = (r<=s);
color NetDelay = int with 25..75 declare ran; fun Delay() = ran'NetDelay();
var wait:TIME;
```

ნახ.1.11. დროითი ფერადი პეტრის ქსელი მარტივი ქსელური პროტოკოლისთვის

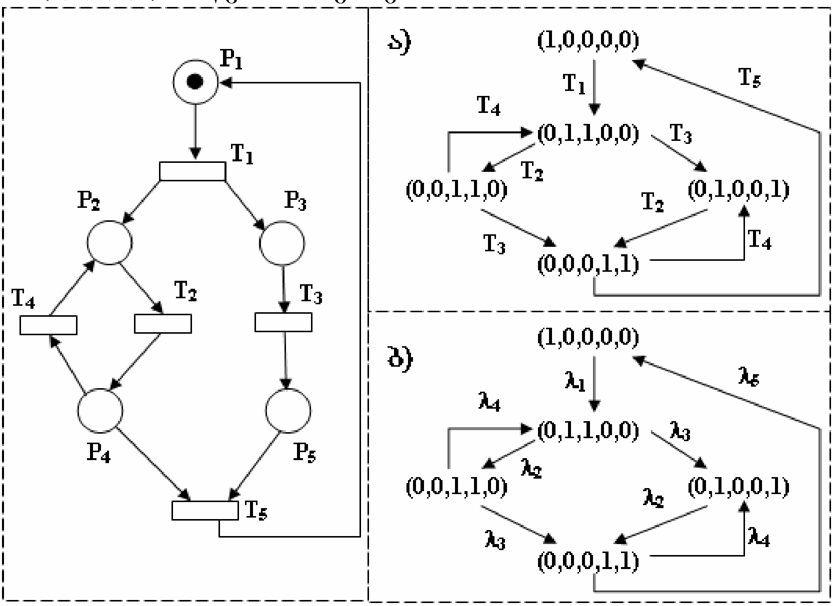
ტრანზაქციულ-დროით პეტრის ქსელებს, სადაც გადასასვლელის დაყოვნების დრო შემთხვევით განაწილებულ ექსპონენციალურ ფუნქციას წარმოადგენს, **ალბათური** ანუ **სტოქასტური პეტრის ქსელები** (**Stochastic Petri Nets**) ეწოდება, ხოლო სტოქასტური პეტრის ქსელი, რომელიც **დროითთან** ერთად **არადროით** (**მყისიერ**) გადასავლელებსაც შეიცავს, **განზოგადებულ სტოქასტურ პეტრის ქსელს** (**Generalized Stochastic Petri Nets**) წარმოადგენს [19]. ამგვარი ქსელის ქცევა **ალბათური** (მაგალითად, **მარკოვის**) **პროცესებით** აღიწერება.

მათემატიკურად სტოქასტური პეტრის ქსელი მიიღება პეტრის ქსელის განსაზღვრებაზე $\boldsymbol{\Lambda = \{\lambda_1, \lambda_2, \ldots \lambda_m\}}$

სიმრავლის დამატებით, სადაც გადასასვლელთა გაშვების დრო ექსპონენციალურადაა განაწილებული და შემთხვევითი $\boldsymbol{\lambda_i}$ სიდიდის განაწილება:

$$F_{\lambda_i}(x) = 1 - e^{-\lambda_i x}$$

სტოქასტური პეტრის ქსელის მაგალითი მოცემულია 1.12 ნახაზზე, სადაც გადასასვლელი $\mathbf{T_1}$ ნებადართულია $\mathbf{M_0=(1,0,0,0,0)}$ საწყის მარკირებაში.



ნახ.1.12. სტოქასტური პეტრის ქსელი ა) მიღწევად მარკირებათა “ხით” და ბ) ექვივალენტური მარკოვის ჯაჭვით

გადასასვლელის დაყოვნების დრო ექსპონენციალურადაა განაწილებული და დამოკიდებულია $\boldsymbol{\lambda_1}$ სიდიდეზე (გადასასვლელის კოეფიციენტი), ისე რომ გადასასვლელის გახსნის საშუალო დროა $\frac{1}{\lambda_1}$.

T_1-ის გახსნის შემდეგ მიიღება მარკირება **M_1=(1,0,0,0,0)** სადაც პარალელურად ნებადართული გადასასვლელებია **T_2** და **T_3**. თუ **T_2** პირველად გაიხსნა, მიიღება მარკირება **M_2 = (0,1,1,0,0)**, ხოლო თუ **T_3** – **M_3 = (0,1,0,0,1)**. მომდევნო მარკირებები უკვე იმაზეა დამოკიდებული, „შეჯიბრს" რომელი გადასასვლელი მოიგებს. ალბათობა იმისა, რომ **T_2** პირველად გაიხსნება, უდრის:

$$P[T_2] = P[\lambda_2 < \lambda_3] = \int_0^\infty \left(\int_0^x \lambda_2 e^{-\lambda_2 y} dy\right) \lambda_3 e^{-\lambda_3 x} dx = \int_0^\infty (1 - e^{-\lambda_2 x}) \lambda_3 e^{-\lambda_3 x} dx = \frac{\lambda_2}{\lambda_2 + \lambda_3}$$

ანალოგიურად:

$$P[T_3] = \frac{\lambda_3}{\lambda_2 + \lambda_3}.$$

ამ ფორმულებით ცხადი ხდება ისიც, რომ მარკირებათა ცვლილების ალბათობები გარკვეულ წინა მარკირებებში ყოფნის დროზე ("წინაისტორია") არ არის დამოკიდებული. სტოქასტური პეტრის ქსელების წარმოდგენა და რაოდენობრივი ანალიზი შეიძლება შესაბამისი მარკოვის პროცესების ანალიზით განხორციელდეს, რაც ასევე 1.12 ნახაზზეა ასახული. ამ მძლავრი მათემატიკური აპარატის ინტეგრაცია სტოქასტურ პეტრის ქსელებს მიმზიდველ მოდელირების საშუალებად აქცევს, განსაკუთრებით **კონფლიქტების** მოდელირებისთვის.

ობიექტური პეტრის ქსელები (**Object Petri Nets**) (და მისი გაფართოებები: **ობიექტ-ორიენტირებული** და **მაჩვენებლიანი** პეტრის ქსელები) პეტრის ქსელების თეორიისა და ობიექტ-ორიენტირებული დაპროგრამების თეორიის შეჯერებით მიღებული პეტრის ქსელის ტიპია [20]. ობიექტური პეტრის ქსელი ერთი **სისტემური** და რამდენიმე **ობიექტური** ქსელისგან შედგება, სადაც ობიექტური ქსელები **მარკერთა** როლში გამოდის. ფაქტობრივად, მიიღება პეტრის ქსელების სიმრავლე ერთი პეტრის ქსელის პოზიციებში.

ობიექტური ქსელები **ელემენტარულ სისტემურ ქსელებს** წარმოადგენს, ხოლო სისტემური ქსელი **მაღალი დონის პეტრის ქსელია**, რომლის პოზიციებშიც დაშვებულია როგორც ობიექტური ქსელების, ასევე ჩვეულებრივი შავი მარკერების არსებობა, ოღონდ არა ერთსა და იმავე პოზიციაში. შესაბამისად, რკალის ანოტაცია შეიძლება იყოს ნატურალური რიცხვი შავი მარკერებისთვის ან ობიექტური ქსელების განსაზღვრულ იდენტიფიკატორთა სიმრავლე.

სისტემური ქსელის ყოველ გადასასვლელს შეუძლია ობიექტური ქსელის გადატანა **არაუმეტეს** ერთი შემავალი პოზიციიდან **არაუმეტეს** ერთ გამომავალ პოზიციაში. ამასთანავე ერთ გაშვებაზე არაუმეტეს ერთი ობიექტური ქსელის გადატანაა ნებადართული.

ობიექტურ პეტრის ქსელებს სხვა ტიპის პეტრის ქსელებისგან გადასასვლელის როლის ზრდაც გამოარჩევს: სისტემური და ობიექტური ქსელების ზოგიერთ გადასასვლელს ემატება სპეციალური ფუნქცია, რომელსაც **ინტერაქცია** ეწოდება. ინტერაქცია 2 ტიპისაა: **სისტემ-ობიექტური** და **ობიექტ-ობიექტური**. პირველი სისტემური და ობიექტური ქსელების გადასასვლელთა სინქრონულ ურთიერთობას უზრუნველყოფს, მეორე – ობიექტური ქსელების ურთიერთსინქრონიზაციას.

სისტემ-ინტერაქციული გადასასვლელის გაშვების წესი შემდეგია: თუ სისტემური ქსელის გადასასვლელი ინტერაქციულია და მისი ინტერაქცია ქსელში მარკერის სახით მოძრავი ობიექტური ქსელის ნებადართულ ინტერაქციას თანხვდება, მაშინ სისტემური ქსელის ინტერაქციული გადასასვლელის გაშვებისას ქსელში მოძრაობის პარალელურად გაიხსნება ობიექტური ქსელის ინტერაქციული გადასასვლელიც.

სხვა შემთხვევაში (თუ ინტერაქციები არ თანხვდება, ან სისტემური ან ობიექტური ქსელის გადასასვლელები ინტერაქციებს არ შეიცავს), ობიექტური ქსელი სისტემურში უცვლელი სახით გადაადგილდება. სწორედ

ამგვარი მიდგომა განაპირობებს ობიექტური პეტრის ქსელების **ობიექტ-ორიენტირებულ** ხასიათს.

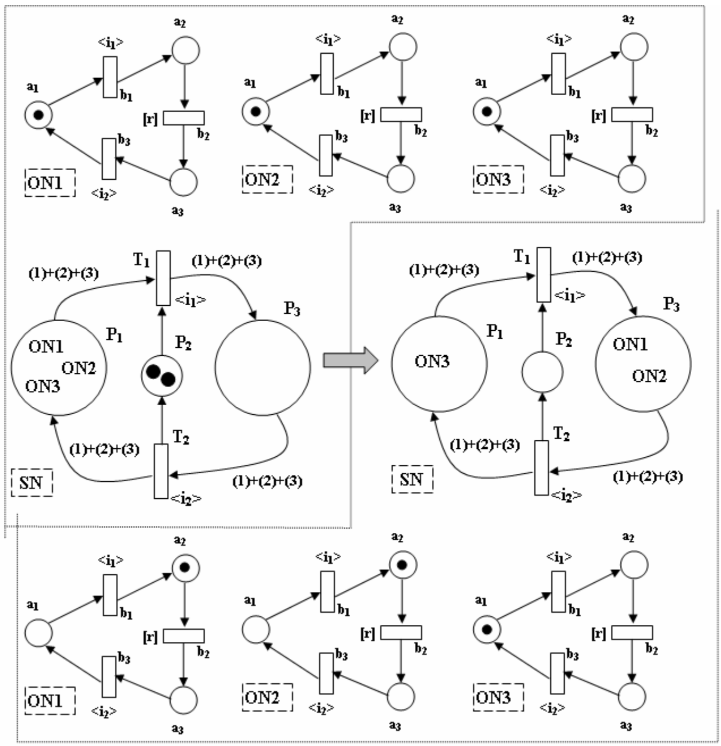
ზემოთ განხილული ფერადი პეტრის ქსელები სტრუქტურული დაპროგრამების თეორიასთან მჭიდრო კავშირში იმყოფება, შესამაბისად, შესაძლებელია ფერადი პეტრის ქსელებიდან მოქნილი გადასვლა ობიექტურ პეტრის ქსელებზე (როგორც სტრუქტურულიდან ობიექტორიენტირებული დაპროგრამების იდეოლოგიაზე) გარკვეული ახალი თვისებების შემოტანით.

რამდენიმე მარტივი შესაბამისობა ობიექტ-ორიენტირებულ დაპროგრამებასა და ობიექტურ პეტრის ქსელებს შორის 1.1 ცხრილშია მოცემული.

ობიქტ-ორიენტირებული დაპროგრამებისა და ობიექტური პეტრის ქსელების ექვივალენტური ელემენტები ცხრ.1.1.

ობიექტ-ორიენტირებული დაპროგრამება	ობიექტური პეტრის ქსელები
პროგრამული მოდული	სისტემური ქსელი
კლასი	ობიექტური ქსელის განსაზღვრება
ობიექტი	ობიექტური ქსელი კონკრეტული მარკირებით
ცვლადი	სისტემური ქსელის მარკერი
კლასის წევრი-ცვლადი	ობიექტური ქსელის მარკერი
გარე ფუნქცია	სისტემური ქსელის ინტერაქციული გადასასვლელი
კლასის წევრი-ფუნქცია	ობიექტური ქსელის გადასასვლელი

გავავლოთ პარალელი: გარე ფუნქციას (სისტემური ქსელის გადასასვლელი) კლასის წევრი-ცვლადის (ობიქტური ქსელის მარკერი) მოდიფიცირება (სხვა პოზიციაში გადანაცვლება) შეუძლია მხოლოდ მოცემული კლასის ობიექტის (მარკირებული ობიექტური ქსელი) გავლით, შესაბამისი წევრი-ფუნქციის გამოძახებით (ინტერაქციული გადასასვლელის გაშვება), როგორც ეს 1.13 ნახაზზეა მოცემული.



ნახ.1.13. სისტემური და ობიექტური ქსელების მარკირებათა ცვლილებები T_2 გადასასვლელის 2-ჯერ

სისტემური ქსელის **(SN)** გადასასვლელი $\mathbf{T_1}$ და ობიექტური ქსელების **(ON1, ON2, ON3)** $\mathbf{b_1}$-გადასასვლელები შეიცავს ერთნაირ **სისტემ-ობიექტურ ინტერაქციებს (<$\mathbf{i_1}$>),** რაც ნიშნავს, რომ თუ $\mathbf{T_1}$-ის გახსნისას ობიექტური ქსელი **ON1** გადაადგილდება, მაშინ იგი გამოიყენებს ერთ შავ მარკერს $\mathbf{P_2}$ პოზიციიდან და გადასასვლელის გახსნისას $\mathbf{P_1}$-დან მოხვდება $\mathbf{P_3}$ პოზიციაში და ამავდროულად ობიექტურ ქსელ **ON1**-ში მარკერი $\mathbf{a_1}$ პოზიციიდან $\mathbf{a_2}$-ში მოხვდება $\mathbf{b_1}$-ის გახსნის შედეგად.

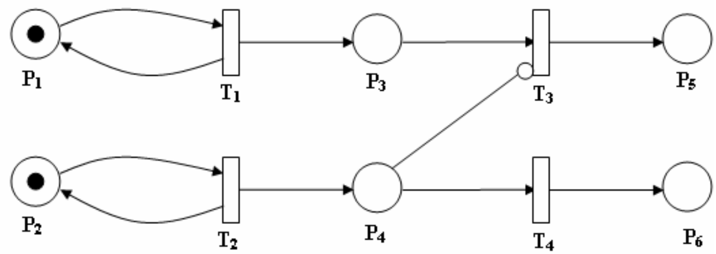
ობიექტურ ქსელებში გადასასვლელ $\mathbf{b_2}$-ისთვის განსაზღვრულია **ობიექტ-ობიექტური ინტერაქცია [r],** რომელიც ობიექტურ ქსელებს სისტემურ ქსელში პოზიციის შეუცვლელად საკუთარი შიდა მარკერების გადაადგილების საშუალებას აძლევს. მაგალითად, 1.13 ნახაზის ქვედა ნაწილში ობიექტური ქსელები **ON1** და **ON2** სისტემური ქსელის $\mathbf{P_3}$ პოზიციაში იმყოფებიან და რახან მათი $\mathbf{b_2}$ გადასასვლელები ნებადართულია, ისინი გაიშვებიან კიდეც სინქრონულად $\mathbf{P_3}$ პოზიციის დატოვების გარეშე, რის შემდეგაც უკვე სისტემური ქსელის $\mathbf{T_2}$ გადასასვლელიც ნებადართული გახდება.

1.2.3. პეტრის ქსელების გაფართოებები

პეტრის ქსელების მთელი აღწერილი მრავალ-ფეროვნების მიუხედავად მისი მამოდელირებელი სიმძლავრე შეზღუდულია და რიგი ამოცანების მოდელირებისთვის არასაკმარისი. ამ პრობლემის გადასაჭრელად განსაზღვრულია პეტრის ქსელის **გაფართოებები**, რომლებიც სპეციფიკური საპრობლემო სფეროების მოდელირებისთვის გამოიყენება [24].

ყველაზე ფართოდ გავრცელებული გაფართოებაა **შემაკავებელი რკალი (Inhibitor Arc),** რომელიც პოზიციის „ნულზე შემოწმების" პროცედურას ასრულებს, ანუ გადასასვლელს ნებადართულს ხდის მხოლოდ მაშინ, როცა შესაბამის შემავალ პოზიციაში არცერთი მარკერი არაა წარმოდგენილი. შემაკავებელი რკალი წრიული ბოლოთი გამოისახება. 1.14 ნახაზზე მისი დახმარებით გადასასვლელთა პრიორიტეტული გაშვების ამოცანა მოდელირდება (გადასასვლელი $\mathbf{T_3}$ ნებადართულია მხოლოდ მაშინ, როცა $\mathbf{T_4}$ არ არის ნებადართული, ანუ $\mathbf{P_4}$ პოზიციაში მარკერი არ არის), რაც კლასიკური პეტრის ქსელით შეუძლებელი იყო.

გარდა შემაკავებელი რკალებისა, პეტრის ქსელებში განისაზღვრება პოზიციების და გადასასვლელთა სხვადასხვა გაფართოებები.



ნახ.1.14. პეტრის ქსელი შემაკავებელი რკალით „გადასასვლელთა პრიორიტეტული გახსნის“ ამოცანისთვის

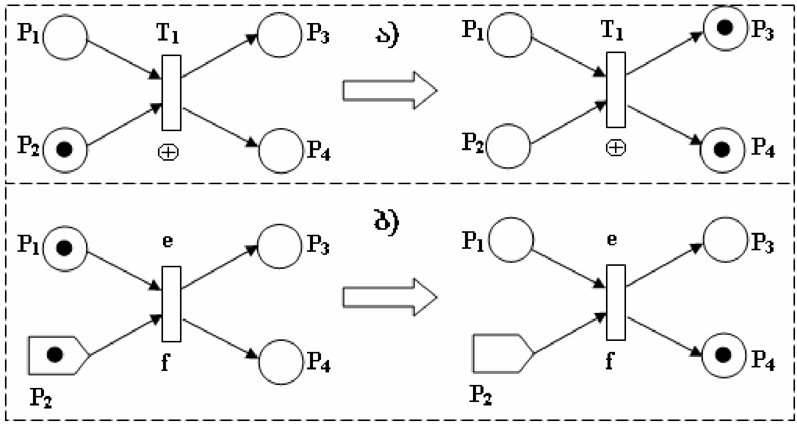
შეზღუდვის არე შემავალი პოზიციების გარკვეული სიმრავლეა, რომელთათვისაც გადასასვლელის გახსნის წესი შემდეგნაირად მოდიფიცირდება: გადასასვლელი შეიძლება გაიხსნას მხოლოდ მაშინ, როცა ერთი მაინც შემავალი პოზიცია ცარიელია.

გაფართოება **„გამომრიცხავი ან“** გულისხმობს გადასასვლელის გახსნას მაშინ, როცა შემავალი პოზიციებიდან მარკერი ერთსა და მხოლოდ ერთ პოზიციაშია (ნახ.1.15).

გადამრთველი გადასასვლელი შეიცავს სპეციალურ შემავალ **გადამრთველ პოზიციას** და ზუსტად 2 გამომავალ პოზიციას (**e** და **f** პოზიციები).

გადამრთველი გადასასვლელისთვის გადამრთველ პოზიციაში მარკერის არსებობა-არარსებობას მნიშვნელობა არა აქვს, იგი გაიხსნება, თუ სხვა შემავალი პოზიციები შევსებულია მარკერებით.

ამასთან, გამომავალი პოზიციებიდან მარკერს მიიღებს მხოლოდ: პოზიცია **e** - თუ გადამრთველ პოზიციაში მარკერი არ არის, პოზიცია **f** – თუ მარკერი მოცემულია.



ნახ.1.15. პეტრის ქსელები ა) „გამომრიცხავი **ან**“ და
ბ) „გადამრთველი პოზიცია-გადასასვლელის“ გაფართოებებით
გადასასვლელთა გაშვებამდე და გაშვების შემდეგ

აღწერილი სამი გაფართოება, როგორც წესი, ჩანაცვლებადია შემაკავებელი რკალებით, ოღონდ ამ დროს პეტრის ქსელის სტრუქტურა რთულდება.

ზოგადად, გაფართოებები ზრდის პეტრის ქსელების **მამოდელირებელ სიმძლავრეებს**, მაგრამ იმავდროულად აბათილებს ქსელის **ანალიზის** შესაძლებლობას. ამ პრობლემის შემდგომი განხილვა ჩვენი სამუშაოს ფარგლებს სცილდება.

~~1.2.4. უნიფიცირებული პეტრის ქსელების აგების კონცეფცია~~

~~წინა თავებში გაანალიზებულ იქნა პეტრის ქსელების სამყაროს პრაქტიკულად მთლიანი სივრცე, თუმცა უნდა ითქვას, რომ პეტრის ქსელების სრული კლასიფიკაცია კიდევ მრავალი ტიპის ქსელებს მოიცავს.~~

~~ზოგადად, ახალი ტიპის განსაზღვრა შესასრულებელი ამოცანის სპეციფიკაზეა დამოკიდებული. რადგანაც პეტრის ქსელები ამოცანების ფართო~~