

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

**RENKLİ PETRİ AĞLARINDA ERİŞİLEBİLİRLİK AĞACI VE P
DEĞİŞMEZLERİ ANALİZİ**

**YÜKSEK LİSANS TEZİ
Serkan YORULMAZ**

Anabilim Dalı : Elektronik ve Haberleşme Mühendisliği

Programı : Elektronik Mühendisliği

HAZİRAN 2009

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

**RENKLİ PETRİ AĞLARINDA ERİŞİLEBİLİRLİK AĞACI VE P
DEĞİŞMEZLERİ ANALİZİ**

**YÜKSEK LİSANS TEZİ
Serkan YORULMAZ
(504071220)**

Tezin Enstitüye Verildiği Tarih : 01 MAYIS 2009

Tezin Savunulduğu Tarih : 04 HAZİRAN 2009

**Tez Danışmanı : Doç. Dr. Mürvet KIRCI (İTÜ)
Diğer Jüri Üyeleri : Prof. Dr. Ece Olcay GÜNEŞ (İTÜ)
Doç. Dr. Salman KURTULAN (İTÜ)**

HAZİRAN 2009

ÖNSÖZ

Her zaman yanımda olan aileme, desteklerinden ve deneyimlerinden yararlandığım sayın hocam Doç. Dr. Mürvet KIRCI'ya, bu çalışma boyunca manevi desteklerinden dolayı sayın Azize KARAALIOĞLU'na ve yüksek lisans hayatım boyunca maddi desteklerinden dolayı TÜBİTAK'a teşekkürü bir borç bilirim.

HAZİRAN 2009

Serkan YORULMAZ
Elektronik ve Haberleşme Mühendisi

İÇİNDEKİLER

Sayfa

ÖNSÖZ.....	iii
İÇİNDEKİLER.....	v
KISALTMALAR.....	vii
ÇİZELGE LİSTESİ.....	ix
ŞEKİL LİSTESİ.....	xi
ÖZET	xiii
SUMMARY.....	xv
1. GİRİŞ.....	1
2. PETRİ AĞLARI.....	3
2.1 Petri Ağı Grafi.....	5
2.2 Klasik Petri Ağı Tanımı ve Bileşenleri.....	6
2.3 Klasik Petri Ağı Alt Sınıfları.....	11
2.3.1 Sıradan (Ordinary) Petri Ağları.....	11
2.3.2 Temiz (Pure) Petri Ağları.....	11
2.3.3 Binary Petri Ağı.....	12
2.4 Klasik Petri Ağı Dinamik Davranışı.....	12
2.4.1 Petri Ağı'nın işaretlenmesi.....	13
2.4.2 Mümkün kılınmış geçiş.....	14
2.4.3 Ateşleme dizisi.....	15
2.4.4 Erişilebilir işaretleme.....	15
2.4.5 Canlılık.....	17
2.4.6 Klasik Petri Ağı'nın matris analizi.....	17
2.4.6.1 P-Değişmezleri.....	20
2.4.6.2 T-Değişmezleri.....	23
2.4.7 Erişilebilirlik ağacı.....	24
3. RENKLİ PETRİ AĞLARI.....	27
3.1 Renkli Petri Ağı Tanımı ve Özellikleri.....	30
3.2 RPA'nın İşaretlenmesi.....	32
3.3 RPA'da Geçişin Mümkün Kılınması.....	32
3.4 Klasik Petri Ağı'ndan Renkli Petri Ağına Dönüşüm.....	38
3.5 RPA-KPA Karşılaştırması.....	44
3.6 Renkli Petri Ağı'nda Erişilebilirlik Ağacı.....	45
3.7 Renkli Petri Ağı'nda P Değişmezleri.....	56
3.8 RPA İçin Martine Silva Algoritması İle P Değişmezlerinin Bulunması.....	58
3.9 RPA İçin Satır Eliminasyonu Yöntemi Algoritması İle P Değişmezlerinin Bulunması.....	61
4. CPA-RPA PROGRAMININ ÖZELLİKLERİ.....	65
4.1 Programın Girdileri.....	65
4.2 Programın Geliştirilme Ortamı.....	66
4.3 Programın Çalışma Mantığı.....	66
4.4 Örnek Çalışma:Susturucu (Egzoz) Üretim Aşamaları.....	71

4.4.1 Örnek çalışma sonuçları.....	76
5. SONUÇLAR VE ONERİLER.....	81
KAYNAKLAR.....	83
EKLER.....	85

KISALTMALAR

PA	: Petri Ağı
KPA	: Klasik Petri Ağı
RPA	: Renkli Petri Ağı
P	: Yer
T	: Geçiş
PLC	: Programlanabilir Mantıksal Kontrolör
VLSI	: Çok Büyük Ölçekli Entegre Devre

ÇİZELGE LİSTESİ

Sayfa

Çizelge 2.1 : Yerler ve geçişlerin mümkün olabilecek anlamları.....	7
Çizelge 3.1: Şekil 3.6'deki KPA'daki yerlerin ve geçişlerin anlamları.....	39
Çizelge 3.2: Şekil 3.7'deki RPA elemanlarının anlamları.....	43
Çizelge 4.1: Program çalışma zamanı ve erişilen durum sayısı.....	70
Çizelge 4.2: Şekil 4.4 Yerlerin anlamları.....	75
Çizelge 4.3: Şekil 4.4 Geçişlerin anlamları.....	75
Çizelge 4.4: Şekil 4.4 Jeton renkleri anlamı.....	76
Çizelge 4.5: Şekil 4.4 Geçiş renkleri anlamı.....	76

ŞEKİL LİSTESİ

Sayfa

Şekil 2.1: Petri Ağı elemanları.....	4
Şekil 2.2: Geçişten yere ok, giriş oku.....	4
Şekil 2.3: Yerden geçişe ok, çıkış oku	5
Şekil 2.4: Örnek Petri Ağı grafi.....	6
Şekil 2.5: Petri Ağı örneği.....	7
Şekil 2.6: Petri Ağı ile modelleme yapıları	9
Şekil 2.7: İlk işaretlemesi yapılmış örnek Petri ağı	10
Şekil 2.8: Petri Ağı'nda öz çevrim	11
Şekil 2.9: Petri Ağı'nda öz çevrimin kaldırılması.....	11
Şekil 2.10: Petri Ağı'nda öz çevrim	12
Şekil 2.11: Petri Ağı'nda öz çevrimin kaldırılması.....	12
Şekil 2.12: İşaretlenmiş Petri Ağı	14
Şekil 2.13: İşaretlenmiş Petri Ağı'nda ateşleme sonrasında işaret değişimi.....	15
Şekil 2.14: Şekil 2.12'deki Petri Ağı'nda t_1 geçişi ateşlendikten sonra erişilen yeni durum $M_1=[2 \ 2 \ 1 \ 1]^T$	16
Şekil 2.15: Şekil 2.14'deki Petri Ağı'nda t_3 geçişi ateşlendikten sonra erişilen yeni durum $M_2=[2 \ 3 \ 1 \ 0]^T$	16
Şekil 2.16: Örnek Petri Ağı.....	18
Şekil 2.17: Şek. 2.16'teki iş. için t_1 geçişi at. sonraki yeni işaretleme.....	20
Şekil 2.18: (a) P değişmez petri ağı (b) P değişmez olmayan Petri Ağı.....	22
Şekil 2.19: Erişilebilirlik ağacı için örnek Petri Ağı.....	25
Şekil 2.20: Şekil 2.19'deki örnek Petri Ağı'nın erişilebilirlik ağacı.....	26
Şekil 3.1: Trafik ışıkları için durumların Petri Ağı notasyonu gösterimi	27
Şekil 3.2: Trafik ışıkları için durumların Renkli Petri Ağı notasyonu gösterimi	28
Şekil 3.3: Çift işlemcili üretim sistemi KPA modeli	29
Şekil 3.4: Şekil 3.1'deki ağın RPA modeli	29
Şekil 3.5: Renkli Petri Ağı örneği	31
Şekil 3.6: Şekil 3.3'deki RPA'nın Klasik Petri Ağı'na açılmış hali	37
Şekil 3.7: İki robotlu bir üretim sistemi.....	38
Şekil 3.8: İki robotlu bir sistemin Klasik Petri Ağı modeli.....	39
Şekil 3.9: Şekil 3.6'de ki KPA'nın RPA modeli	42
Şekil 3.10: RPA-KPA ilişkisi.....	44
Şekil 3.11: Şekil 3.9'deki RPA'nın $t_{1,a}$ için ateşlenmesi sonrası erişilen yeni durumu gösteren Renkli Petri Ağı.....	49
Şekil 3.12: Birbirlerini kapsayan erişilebilir durumlar.	51
Şekil 3.13: Şekil 3.7'deki RPA'nın erişilebilirlik ağacı.....	55
Şekil 4.1: Akış diyagramı elemanları.....	67
Şekil 4.2: Arama programı akış diyagramı	68
Şekil 4.3: Erişilebilirlik ağacı akış diyagramı	69

Şekil 4.4: Egzoz bileşenleri.....	72
Şekil 4.5: Susturucu (Egzoz) üretim aşamaları.....	73
Şekil 4.6: Susturucu (Egzoz) üretim aşamaları renkli petri ağı modeli.....	74
Şekil A.1: Parametre girdileri dosyası.....	87
Şekil A.2: Başlangıç işaretlemesi dosyası.....	69
Şekil A.3: Aranılan işaretleme dosyası.....	72
Şekil A.4: Çıktı matrisleri dosyası.....	89
Şekil A.5: Girdi matrisleri dosyası.....	90
Şekil A.6: Arayüz programı giriş sayfası.....	91
Şekil A.7: Arayüz programı veri giriş sayfası.....	92
Şekil A.8: Arayüz programı açıklama sayfası.....	92
Şekil A.9: Arayüz programı girişlerin yapılması sayfası.....	93
Şekil A.10: Arayüz programı analizin başlatılması sayfası.....	93
Şekil A.11: Arayüz programı hesaplama sonrası çıktı sayfası.....	94

RENKLİ PETRİ AĞLARINDA ERİŞİLEBİLİRLİK AĞACI VE P DEĞİŞMEZLERİ ANALİZİ

ÖZET

Bu çalışmada, ayrık olaylı ve eş zamanlı sistemleri modellemede ve analiz etmede kullanılan Petri Ağları'nın gelişmiş bir tipi olan Renkli Petri Ağı ele alınmıştır. Renkli Petri Ağları'nın temelini oluşturan Klasik Petri Ağları anlatılmış ve sonrasında da Renkli Petri Ağları'nın matematiksel modeli verilmiştir. Bu model üzerinden ağın davranışını gösteren erişilebilirlik ağacı ve P değişmezleri analizleri yapılmıştır. Erişilebilirlik ağacı analizi için kurulan algoritma üzerinden ve P değişmezleri için Martinez Silva algoritması üzerinden bilgisayar programları geliştirilmiştir. Geliştirilen programların zaman performansları ölçülmüştür.

İkinci bölümde Renkli Petri Ağları'nın temelini oluşturan ve anlaşılmasını kolaylaştıran Klasik Petri Ağları anlatılmıştır. Petri Ağı grafi tanımlanmış ve Klasik Petri Ağları için matematiksel tanım ve bileşenleri verilmiştir. Klasik Petri Ağı ile oluşturulabilecek yapılar anlatılmış, Klasik Petri Ağı alt sınıfları verilmiştir. Klasik Petri Ağı'na ilişkin genel kavram ve tanımlamalar verilerek matris analizleri yapılmıştır. Erişilebilirlik ağacı ile P ve T değişmezleri tanımlamaları yapılmıştır.

Üçüncü bölümde Renkli Petri Ağları'ndan bahsedilmiştir. Matematiksel tanımı ve özellikleri verilmiştir. Renkli Petri Ağları için ateşlemenin mümkün kılınması ve işaretleme kavramlarından bahsedilmiş ve sonrasında Klasik Petri Ağları'ndan Renkli Petri Ağları'na dönüşümün nasıl olacağı anlatılarak bu iki Petri Ağı türü karşılaştırılmıştır. Renkli Petri Ağları'nda erişilebilirlik ağacı kavramından bahsedilerek örnekler üzerinden erişilebilirlik ağacı verilmiştir. Erişilebilirlik ağacını çıkartan algoritma geliştirilmiştir. Sonrasında Renkli Petri Ağı'nda P değişmezleri incelenmiş ve P değişmezlerini bulmakta kullanılan Martinez Silva algoritması Renkli Petri Ağları'na uygulanmıştır. P değişmezlerini hesaplamada kullanılabilecek bir başka yöntem olan Satır Eliminasyon Yöntemi kullanılarak P değişmezleri hesaplanmıştır.

Dördüncü bölümde erişilebilirlik ağacı analizi için geliştirilen program ile Martinez Silva algoritmasını Renkli Petri Ağı'na uygulayarak P değişmezlerini hesaplayan programın birleştirilmesinden meydana gelen CPN-RPA programının özellikleri anlatılmıştır. Programın ihtiyaç duyduğu girdiler ve bu girdilere karşılık verdiği çıktılar anlatılmıştır. Üçüncü bölümde incelenen örneklere ilişkin programın çıktıları verilmiştir. Programın akış diyagramı sunulmuş ve örnek olarak egzoz üretim aşamaları modellenerek programın çalışması sonucu elde edilen çıktılar verilmiştir.

Son bölüm olan beşinci bölümde yapılan çalışmanın sonuçları ve gelecekte yapılabilecek çalışmalar için önerilerde bulunulmuştur

REACHABILITY TREE AND P INVARIANTS ANALYSIS OF COLORED PETRI NETS

SUMMARY

In this study, Colored Petri Net which is an advanced model of Petri Nets is analysed. It is used to model discrete event and concurrent systems. Classical Petri Net which is the base of Colored Petri Nets is explained and then the mathematical model of Colored Petri Nets is given. Based on this model, the analysis of reachability tree which shows the behavior of Petri Net and the analysis of P invariants are prepared. Computer programs are developed for reachability tree over formed algorithm and for P invariants over Martinez Silva algorithm. Time performances of these programs are measured.

In the second part, Classical Petri Nets which form the base of Colored Petri Nets and make easy to understand Colored Petri Nets are explained. Graph of Petri Net is introduced, the mathematical definition and the elements of Classical Petri Net are given. Structures which can be built with the Classical Petri Nets and the subclasses of the Classical Petri Nets are mentioned. General concepts and definitions are given and the matrix analysis of Classical Petri Nets is calculated. Analysis of reachability tree, P and T invariants are explained.

In the third part, Colored Petri Nets are examined in detail. Mathematical definition and the properties of Colored Petri Nets are given. Enabling the firing condition of the transitions and marking of the net are explained and then the transformation conditions of Classical Petri Nets to Colored Petri Nets is represented. The comparison of these two kinds of Petri Nets is given. Reachability tree of Colored Petri Nets are explained and the examples are given over these explanations. The algorithm to calculate reachability tree is developed. After this, P invariants of Colored Petri Nets is analysed and the Martinez Silva Algorithm which is used to calculate P invariants is applied to Colored Petri Nets. P invariants are calculated by an other calculation method which is called Row Elimination Method .

In the fourth part, the features of CPN-RPA software, which is the combination of two programs, are explained. One of these program is used for the analysis of reachability tree by the developed algorithm and the other program is used for the calculation of P invariants by applying Martinez Silva algorithm into Colored Petri Nets. Inputs and the outputs of the CPN-RPA are mentioned. Software outputs related to examples at section three are given at this section. Flow chart of this software is represented and with modelling exhaust production stages as an example, the CPN-RPA software outputs are given.

At fifth part which is the last part, results of this study and suggestions for the future studies are given.

1. GİRİŞ

Petri Ağları (PA) grafiksel ve matematiksel olarak ayırık olaylı ve eş zamanlı çalışan sistemlerin modellenmesinde ve analiz edilmesinde kullanılan araçlardır. Grafiksel olarak akış diyagramlarına veya durum makinelerine benzerler. Matematiksel bir araç olarak ise durum matrisleri ve diğer matematiksel analizler ile sistem davranışı araştırılabilir.

Petri Ağları teorisi ilk olarak Carl Adam Petri tarafından 1962’de Bonn’da sunulan “Communication with automata” (Otomata ile haberleşme) isimli doktora çalışması olarak ortaya çıkmıştır [1]. 1968 ve 1976 yılları arasında araştırmacılar tarafından teori geliştirilmiştir. Birçok alt bileşen sistemlerine sahip kombine bir sistemin modellemesi amacıyla kullanılabilen Petri Ağları’nın kullanım alanları üretim tesisi otomasyonu, sistem performans ölçümleri, haberleşme protokolleri, dağılmış yazılım sistemlerinin özelliklerinin belirlenmesi ve tasarımı, iş akış modellemeleri, PLC sistemlerin ve VLSI devrelerinin performans değerlendirmesi gibi alanlar olabilmektedir.

Birçok çeşitleri geliştirilmiştir. Renkli, Zamanlı, Deterministik, Stokastik, Hibrit vb. türlerde Petri Ağları’na rastlamak mümkündür.

Petri Ağları ile ilgili birçok yöntem, araç ve algoritma geliştirilmiş olmasına karşın bunların elektronik sistemlerin tasarımına yönelik alanlarda kullanımları henüz standartlaşmayı tamamlamamıştır. Bu yüzden Petri Ağları’nın elektronik sistemlerin tasarımında kullanımına yönelik çalışmalar yapılabilir.

Bu çalışmada K.Jensen [2] tarafından tanımlanan ve jetonlara etiketlenen renk bilgisini temel alan Renkli Petri Ağları’nın (RPA) davranışsal ve yapısal özelliklerinden erişilebilirlik ağacı ve P değişmezleri analizi incelenmiştir.

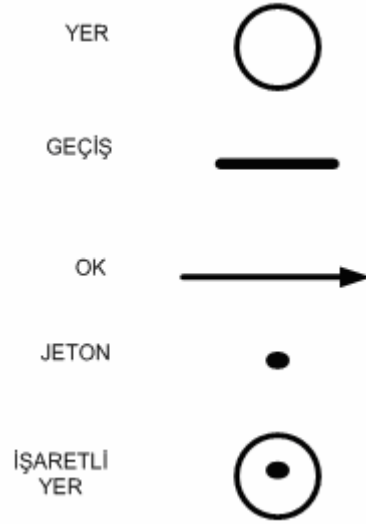
2. PETRİ AĞLARI

Petri Ağları ilk defa Carl Adam Petri [1] tarafından ortaya atılmıştır. Petri Ağları, yerlerin ve geçişlerin, - yönlendirilmiş ve yerlerden geçişlere, geçişlerden yerlere akış hakkında bilgi veren- , oklar ile birbirine bağlandığı çift taraflı bir grafiksel sunuştur.

Petri Ağları'nın uygulaması modelleme ile olmaktadır. Birçok araştırma alanında incelemeler doğrudan olaylar üzerinde yapılmaz, bunun yerine dolaylı olarak modeller kullanılarak analizler gerçekleştirilir. Bir model incelenmekte olan nesnenin veya sistemin önemli olduğu yönlerinin bir gösterimidir. Bu gösterim çoğu zaman matematiksel ifadeler ile yapılmaktadır. Elektrik, Elektronik, Astronomi, Nükleer Fizik, Sosyoloji, Biyoloji ve daha birçok alanda modelleme teknikleri kullanılır ve bu modeller matematiksel eşitliklerle ifade edilebilir. Modellemenin başarılı olabilmesi için incelenen olaya ve modelleme tekniğine hakim olmak gerekir.

Ayrık olaylı sistemlerin modellenmesinde Petri Ağları kullanılabilir. Sistem analizi de yine Petri Ağları'ndan yararlanarak yapılabilir. Analizde görülen hatalardan yola çıkarak tasarım yeniden incelenir. Bu adımlar sorunsuz çalışma elde edilene kadar tekrarlanır.

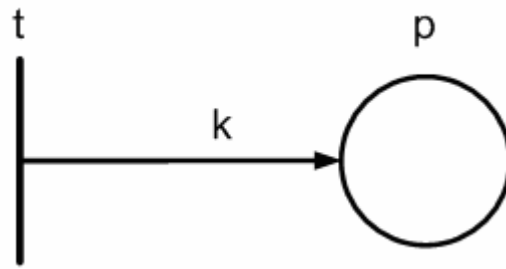
Petri Ağları'nda olaylar geçişlerle ilişkilendirilir. Bir geçişin oluşması için gerekli olan koşullar yerler ile gösterilir. Olayların gerçekleşmesi için gerekli olan durumlar “giriş” yerleri, geçişe çıkış olan ve geçişten etkilenen durumlar ise “çıkış” yerleri olarak adlandırılır. Geçişler, yerler ve bunlar arasındaki oklar Petri Ağı yapısını tanımlayan temel bileşenlerdir.



Şekil 2.1: Petri Ağı elemanları

İşaretleli bir Petri Ağı'nda ise, her yer, sistemin anlık/kısmi durumunu veren , sıfır veya pozitif tamsayıda jetona sahiptir. Bir geçişin ateşlenmesi ile oklar üzerinde gösterilen sayıda jeton ilgili yerlerden azalırken, ilgili yerlere de yine oklar üzerinde gösterilen sayıda jeton eklenmektedir.

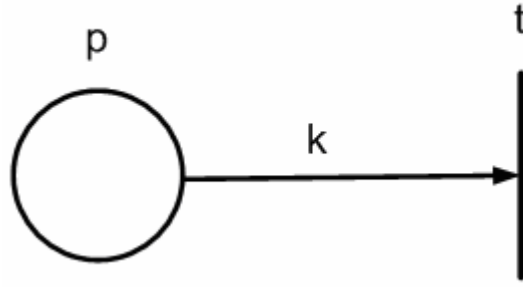
Geçişten yere yönelmiş bir ok ve ağırlığını şöyle ifade edilsin, $w(t, p)=k$,



Şekil 2.2: Geçişten yere ok, giriş oku

Bu demektir ki t geçişi ateşlendiği zaman, p yeri k tane jeton alacaktır.

Yerden geçişe yönelmiş bir ok ve ağırlığını şöyle ifade edilsin, $w(p, t)=k$,



Şekil 2.3: Yerden geçişe ok, çıkış oku

Bu demektir ki t geçişi ateşlendiği zaman, p yeri k tane jeton verecektir.

Grafiksel olduğu gibi matematiksel olarak da bir sistemin davranışını modelleme, buna göre tasarım yapma ve tasarımın doğruluğunu test etme imkanı tanıyan Petri Ağları birçok türe sahiptir.

Bu çalışmada kullanılan “Renkli Petri Ağı” Petri Ağı çeşidinin daha iyi anlaşılabilmesi için, temelini oluşturan “Klasik Petri Ağı” hakkında aşağıda bilgi verilecektir.

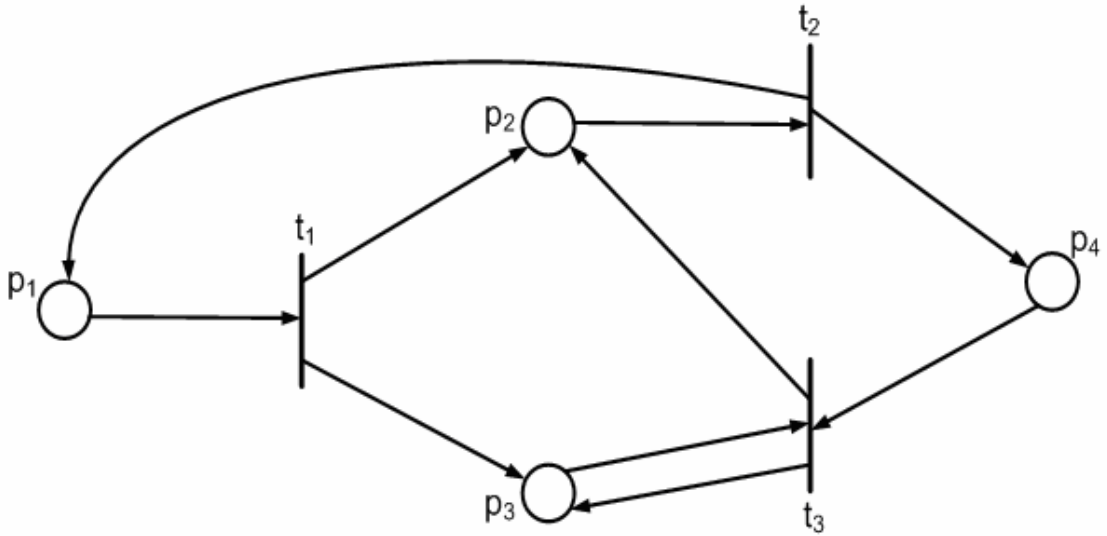
2.1 Petri Ağı Grafı

Bir Petri Ağı’nın grafiksel gösterimi, Petri Ağı teorisinin kavramlarını göstermesi bakımından çok daha kullanışlıdır. Bir Petri Ağı grafı, Petri Ağı yapısını gösteren iki elemanlı, yönlü ve ağırlıklı bir graftır. Bir Petri Ağı yapısı yerler ve geçişlerden oluşur. Buradan hareketle bir Petri Ağı grafında iki tür düğüm bulunması gerektiği söylenebilir.

Bir daire (○) yere işaret ederken, bir çizgi veya dikdörtgen (|, □) geçişleri temsil eder. Daireler yerleri temsil ettiğinden bunlar yer adını alır, benzer şekilde çizgiler veya dikdörtgenler de geçiş adını alır.

Yönlü çizgiler (dallar) yerleri ve geçişleri birbirine bağlar. Bir p_i yerinden t_j geçişine yönelen bir ok o geçişe bir giriş yeri olduğunu işaret eder. Benzer şekilde bir geçişten yerlere yönelen oklar da o geçişin çıkışlarını ifade eder. Bir Petri Ağı’nda bir düğümden bir diğerine çizilen oklar farklı ağırlıklarda olabileceğinden ağırlıklı bir graftır. Yine bu oklar yönlü olduğundan oluşan yapı “yönlü ağırlıklı graf” ismini alır. Bir Petri Ağı grafının düğümleri yerler ve geçişler olmak üzere iki kümeye ayrılır, öyle ki her bir kümenin bir elemanından diğer kümenin bir elemanına yönelen yönlü

oklar tanımlanabilir. Bu yapıya “iki parçalı yönlü ağırlıklı graf” denir. Kısaca bileşenlerini tanımlanan bu yapılara “Petri Ağı Grafı” ismini verilir [3].



Şekil 2.4: Örnek Petri Ağı grafı

2.2 Klasik Petri Ağı Tanımı ve Bileşenleri

Matematiksel olarak, Petri Ağı 5-bileşenli bir küme ile gösterilir, [1,4,5,6]

$$PA : (P; T; A; W; M_0)$$

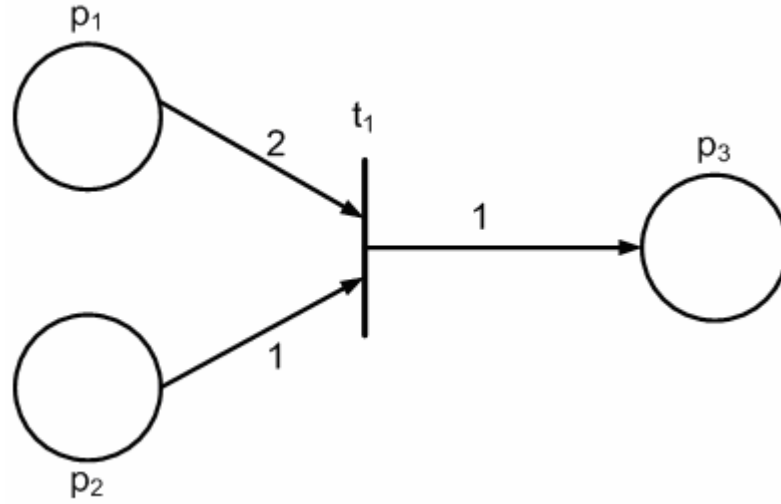
$P = p_1; p_2; \dots; p_m$, sonlu yerler kümesi

$T = t_1; t_2; \dots; t_n$, sonlu geçişler kümesi

$A \subseteq (P \times T) \cup (T \times P)$, sonlu oklar kümesi

$W : A \rightarrow 1; 2; 3; \dots$, ok ağırlık fonksiyonu

$M_0 : P \rightarrow 0; 1; 2; 3; \dots$, başlangıç durumu işaretlemesi



Şekil 2.5: Petri Ağı örneği

Petri Ağı iki tipte düğümden oluşmaktadır, bunlar yerler (P) ve geçişler (T) dir. Oklar (A) yerlerden geçişlere (PxT) veya geçişlerden yerlere (TxP) yönlendirilmiş olabilir. Petri Ağı'nda yerler daireler veya elipslerle, geçişler ise çizgi veya dikdörtgen kutular ile gösterilir. Oklar ağırlıklarını gösteren (w) pozitif tamsayılarla etiketlenmiş olarak gösterilirler. Bu ağırlık etiketleri yere eklenen veya yerden eksilen jeton sayısını gösterir. Ok ağırlığı 1'den fazla olan okların bulunduğu ağa Genel Petri Ağı denirken tüm okların ağırlıkları 1 olan Petri Ağı'na ise Basit Petri Ağı denir.

Yerler Petri Ağı'nın pasif elemanlarıdır ve gerçek dünyadaki karşılıkları durumlar, şartlar veya kaynaklar olabilir. Geçişler ise Petri Ağı'nın aktif elemanlarıdır ve gerçek dünyada olaylar, hareketler vb aktivitelere denk düşerler. Yerler ve geçişlerin mümkün olabilecek anlamları aşağıdaki çizelge ile verilebilir.

Çizelge 2.1 : Yerler ve geçişlerin mümkün olabilecek anlamları

Giriş Yeri	Geçiş	Çıkış Yeri
İhtiyaç Duyulan Kaynaklar	İş/Görev	Salıverilen Kaynaklar
Şimdiki Durumlar	Olaylar	Sonraki Durumlar
Giriş Sinyali	Sinyal İşleyici	Çıkış Sinyali
Koşullar	Lojik Koşul	Sonuçlar

Petri Ağları daha önce de belirttiğimiz gibi birçok alanda birçok araştırmacı tarafından kullanılmaktadır. Bunun sebebi ayrık olaylı sistemlerin ve özellikle aşağıda göstereceğimiz yapıların modellenmesinde güzel bir formalizasyona sahip olmasıdır [1]. Petri Ağlarıyla bir çok yapı modellenenebilir bunlardan bazıları aşağıda anlatılmıştır[7].

t_1 ve t_2 birbirlerinden farklı geçişler olsun;

Çatışmalı Yapılar: Geçişlerinin aynı anda mümkün kılındığı fakat birinin ateşlenmesi diğerinin ateşlenebilirliğini ortadan kaldırdığı yapılardır.

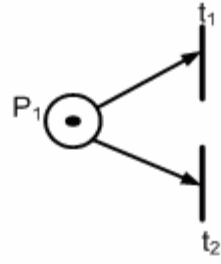
Eş zamanlı Yapılar: Geçişlerin ateşlenebilirliğini belirleyen giriş yerlerinin birbirlerinden farklı bağımsız olduğu yapılardır ve bu yapılarda her iki geçiş de aynı anda ateşlenebilir.

Sıralı Yapılar: Geçişlerin birinin ateşlenebilir hale gelmesi için diğerinin ateşlenmesi gerekir, yani ateşlenmeleri birbirine bağlı ve birbirinin ardından oluşur.

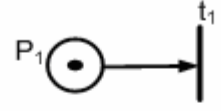
Dallanmalı Yapılar: t_1 'in ateşlenmesi sonrasında birbirinden farklı iki ayrı yere jeton kazandıran yapılardır. Dallanma sonrasında eş zamanlı yapı meydana gelir.

Birleşmeli Yapılar: t_1 'i mümkün kılan yer sayısının birden fazla olduğu ve t_1 ateşlenmesi sonrasında sadece bir tane yerin jeton kazandığı yapılardır. Eş zamanlı yapıların birleştirilmesinde kullanılır.

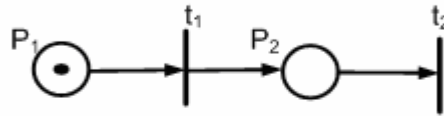
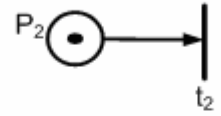
Senkronlaştırma Yapıları: Eşzamanlı yapıların birbirleriyle senkron yani aynı anda mümkün kılınmasını gerektiren yapılardır.



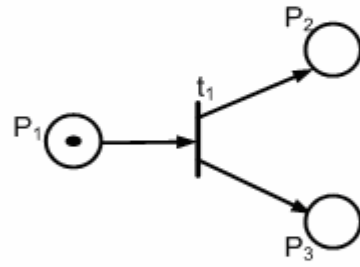
Çatışmalı Yapı



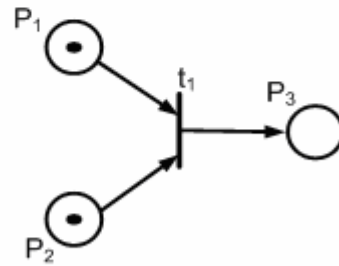
Eş zamanlı Yapı



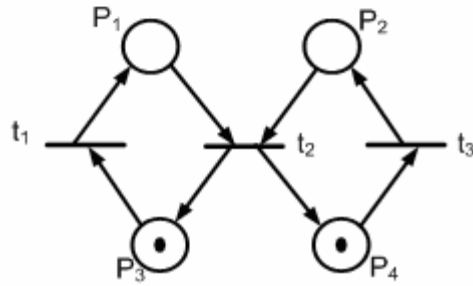
Sıralı Yapı



Dallanmalı Yapı

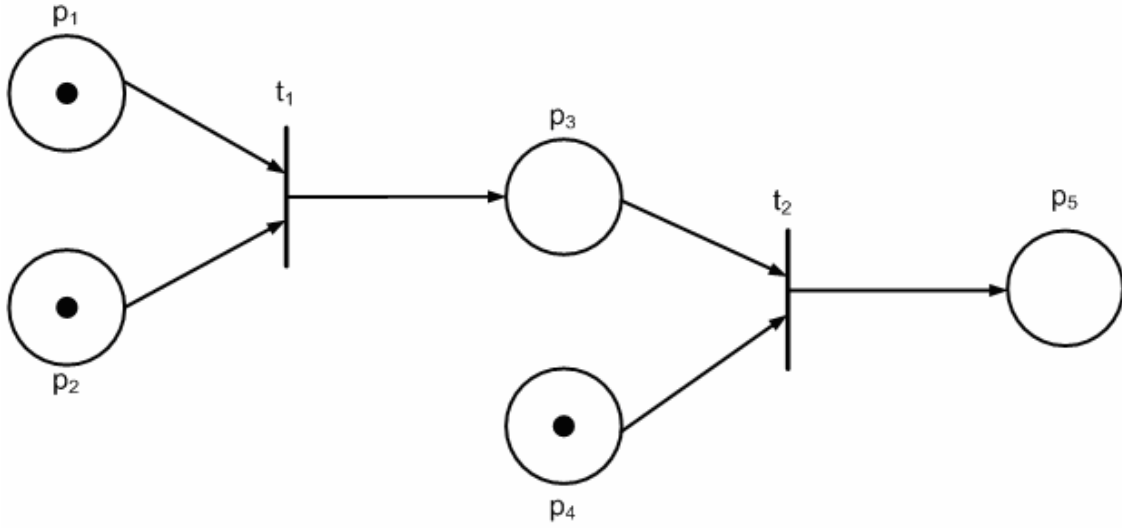


Birleşmeli Yapı



Senkronlaştırma Yapısı

Şekil 2.6: Petri Ağı ile modelleme yapıları



Şekil 2.7: İlk işaretlemesi yapılmış örnek Petri Ağı

Şekil 2.7’deki Örnek Petri Ağı için,

$$P = \{P_1, P_2, P_3, P_4, P_5\} \quad T = \{t_1, t_2\}$$

$$M_0 = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

$$A = \{(p_1, t_1), (p_2, t_1), (t_1, p_3), (p_3, t_2), (p_4, t_2), (t_2, p_5)\}$$

$$w(p_1, t_1)=1, w(p_2, t_1), w(t_1, p_3)=1, w(p_3, t_2)=1, w(p_4, t_2)=1, w(t_2, p_5)=1$$

$$w(p_1, t_2)=0, w(t_1, p_1)=0, w(p_2, t_2)=0, w(t_1, p_2)=0, w(p_3, t_1)=0 \dots$$

2.3 Klasik Petri Ağı Alt Sınıfları

2.3.1 Basit (Ordinary) Petri Ağları

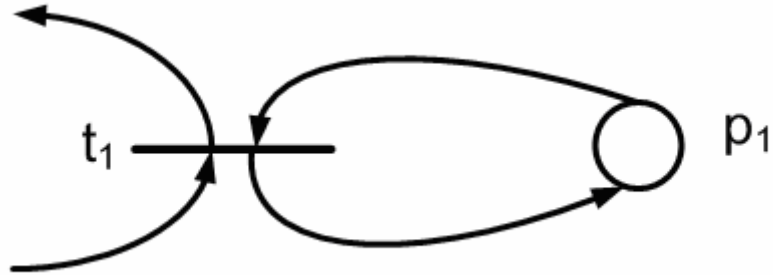
Bir PA : (P; T; A; W; M₀) Petri Ağı tanımlı olsun. Bu ağdaki birbirini izleyen her $p_i \in P$ ve $t_j \in T$ çiftinde aradaki dalın ağırlığı 1 ise bu durumda buna “Basit Petri Ağı” denir.[3]

$$w(p_i, t_j)=1 \text{ ve } w(t_j, p_i)=1, \forall p_i \in P \wedge \forall t_j \in T \text{ için}$$

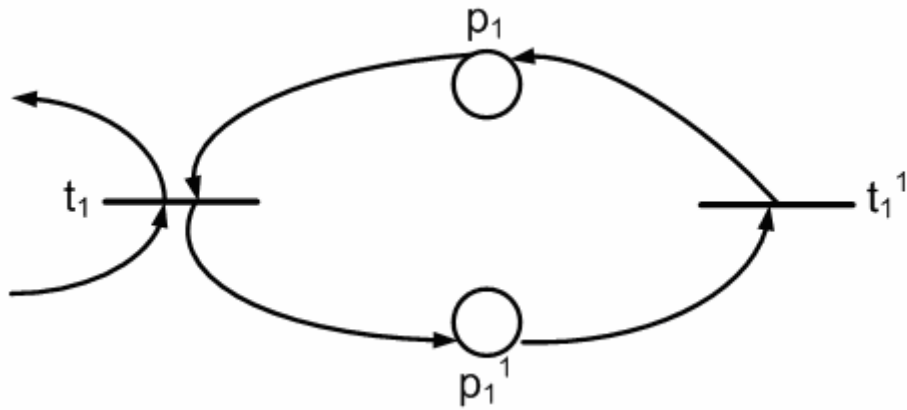
2.3.2 Temiz (Pure) Petri Ağları

Bir PA : (P; T; A; W; M₀) Petri Ağı tanımlı olsun. Bu ağda bir $p_i \in P$ yeri, $t_j \in T$ geçişin hem giriş fonksiyonunda hem de çıkış fonksiyonunda yer alıyorsa bu durumda ağda bir öz çevre vardır denir. Hiçbir öz çevre barındırmayan ağlara “Temiz Petri Ağı” denir.[3]

Eğer $w(p_i, t_j)=k$ ve $k > 1$ iken $w(t_j, p_i)=0$ veya $w(t_j, p_i)=k$ ve $k > 1$ iken $w(p_i, t_j)=0$ ise, $\forall p_i \in P \wedge \forall t_j \in T$ için, Petri Ağı “Temiz Petri Ağı”dır.



Şekil 2.8: Petri Ağı’nda öz çevrim

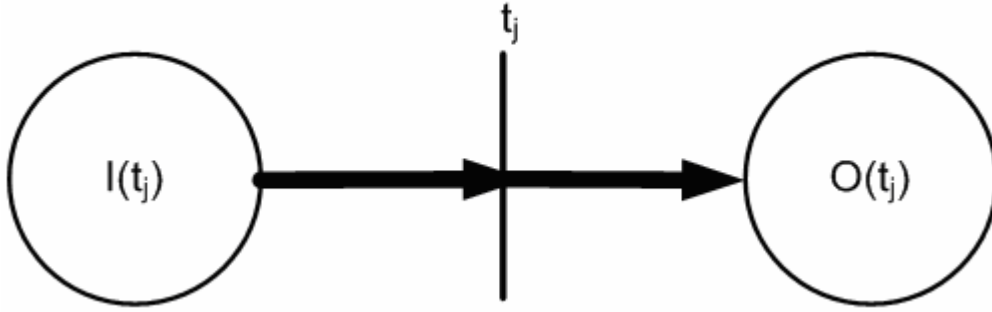


Şekil 2.9: Petri Ağı’nda öz çevrimin kaldırılması

2.3.3 İkili (Binary) Petri Ağları

Bir PA : (P; T; A; W; M₀) Petri Ağı tanımlı olsun. Herhangi bir anda, herhangi bir ateşleme sonrası Petri Ağı'nda tüm yerlerden sadece bir yerde bir jeton bulunabilen yada bulunabilecek şekilde işaretlenmiş ağa “İkili Petri Ağı” denir [3].

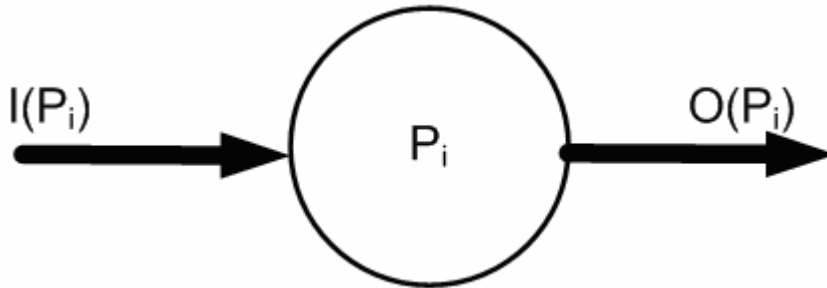
2.4 Klasik Petri Ağı Dinamik Davranışı



Şekil 2.10: Geçiş – yer ilişkisi

$$I(t_j) \subset P \wedge O(t_j) \subset P$$

Yani t_j geçişine izin veren yerler, t_j 'nin giriş yerleridir ve P yerler kümesinin alt kümesidir. Aynı şekilde t_j geçişinin çıkış yerleri de P'nin bir alt kümesidir.



Şekil 2.11: Yer – geçiş ilişkisi

$$I(P_i) \subset T \wedge O(P_i) \subset T$$

P_i yerinin giriş ve çıkış geçişleri T kümesinin bir alt kümesidir.

$I(t_j) = \{ p_i \in P ; (p_i, t_j) \in A \}$, t_j 'ye giriş olan p_i 'ler kümesi

$O(t_j) = \{ p_i \in P ; (t_j, p_i) \in A \}$, t_j 'ye çıkış olan p_i 'ler kümesi

$I(p_i) = \{ t_j \in T ; (t_j, p_i) \in A \}$, p_i 'ye giriş olan t_j 'ler kümesi

$O(p_i) = \{ t_j \in T ; (p_i, t_j) \in A \}$, p_i 'ye çıkış olan t_j 'ler kümesi

Eğer $p_i \notin I(t_j)$ ise $w(p_i, t_j) = 0$ dir. Yani p_i, t_j geçişinin giriş yeri değilse p_i 'den t_j 'ye ok ağırlığı sıfırdır.

Eğer $p_i \notin O(t_j)$ ise $w(t_j, p_i) = 0$ dir. Yani p_i, t_j geçişinin çıkış yeri değilse t_j 'den p_i 'ye ok ağırlığı sıfırdır.

Diğer bir gösterim ise

t geçişinin giriş yerleri $\bullet t := \{p \in P \mid I(t) > 0\}$

t geçişinin çıkış yerleri $t \bullet := \{p \in P \mid O(t) > 0\}$

p yerinin giriş geçişleri $\bullet p := \{t \in T \mid I(p) > 0\}$

p yerinin çıkış geçişleri $p \bullet := \{t \in T \mid O(p) > 0\}$

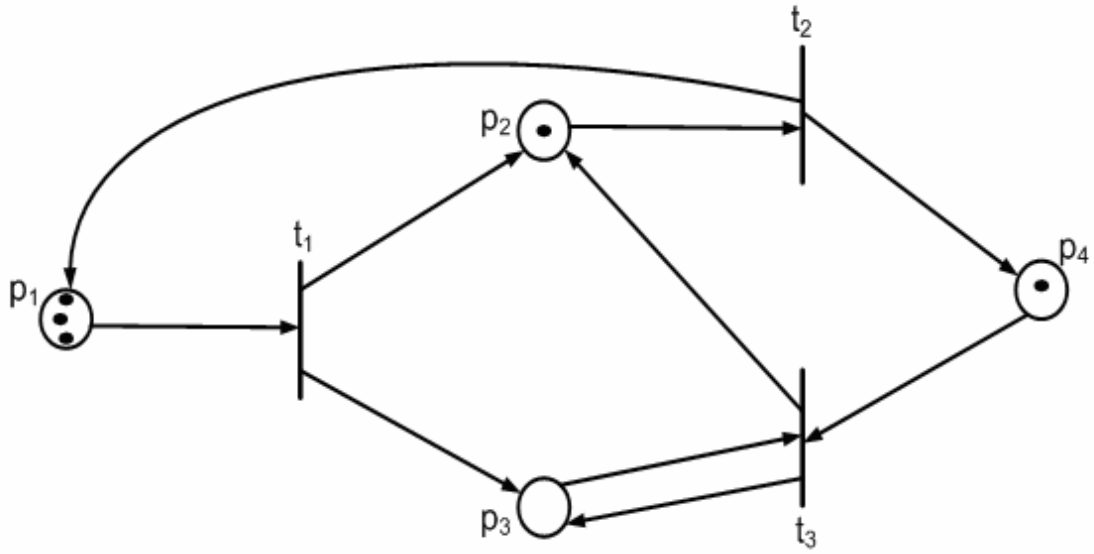
şeklindedir.

2.4.1 Petri Ağı'nın işaretlenmesi

Jetonlar Petri Ağı'nı dinamik kılan bileşenlerdir. Belirli bir zamanda Petri Ağı'nda yerlerin sahip olduğu jeton sayısına (ağın durumuna) “İşaretleme” denir. Petri Ağı'nın ilk andaki (daha hiçbir geçişin ateşlenmediği) işaretlemesine “Başlangıç Durumu İşaretleme” denir.

Petri Ağı'nın durum değiştirmesi işaretlemenin değişmesi dolayısıyla ateşlemenin/lerin olduğu anlamına gelir. İşaretleme M ile gösterilir ve $m \times 1$ boyutunda bir matristir, burada m ağdaki yerlerin sayısıdır, $m=|P|$.

M işaretlemesinin i . yerinin işaretlemesi $M(p_i)$ ile gösterilir ve p_i 'nci yer için jeton sayısını (işaretleme) gösterir. Sistemin başlangıç durumundaki işaretlemesi M_0 ile gösterilir.



Şekil 2.12: İşaretlenmiş Petri Ağı

$$M_0 = \begin{bmatrix} 3 \\ 1 \\ 0 \\ 1 \end{bmatrix}$$

2.4.2 Mümkün kılınmış geçiş

Bir Petri Ağı'nda $t_j \in T$ geçişinin mümkün kılınması demek ateşlenebilme kabiliyetine sahip olabilmesi demektir. Matematiksel olarak, $t_j \in T$ ve $\forall p_i \in I(t_j)$ ($\forall p_i \in \bullet t$) için

$$M(p_i) \geq w(p_i, t_j) \quad (2.1)$$

sağlıyorsa t_j geçişine etkin yani “mümkün kılınmış geçiş” denir. Yani geçişin giriş yerlerinde, en az bu yerlerden bu geçişe giden okların ağırlıklarında jeton var ise bu geçiş mümkün kılınmıştır.

Bir geçiş mümkün kılınmış ise “ateşlenebilir” denir.

2.4.3 Ateşleme dizisi

Petri Ağı'nın ateşleme dizisi (olay dizisi – oluş dizisi) sonlu sayıda geçişten oluşan bir kümedir ($\{t_1, t_2, t_3, \dots, t_n\}$, $n > 0$) öyle ki $M_1, \dots, M_{n+1}$ işaretlemeleri için $M_i[t_i > M_{i+1}, \forall i=1 \dots n$.

Ateşleme dizisi $\sigma = \{t_1, t_2, t_3, \dots, t_n\}$ için M işaretlemesinden elde edilen en son işaretleme kısaca $M_i[\sigma > M_{n+1}]$ ile gösterilir. Başlangıç durumundan erişilebilir tüm işaretlemeler kümesi $R(M_0)$ ile gösterilir

Boş küme ateşleme dizisi ε ile gösterilirse $M[\varepsilon > M]$ dir.

2.4.4 Erişilebilir işaretleme

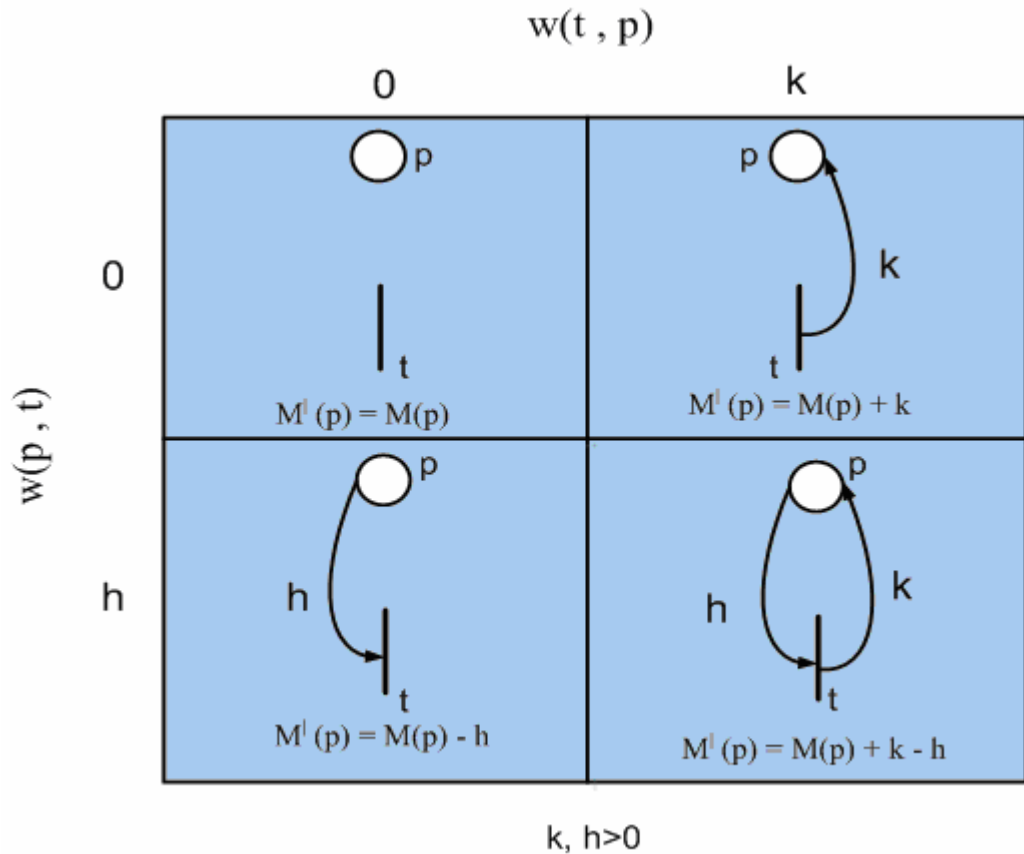
$t \in T$ geçişi M işaretlemesi altında ateşlenirse yeni bir işaretleme oluşur, M^t .

Bu yeni işaretleme;

$$M^t(p) = M(p) - w(p, t) + w(t, p), \quad \forall p \in P \quad (2.2)$$

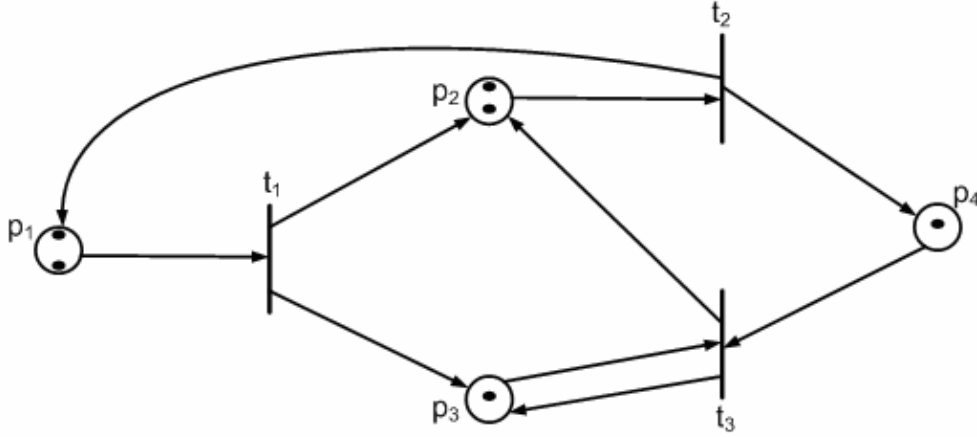
Şeklinde bulunur ve $M[t > M^t]$ şeklinde gösterilir. M^t işaretlemesi M işaretlemesinden erişilebilirdir.

Erişilebilir olduğunu göstermek üzere $M \rightarrow M^t$ yazılır.

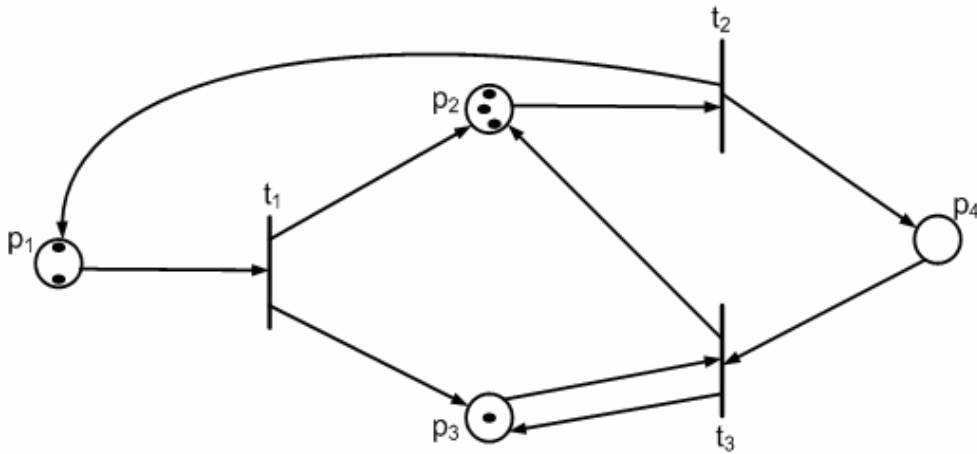


Şekil 2.13: İşaretlenmiş Petri Ağı'nda ateşleme sonrasında işaret değişimi

Ateşleme dizisi σ için $M[\sigma > M^0]$, M işaretlenmesi altında Petri Ağı'ndaki mümkün kılınmış geçişlerin ateşlenmesi halinde oluşacak ateşleme dizileri sonucunda varılan tüm yeni işaretlemelerin oluşturduğu kümeye erişilebilir işaretlemeler kümesi denir.



Şekil 2.14: Şekil 2.12'deki Petri Ağı'nda t_1 geçişi ateşlendikten sonra erişilen yeni durum $M_1 = [2 \ 2 \ 1 \ 1]^T$



Şekil 2.15 Şekil 2.14'daki Petri Ağı'nda t_3 geçişi ateşlendikten sonra erişilen yeni durum $M_2 = [2 \ 3 \ 1 \ 0]^T$

Ateşleme dizisi $\sigma = \{t_1, t_3\}$ için $M_2 = [2 \ 3 \ 1 \ 0]^T$ işaretlemesi $M_0 = [3 \ 1 \ 0 \ 1]^T$ işaretlemesinden erişilebilir bir işaretlemedir. $M_0[\sigma > M_2]$

2.4.5 Canlılık

Petri Ağı'ndaki canlılık tanımı kilitlenme yani ağın ölü olmasıyla ilişkili bir kavramdır. Petri Ağı'nın, başlangıç durumu işaretlenmesinden ağdaki hiçbir geçişin ateşlenemeyeceği işaretlemeye $M \in R(M_0)$ varmasıyla artık yeni bir ateşleme gerçekleşemez ve ağ ölü kilitlendi denir. Ölü kilitlenmenin olmadığı ağa 'canlıdır' denir [5].

2.4.6 Klasik Petri Ağı'nın matris analizi

Bir sistemin modellenmesinin amacı, bu sisteme ait matematiksel bir tanımlama yapmak, böylece modellenen sistemin yapısal analizini yapmak ve performans özelliklerini incelemektir. Bu yüzden Petri Ağları'nın matematiksel bir araç olarak kullanılmasını incelememiz gerekmektedir.

Oklar iki gruba ayrılabilir, giriş ve çıkış okları. Okun giriş veya çıkış oku olması ilgili geçişe göre alakalandırılacaktır.

1. $W(p,t): P \times T \rightarrow 0, 1, 2, \dots$ yerlerden geçişlere yönelmiş okları belirtmektedir. Örneğin $w(p_i, t_j) > 0$ ise p_i den t_j ye yönelmiş bir ok söz konusudur. $w(p_i, t_j) = k$, $k > 1$ ise okun yanına k ok ağırlığı etiketi eklenir.
2. $W(t,p): P \times T \rightarrow 0, 1, 2, \dots$ geçişlerden yerlere yönelmiş okları belirtmektedir. Örneğin $w(t_j, p_i) > 0$ ise t_j den p_i ye yönelmiş bir ok söz konusudur. $w(t_j, p_i) = k$, $k > 1$ ise okun yanına k ok ağırlığı etiketi eklenir.

Bir geçişin mümkün kılınabilmesi (ateşlenebilmesi) için gerekli şart denklem (2.1)'de verilmişti,

$$M(p_i) \geq w(p_i, t_j) ; t_j \in T \text{ ve } \forall p_i \in I(t_j) \quad (\forall p_i \in \bullet t)$$

ve ateşlenme sonrasında belirli bir yere ait yeni işaretleme denklem (2.2)'de verilmişti,

$$M^1(p) = M(p) - w(p,t) + w(t,p) \quad \forall p \in P$$

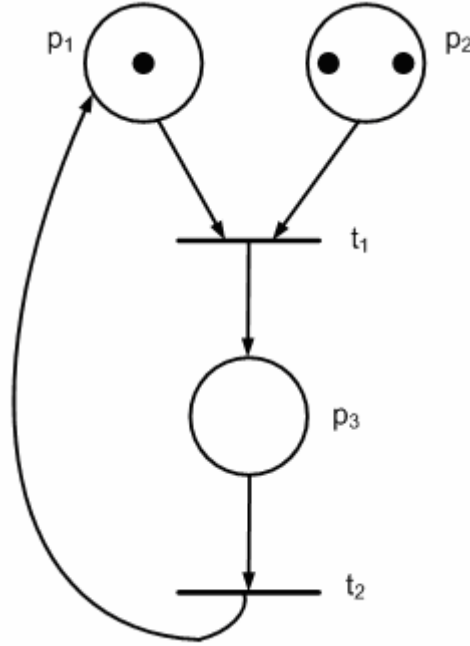
dir.

Bunlardan sonra Petri Ağı'nın yapı matrisini tanımlamamız gerekmektedir. m tane yere sahip n tane de geçişe sahip bir Petri Ağı'nın $m \times n$ boyutunda "yapı matrisi"

bulunmaktadır ve bu matris Petri Ağı'nın davranışını özetlemektedir. A matrisinin elemanları,

$$A(p_i, t_j) = w(t_j, p_i) - w(p_i, t_j) \quad (2.3)$$

Şeklinde hesaplanmaktadır. A: m x n boyutunda matristir.



Şekil 2.16: Örnek Petri Ağı

$$P = \{ p_1, p_2, p_3 \}, T = \{ t_1, t_2 \},$$

$$w(p_1, t_1) = w(p_2, t_1) = w(p_3, t_2) = 1$$

$$w(p_3, t_1) = w(p_1, t_2) = w(p_2, t_2) = 0$$

$$w(t_1, p_3) = w(t_2, p_1) = 1$$

$$w(t_1, p_1) = w(t_1, p_2) = w(t_2, p_3) = w(t_2, p_2) = 0$$

$$W(p, t) = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \quad W(t, p) = \begin{bmatrix} 0 & 1 \\ 0 & 0 \\ 1 & 0 \end{bmatrix}$$

$$A(p, t) = W(t, p) - W(p, t) = \begin{bmatrix} -1 & 1 \\ -1 & 0 \\ 1 & -1 \end{bmatrix} \quad M_0 = \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix}$$

U_k , k. ateşlenen geçişi gösteren matristir, örneğin

$$U_k = \begin{bmatrix} t_1 \\ t_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, t_1 \text{ ateşlenme için izinlidir; } t_2 \text{ ise izinsizdir, ateşlenemez.}$$

t_1 ateşlendiğinde p_1 yerindeki yeni işaretleme:

$$M^l(p_1) = M(p_1) - w(p_1, t_1) + w(t_1, p_1)$$

$$M^l(p_1) = 1 - 1 + 0$$

$$M^l(p_1) = 0$$

olarak bulunur.

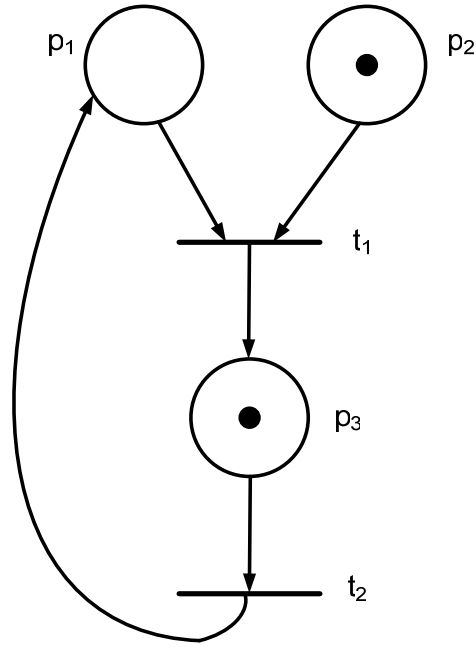
Fakat bunu tek tek her yer için yapmak yerine A yapı matrisi ve ateşlenen geçişi gösteren U matrisi ile tek adımda buluruz.

$$M_k = M_{k-1} + A U_k \quad (2.4)$$

Yukarıdaki örnek için M_0 durumunda t_1 ateşlenirse:

$$M_1 = M_0 + A \times U_0, \quad U_0 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \text{ ilk ateşlemeyi gösterir}$$

$$M_1 = \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix} + \begin{bmatrix} -1 & 1 \\ -1 & 0 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix} \text{ bulunur.}$$



Şekil 2.17: Şekil 2.16’teki işaretleme için t_1 geçişi ateşlendikten sonraki yeni işaretleme

2.4.6.1 P-Değişmezleri

Bahsedildiği gibi ateşleme sonrası ağıta ait tüm yerlerin işaretlenmesi denklem (2.4) ile bulunabilmektedir.

$$M_k = M_{k-1} + A U_k$$

M_0 ’a σ ateşleme dizi uygulandıktan sonra yeni elde edilen işaretleme $M_0 [\sigma > M_n$ şeklinde ifade edilir. σ , 1’den n ’e kadar ateşlemelerden oluşsun ve bu ateşlemeler için

$$\sum_{k=1}^n U_k = f_n$$

olsun

$$M_n = M_0 + A f_n$$

elde edilir.

Bu fonksiyonun her iki tarafını X matrisi ile çarparsak sonuç değişmez.

$$X M_n = X M_0 + X A f_n$$

X öyle seçilir ki $X A=0$ dır. Dolayısıyla

$$X M_n = X M_0 \text{ olur.}$$

Bu eşitliği sağlayan tüm X değişmezleri “P Değişmezleri” olarak adlandırılır [6]. Bunun sebebi tüm X ler için ilgili yerlerdeki jeton sayısı toplamının sabit olmasıdır. Hangi ateşleme dizisi seçilirse seçilsin P- Değişmezleri tarafından kapsanan yerlerdeki jeton sayısı toplamı değişmez [5].

Şekil 2.16’deki örneğimizi incelersek

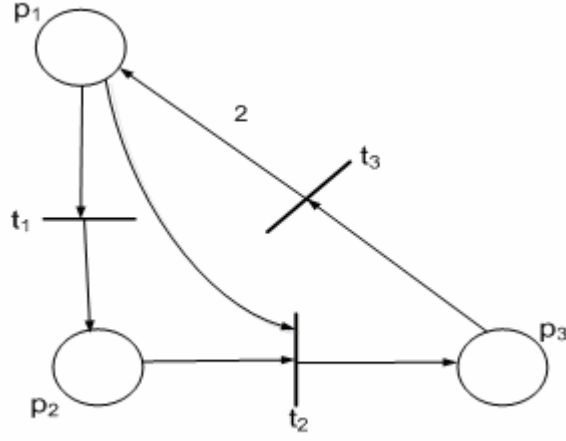
$$A(p,t)= W(t,p)- W(p,t)= \begin{bmatrix} -1 & 1 \\ -1 & 0 \\ 1 & -1 \end{bmatrix} \text{ idi}$$

$$X A=0 \text{ ise } X = [1 \ 0 \ 1] \text{ dir.}$$

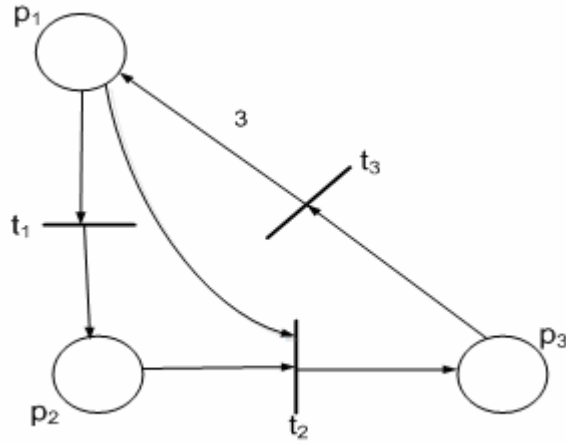
$$X M_1 = X M_0 \text{ ise } [1 \ 0 \ 1] \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} = [1 \ 0 \ 1] \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix} = 1 \text{ dir.}$$

P-Değişmezleri analizi, Petri Ağı’nın kapasite sınırlamalarının doğruluğunun kontrolünde çok yardımcı olmaktadır. Yukarıdaki örneğe bakacak olursak p_1 ve p_3 yerlerindeki jeton sayısı ile P değişmezleri çarpımı sabittir. Bu örnek için p_1 ve p_3 yerlerinden birisindeki jeton sayısı 0 olursa diğer yerdeki jeton sayısı maksimum 1 olabilir [5].

P değişmezleriyle ilgili küçük bir örnek de aşağıda gösterilmiştir. Şekil 2.18(a) P değişmez bir Petri Ağıyken, Şekil 2.18(b) P değişmez bir ağ değildir, çünkü p_1 veya p_2 yerinde her zaman fazladan jeton birikecektir [5].



(a)



(b)

Şekil 2.18: (a) P değişmez Petri Ağı (b) P değişmez olmayan Petri Ağı

Şekil 2.18(a) için $p_1 + p_2 + (2 \times p_3) = k$, $k < +\infty$ dır. Yani P değişmezleri matrisi $X = [1 \ 1 \ 2]$ dir.

$$A = \begin{bmatrix} -1 & -1 & 2 \\ 1 & -1 & 0 \\ 0 & 1 & -1 \end{bmatrix}, \text{ P değişmezlik şartı } X A = 0 \text{ idi,}$$

$$X A = [1 \ 1 \ 2] \begin{bmatrix} -1 & -1 & 2 \\ 1 & -1 & 0 \\ 0 & 1 & -1 \end{bmatrix} = 0$$

Ayrıca $X M = X M'$ sağlanmalıdır.

Örneğin $M = \begin{bmatrix} 0 \\ 1 \\ 3 \end{bmatrix}$ işaretlemesi için

$$X M = \begin{bmatrix} 1 & 1 & 2 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 3 \end{bmatrix} = 7 \text{ dir.}$$

M işaretlemesi altında iken t_3 geçişi ateşlenir ise oluşacak yeni işaretleme $M' = \begin{bmatrix} 2 \\ 1 \\ 2 \end{bmatrix}$ olmaktadır.

$$X M' = \begin{bmatrix} 1 & 1 & 2 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \\ 2 \end{bmatrix} = 7 \text{ dir.}$$

Yani Petri Ağı'nın P değişmezleri matrisi ile ağın işaretlemesini gösteren matrislerinin çarpımı her zaman birbirine eşittir. Fakat Şekil 2.18(b) için böyle bir eşitliği sağlayan X matrisi bulunamaz, bu da demektir ki bu Petri Ağı için jeton korunumu söz konusu değildir. Ağ grafından da görüldüğü üzere p_1 yeri t_3 ateşlenmesiyle jeton kazanmakta, t_1 ve t_2 nin ateşlenmesi ile jeton kaybetmektedir. t_3 ateşlenmesi ile 3 jeton kazanmakta, t_1 ve t_2 ateşlenmesi ile 1'er tane jeton kaybetmektedir. Dolayısıyla p_1 yerinde her t_3 ateşlenmesiyle fazladan 1 jeton ağa eklenmektedir. Bu jeton eklenmesi ağın jeton kapasitesini sonsuza çıkarmaktadır. Bu tip ağlara “Sınırsız Ağlar” denir. Ağın jeton korunumlu olmadığı P değişmezlerinin bulunmaması ile görülür.

2.4.6.2 T-Değişmezleri

Herhangi bir M işaretlemesi için

$$M = M_0 + A f$$

İfadesini elde etmiştik. Öyle bir ateşleme dizileri toplamı f olsun ki $A f = 0$ olsun.

$$M = M_0 \text{ olur}$$

Burada elde edilebilecek tüm f değerlerine T- Değişmezleri denir[6]. T – Değişmezleri analizi ile Petri Ağı’nın tersine çevrilebilir olup olmadığı yani başlangıç durumuna dönüp dönemeyeceği incelenir.

Şekil 2.17’deki örnek için T değişmezleri bulunamaz.

$$A = \begin{bmatrix} -1 & 1 \\ -1 & 0 \\ 1 & -1 \end{bmatrix}, \quad A f = 0, \quad \begin{bmatrix} -1 & 1 \\ -1 & 0 \\ 1 & -1 \end{bmatrix} f = 0 \text{ ‘ı sağlayan bir } f \text{ değeri bulunamaz ki bu, ağın}$$

başlangıç durumuna hiçbir zaman dönemeyeceğini söyler. Şekilden de görüleceği gibi p_2 yerindeki jeton sayısı azalır ve bir daha hiç artma göstermez, kilitlenme durumuna girer.

T değişmezlerinin varlığı Petri Ağı’nın tersine çevrilebilir olmasını göstermek için gerekli bir şarttır. Petri Ağı tersine çevrilebilir ise T değişmezleri mevcuttur, fakat T değişmezlerinin varlığı Petri Ağı’nın tersine çevrilebilir olduğunu ispatlamaz. Petri Ağı’nın tersine çevrilebilirliği $M \in R(M_0)$ iken $M_0 \in R(M)$ olmasıyla tanımlanabilir[8].

2.4.7 Erişilebilirlik ağacı

Petri Ağları’nın birçok özelliği ağın ‘Erişilebilirlik Grafiği (Ağacı)’ ile incelenebilir. Erişilebilirlik ağacı Petri Ağı’nın erişilebileceği işaretlemeleri göstermektedir. Petri Ağıyla ilgili en önemli soru herhangi bir işaretleme M ’nin herhangi bir ateşleme dizisi ile erişilebilir olup olmadığıdır.

M_0 başlangıç durumundan başlayarak herhangi bir ateşleme dizisi ile ulaşılabilen tüm $M \in R(M_0)$ işaretlemelerini gösteren grafiksel gösterime “Erişilebilirlik Ağacı” veya “Erişilebilirlik Grafi” denir.

$$M = M_0 + Au_1 + Au_2 + \dots + Au_k$$

$$M = M_0 + \sum_{n=1}^k Au_n$$

$$R(M_0) = \{M \mid M \text{ erişilebilir bir işaretlemedir}\}$$

Erişilebilirlik ağacı ile erişilebilir durumlar ve aralarındaki ateşleme dizileri gözlenebilir.

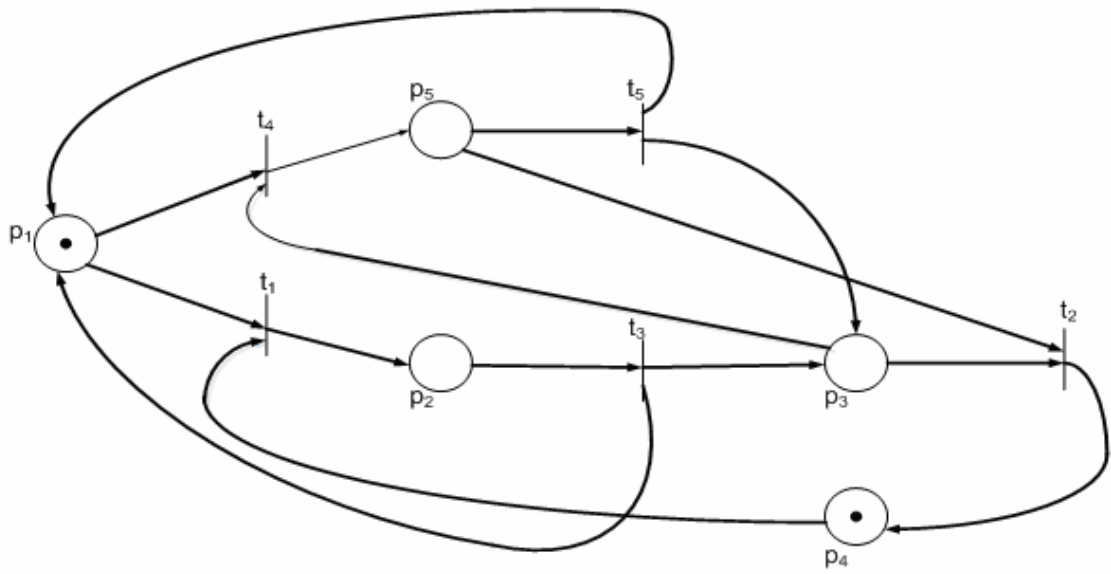
Erişilebilirlik ağacı $R_T = \{R(M_0), I, f, T\}$ kümesi ile gösterilebilir.

$R(M_0)$: Erişilebilir durumları

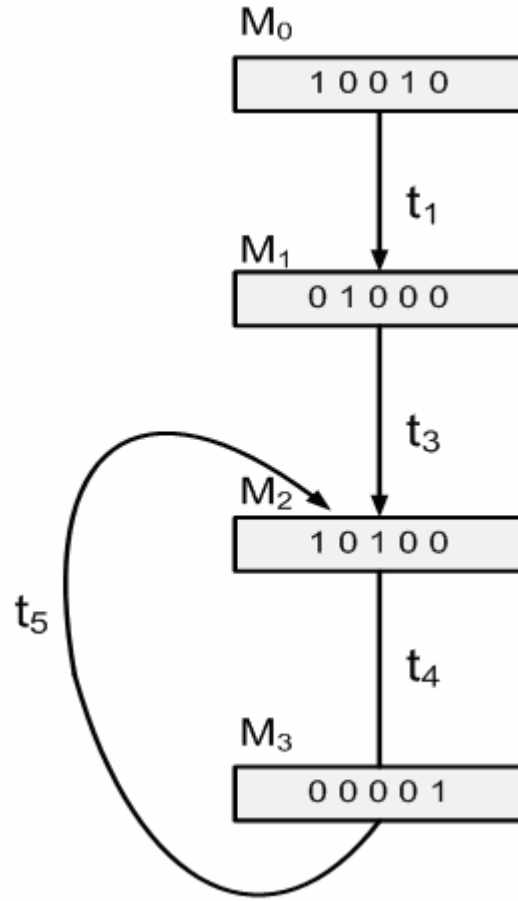
$I: R(M_0) \times R(M_0)$, erişilebilir durumlar arasındaki ilişki okları kümesi

$f: I \longrightarrow T$, her ok için etiketleme fonksiyonudur. Okun çıktığı işaretlemeden girdiği işaretlemeye hangi geçişin ateşlenmesi ile geçtiğini gösterir. (M_i, t_j, M_k) , üçlü kümesiyle gösterilir ve M_i işaretlemesindeyken t_j geçişi ateşlenmesi ile M_k işaretlemesine varılır şeklinde okunur.

T : ağaç üzerinde gösterilen okların kümesidir.



Şekil 2.19: Erişilebilirlik ağacı için örnek Petri Ağı



Şekil 2.20: Şekil 2.19'deki örnek Petri Ağı'nın erişebilirlik ağacı

3. RENKLİ PETRİ AĞLARI

Bu bölümde Renkli Petri Ağları (RPA)'nı tanıtılacaktır. İlk olarak 1970'lerin sonralarına doğru K.Jensen[2] tarafından tanımlanan ve jetonlara etiketlenen renk bilgisini temel alan Renkli Petri Ağlarıyla, Klasik Petri Ağları ile çok daha fazla yer ve geçiş ile ifade edilen modellemeleri daha az sayıda yer ve geçiş ile modellemek amaçlanmıştır. Böylece sayıca daha az elemanlı ve renklere göre davranışı değişebilen ağlar elde edilmektedir.

Renkli Petri Ağları, Klasik Petri Ağları'nın genişletilmiş hali olup büyük grafiksel sunuma sahip sistemlerin toparlanmış sunumunu yapmakta kullanılır. Jetonlar ve geçişler için kullanılan renkler ile büyük bir Petri Ağı sıkıştırılıp çok daha küçük fakat aynı özellikteki sistemi tanımlayan bir RPA elde edilir.[9]

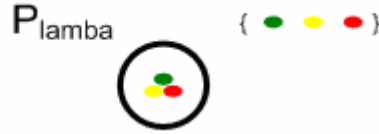
Jetonları birbirinden farklılaştırmak için jetonlara ilişik farklı renkler kullanılır. Her yer kendisi ile ilgili jeton sayısı veya jetonların renkleri ile ilişkilendirilir. Aynı şekilde her geçişte kendisiyle ilgili olan renklere göre davranış gösterir ve bu ilişkiye göre ateşleme yapılabilir.

Renkli Petri Ağlarına kadar jetonlar hep içi siyah renkli bir daire olarak düşünülmüştür. Örneğin bir trafik ışığının modellenmesi istenildiğinde, bu trafik ışığının 3 ayrı renk için ayrı lambalarla görevini yerine getirdiği düşünülürse, kırmızı, yeşil ve sarı renk için 3 ayrı yer ve bu yerler arasında da geçişlerden oluşacak bir modelleme yapılacaktır.



Şekil 3.1: Trafik ışıkları için durumların Petri Ağı notasyonu gösterimi

Şimdiyse tek bir lambanın üç rengi de sırasıyla gösterdiği bir trafik lambası göz önüne alınsın. Böyle bir durumda jetonun hangi trafik ışığı rengini gösterdiği bilinemez çünkü jeton tek bir yerde tek bir renktedir. Oysa ki bir önceki lambada her ışık için ayrı lambalar ayrı yerler ile gösterilmişti, burada ise tek bir lamba ve üç ayrı ışık bilgisi bulunmalıdır. Dolayısıyla tek bir tip jeton bu bilgiyi ayırt etme yeteneği veremez.

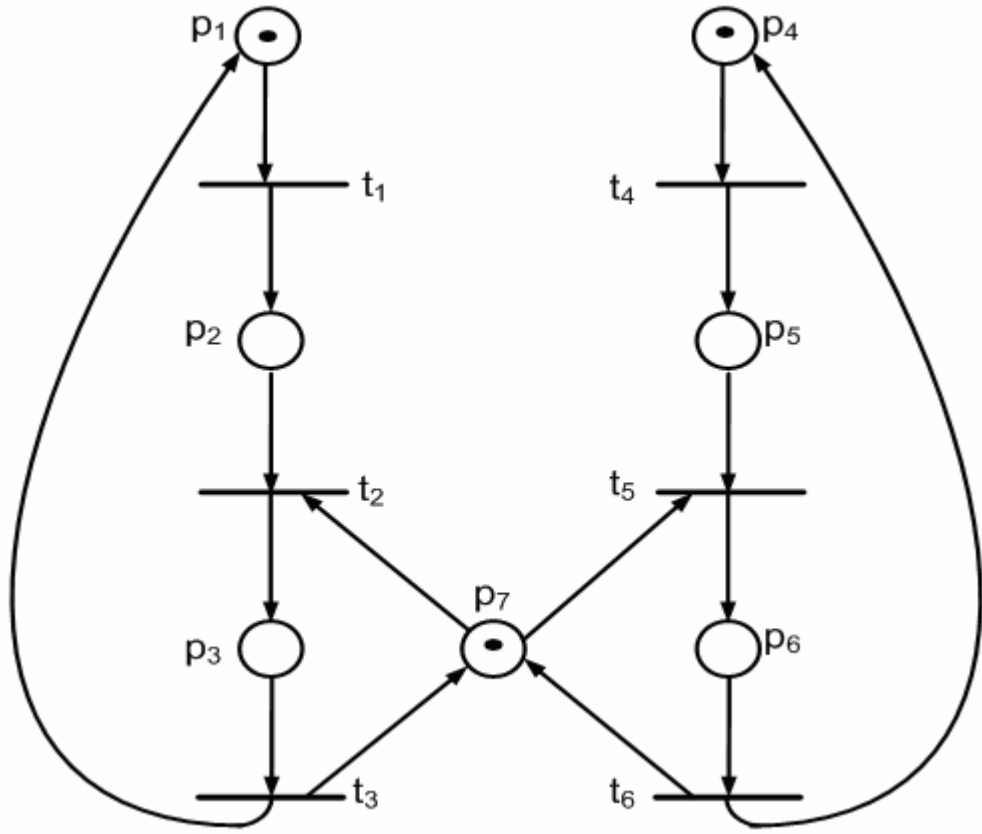


Şekil 3.2: Trafik ışıkları için durumların Renkli Petri Ağı notasyonu gösterimi

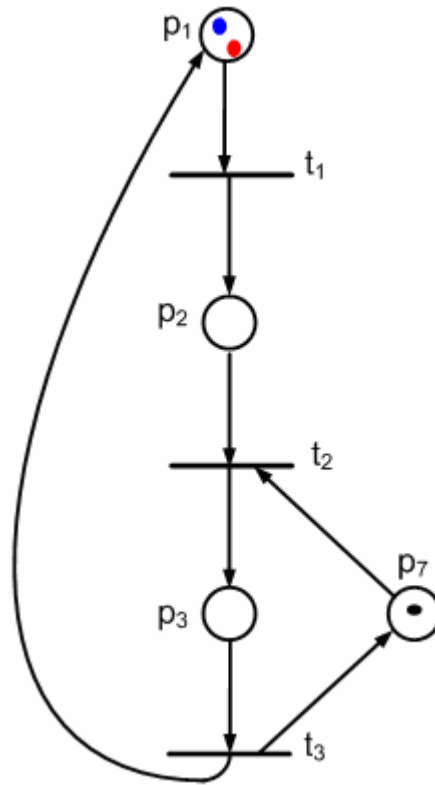
Bu yüzden farklı tiplerde jetonlar ile bu ayırt etme yetisi ağa verilir. Bir yerin sahip olabileceği jetonların türleri yerin sağ üst kısmına $\{ \dots \}$ arasına yazılabilir.

Tüm işlemlerin aynı fakat birbirlerinden bağımsız oldukları bir sistemde geçişlerin renkleri direkt olarak işlemleri gösterecek ve geçişler ile yerlerin renkleri işlemler ile özdeş olacaktır.[10]

Örneğin Şekil 3.3’de [11] iki benzer aşamalı işlemi farklı ürünler için yapan bir Klasik Petri Ağı örneği bulunmaktadır. Yukarıda sözü edilen birbirinden bağımsız fakat benzer işlemlerin bulunduğu bu Klasik Petri Ağı’nda işlemler farklı renklerle gösterilerek aynı yerlerde birbirlerinden ayrılabilirler.



Şekil 3.3: Çift işlemcili üretim sistemi KPA modeli[10]



Şekil 3.4: Şekil 3.2'deki ağın RPA modeli[10]

Rekli jetonlar, Petri Ağı'nın birden fazla aynı problemi modellerken ağın büyümesini engeller.

Geçiş renkleri ile birçok geçiş birarada gösterilebilir. Böylece geçiş renkleri ile benzer sistem elemanlarının davranışları bir geçişle gösterilebilir ve jetonlar renklerine göre farklı davranışlarda bulunabilir.

Ok ağırlıkları ve oklardan geçecek jeton renk bilgisi ilgili geçiş ve rengine bağlı olarak değişebilmektedir.

3.1 Renkli Petri Ağı Tanımı ve Özellikleri

Bu bölümde ilk olarak RPA'nın matematiksel modeli tanımlanacaktır. RPA yedi bileşenli bir küme ile tanımlanır.[12]

RPA: $(P, T, N, O, C, \Omega, M_0)$

P: Sonlu sayıda yerler kümesi

T: Sonlu sayıda geçişler kümesi

N: $N(p_i, t_j):C(t_j) \longrightarrow N(C(p_i))$, t_j geçişine $C(t_j)$ rengi için giriş yeri olan p_i yerinden çıkan $C(p_i)$ renkli ok ağırlığını gösteren matris, giriş matrisi de denir.

O: $O(p_i, t_j):C(t_j) \longrightarrow N(C(p_i))$, t_j geçişinden $C(t_j)$ rengi için çıkış yeri olan p_i yerine giren $C(p_i)$ renkli ok ağırlığını gösteren matris, çıkış matrisi de denir.

C: $T \cup P \longrightarrow \Omega$ Renklendirme fonksiyonu

$C(P)$: Yerlerin sonlu sayıda renk kümesi

$C(T)$: Geçişlerin sonlu sayıda renk kümesi

Ω : Sonlu sayıda renkler kümesi

M_0 : Başlangıç işaretlemesi

$p_i \in P, t_j \in T, \zeta \in C(T)$ ve $c \in C(P)$ dir.

$N(p_i, t_j)(\zeta)$: $1 \times |C(P)|$ boyutunda, t_j geçişi ζ geçiş rengi için ateşlenmesi halinde p_i yerinden t_j geçişine yönelmiş ok / oklar 'ın sayısını gösteren bir matristir.

$N(p_i, t_j)(\zeta, c)$: t_j geçişi ζ geçiş rengi için ateşlenmesi halinde c rengine p_i yerinden t_j geçişine yönelmiş ok sayısını gösterir.

$O(p_i, t_j)(\zeta)$: $1 \times |C(P)|$ boyutunda, t_j geçişi ζ geçiş rengi için ateşlenmesi halinde t_j geçişinden p_i yerine yönelmiş ok / oklar 'ın sayısını gösteren bir matristir ve $p_i \in P$, $t_j \in T$, $n \in C(T)$ dir.

$O(p_i, t_j)(\zeta, c)$: t_j geçişi ζ geçiş rengi için ateşlenmesi halinde c renginde t_j geçişinden p_i yerine yönelmiş jeton sayısını gösterir.

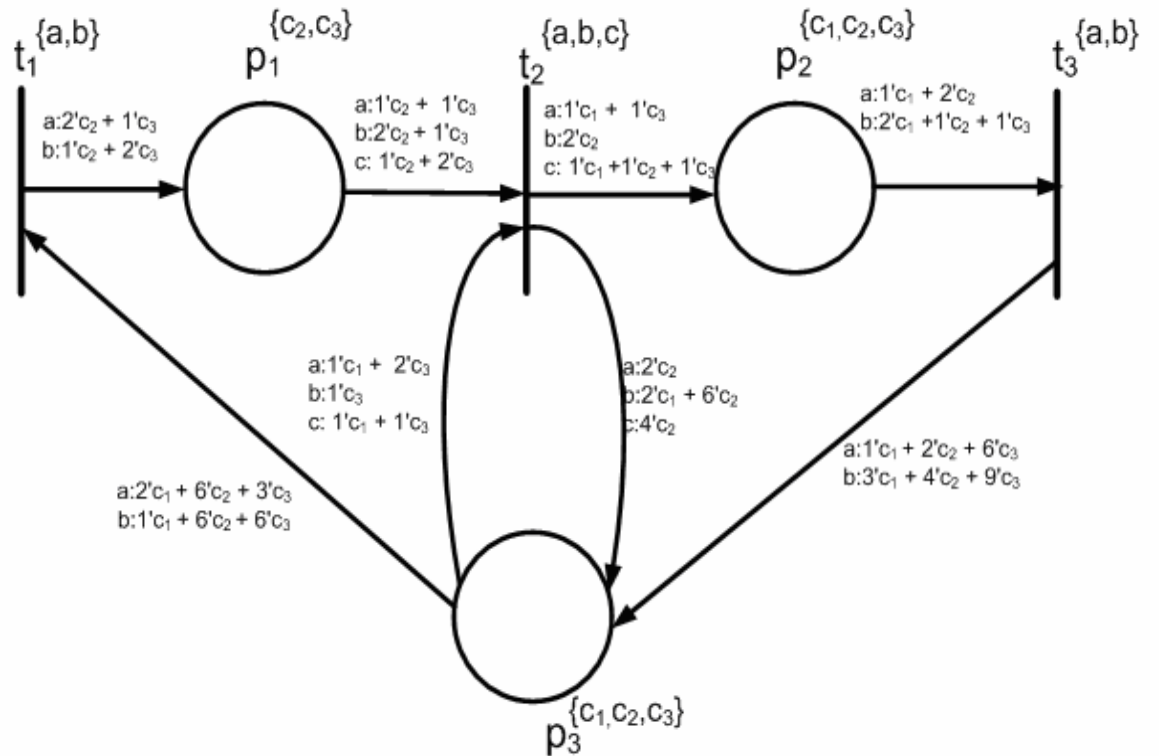
$N(p, t)(\zeta, c)$ kısaca $N(\zeta, c)$ ve $O(p, t)(\zeta, c)$ kısaca $O(\zeta, c)$ şeklinde gösterilir.

$N(\zeta, c)$: $|P| \times |T|$ boyutunda ζ geçiş rengi için yerlerden geçişlere c rengindeki ok ağırlık sayısını veren matristir.

$O(\zeta, c)$: $|P| \times |T|$ boyutunda ζ geçiş rengi için geçişlerden yerlere c rengindeki ok ağırlık sayısını veren matristir.

Bir RPA' da $m = |C(T)| \times |C(P)|$ tane $N(\zeta, c)$ ve $O(\zeta, c)$ matrisi vardır.

Örnek bir RPA[13]:



Şekil 3.5: Renkli Petri Ağı örneği

3.2 RPA'nın İşaretlenmesi

$M: P \longrightarrow N^{|P| \times |C(P)|}$, N doğal sayılar

$$M = [M(p_1), M(p_2), \dots, M(p_n)]^T, n=|P|$$

$M(p)$: $1 \times |C(P)|$ Boyutlu matrisle gösterilir ve p yerinde bulunan $c \in C(P)$ rengindeki jeton sayısını gösteren matristir.

Dolayısıyla M işaret matrisi $|P| \times |C(P)|$ boyutunda bir matristir. $M(p, c)$: $p \in P$ yerindeki $c \in C(P)$ renkli jetonun sayısını verir.

Örneğin Şekil 3.5 için

$C(P) = \{c_1, c_2, c_3\}$, $P = \{p_1, p_2, p_3\}$, $C(T) = \{a, b, c\}$, $T = \{t_1, t_2, t_3\}$ dir.

$M(p_1) = [0 \ 2 \ 3]$, $M(p_2) = [9 \ 9 \ 1]$, $M(p_3) = [7 \ 6 \ 5]$ olsun

$$M = [M(p_1), M(p_2), M(p_3)]^T = \begin{bmatrix} M(p_1) \\ M(p_2) \\ M(p_3) \end{bmatrix} = \begin{bmatrix} 0 \ 2 \ 3 \\ 9 \ 9 \ 1 \\ 7 \ 6 \ 5 \end{bmatrix}$$

olur.

$$M(p_1, c_1) = 0, M(p_1, c_2) = 2, M(p_1, c_3) = 3$$

$$M(p_2, c_1) = 9, M(p_2, c_2) = 9, M(p_2, c_3) = 1$$

$$M(p_3, c_1) = 7, M(p_3, c_2) = 6, M(p_3, c_3) = 5$$

3.3 RPA'da Geçişin Mümkün Kılınması

$t \in T$ geçişinin mümkün kılınması sadece ve sadece

$$M(p, c) \geq N(p, t)(\zeta, c) \quad (3.1)$$

şartının $N(p, t)(\zeta, c) > 0$, $t \in T$, $\zeta \in C(T)$, $\forall p \in P$, $\forall c \in C(P)$ için sağlanması ile olur. Şekil 3.5'deki örnek için giriş ve ok ağırlık matrisleri aşağıdaki gibi tanımlanmış olsun:

$$N(p_1, t_1)(a) = [0 \ 0 \ 0], \quad N(p_1, t_1)(b) = [0 \ 0 \ 0], \quad N(p_2, t_1)(a) = [0 \ 0 \ 0], \quad N(p_2, t_1)(b) = [0 \ 0 \ 0], \\ N(p_3, t_1)(a) = [2 \ 6 \ 3], \quad N(p_3, t_1)(b) = [1 \ 6 \ 6]$$

$$\begin{aligned} N(p_1, t_2)(a) &= [011], & N(p_1, t_2)(b) &= [021], & N(p_1, t_2)(c) &= [021], & N(p_2, t_2)(a) &= [000], \\ N(p_2, t_2)(b) &= [000], & N(p_2, t_2)(c) &= [000], & N(p_3, t_2)(a) &= [102], & N(p_3, t_2)(b) &= [001], \\ N(p_3, t_2)(c) &= [101] \end{aligned}$$

$$\begin{aligned} N(p_1, t_3)(a) &= [000], & N(p_1, t_3)(b) &= [000], & N(p_2, t_3)(a) &= [120], & N(p_2, t_3)(b) &= [211], \\ N(p_3, t_3)(a) &= [000], & N(p_3, t_3)(b) &= [000] \end{aligned}$$

$$\begin{aligned} O(p_1, t_1)(a) &= [021], & O(p_1, t_1)(b) &= [012], & O(p_2, t_1)(a) &= [000], & O(p_2, t_1)(b) &= [000], \\ O(p_3, t_1)(a) &= [000], & O(p_3, t_1)(b) &= [000] \end{aligned}$$

$$\begin{aligned} O(p_1, t_2)(a) &= [000], & O(p_1, t_2)(b) &= [000], & O(p_1, t_2)(c) &= [000], & O(p_2, t_2)(a) &= [101], \\ O(p_2, t_2)(b) &= [020], & O(p_2, t_2)(c) &= [111], & O(p_3, t_2)(a) &= [020], & O(p_3, t_2)(b) &= [260], \\ O(p_3, t_2)(c) &= [040] \end{aligned}$$

$$\begin{aligned} O(p_1, t_3)(a) &= [000], & O(p_1, t_3)(b) &= [000], & O(p_2, t_3)(a) &= [000], & O(p_2, t_3)(b) &= [000], \\ O(p_3, t_3)(a) &= [126], & O(p_3, t_3)(b) &= [349] \end{aligned}$$

t_2 nin ateşlenme şartı sağlanıyor mu? İncelensin.

$\zeta = a$ olsun

t_2 nin a rengi için giriş matrislerinden ilgili olanlara bakalım,

$$N(p_1, t_2)(a) = [011], N(p_2, t_2)(a) = [000], N(p_3, t_2)(a) = [102]$$

$M(p, c) \geq N(p, t)(\zeta, c)$ şartı ve başlangıç durumu işaretlememiz de

$$M = [M(p_1), M(p_2), M(p_3)]^T = \begin{bmatrix} M(p_1) \\ M(p_2) \\ M(p_3) \end{bmatrix} = \begin{bmatrix} 023 \\ 991 \\ 756 \end{bmatrix} \text{ idi.}$$

p_1 yeri şartı sağlıyor mu?

$$N(p_1, t_2)(a) = [011],$$

$$M(p_1, c_1) \geq ? N(p_1, t_2)(a, c_1) \rightarrow 0 \geq ?0 \rightarrow \text{doğru}$$

$$M(p_1, c_2) \geq ? N(p_1, t_2)(a, c_2) \rightarrow 2 \geq ?1 \rightarrow \text{doğru}$$

$$M(p_1, c_3) \geq ? N(p_1, t_2)(a, c_3) \rightarrow 3 \geq ?1 \rightarrow \text{doğru}$$

p_2 yeri şartı sağlıyor mu?

$N(p_2, t_2)(a) = [000]$ yani p_2, t_2 'nin giriş yeri değildir ve ateşlenme şartını bozmaz.

p_3 yeri şartı sağlıyor mu?

$$N(p_3, t_2)(a) = [102]$$

$$M(p_3, c_1) \geq ? N(p_3, t_2)(a, c_1) \rightarrow 7 \geq ?1 \rightarrow \text{doğru}$$

$$M(p_3, c_2) \geq ? N(p_3, t_2)(a, c_2) \rightarrow 6 \geq ?0 \rightarrow \text{doğru}$$

$$M(p_3, c_3) \geq ? N(p_3, t_2)(a, c_3) \rightarrow 5 \geq ?2 \rightarrow \text{doğru}$$

t_2 a rengi için mümkün kılınmıştır, ateşlenebilir.

$$\text{İşaretleme } M = \begin{bmatrix} M(p_1) \\ M(p_2) \\ M(p_3) \end{bmatrix} = \begin{bmatrix} 003 \\ 991 \\ 756 \end{bmatrix} \text{ olsaydı } M(p_1, c_2) \geq ? N(p_1, t_2)(a, c_2) \rightarrow 0 \geq ?1 \rightarrow$$

yanlış olacağından t_2 geçişi a rengi için mümkün kılınamazdı, ateşlenemezdi.

M işaretleme altında t geçişinin $\zeta \in C(T)$ rengi için ateşlenmesi sonrasında yeni

işaretleme $M^|, \forall p \in P, \forall c \in C(P)$ için $M \xrightarrow{t, \zeta} M'$ şeklinde gösterilebilir ve

$$M^|(p, c) = M(p, c) - N(p, t)(\zeta, c) + O(p, t)(\zeta, c) \quad (3.2)$$

dir.

RPA'da ateşleme dizisi $\sigma = \{(t_i, \zeta_i), (t_n, \zeta_m) \dots\}$ şeklinde geçiş ve ateşlendiği renk çiftleri halinde gösterilir.

Ateşleme dizisi $\sigma = \{(t_i, \zeta_i), (t_n, \zeta_m) \dots\}$ için M işaretlemeinden elde edilen işaretleme kısaca $M^||, M[\sigma > M^||$ ile gösterilir.

U, sistem için izinli yani ateşlenebilir geçişi ve ateşleneceği rengi veren $|T| \times |C(T)|$ boyutunda bir dizidir. Aynı anda sadece bir ateşlemenin mümkün olduğu varsayılan sistemlerde, U matrisinde sadece bir tane "1" bulunur bu da ateşlenecek geçişi (satır) ve rengini (sütun) verir.

$A(p, t)(\zeta, c)$, RPA'nın yapı matrisi;

$$A(p, t)(\zeta, c) = O(p, t)(\zeta, c) - N(p, t)(\zeta, c) \quad (3.3)$$

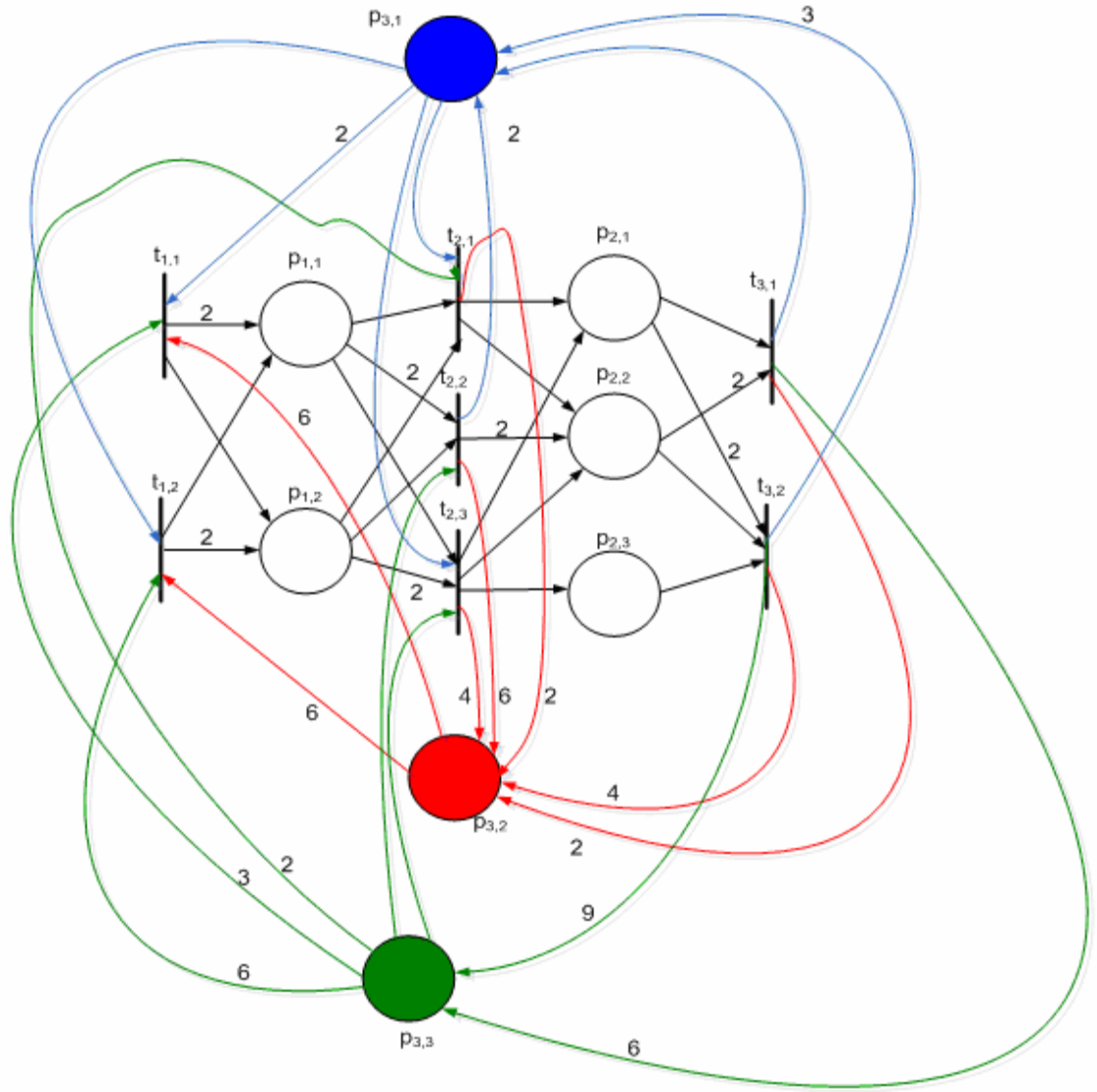
$$M' = M + \begin{bmatrix} (A(p,t)(\zeta, c_1) \ U(\zeta))^T \\ \dots \\ \dots \\ (A(p,t)(\zeta, c_{|C(P)|}) \ U(\zeta))^T \end{bmatrix}^T \quad (3.5)$$

İle bulunur.

RPA'da M'den başlayıp $M^{\parallel}$ işaretlemesine ulaştırarak bir ateşleme dizisi mevcut ise ya da diğer bir deyişle RPA bir ateşleme dizisi ile M den $M^{\parallel}$ işaretlemesine ulaşabiliyorsa $M^{\parallel}$ erişilebilir bir işaretlemedir.

Başlangıç durumundan erişilebilen tüm işaretlemelerin bulunduğu kümeye $R(RPA, M_0)$ erişilebilir işaretlemeler kümesi denir. RPA:incelenen Petri Ağı

$M_x \in R(RPA, M_0)$, ancak bu M_x işaretlemesinde $\forall t \in T$ geçişleri ateşlemez ise RPA ölü kilitlemiştir denir.



Şekil 3.6: Şekil 3.5'deki RPA'nın Klasik Petri Ağı'na açılmış hali[13]

Şekil 3.6, Şekil 3.5'deki RPA'nın KPA'ya dönüştürülmesini göstermektedir. Açıkta ki RPA ile yer ve geçiş sayısı azaltılmakta ve ağın takip edilebilmesi daha kolay olmaktadır.

RPA'ların KPA'lara veya KPA'ların RPA'lara dönüştürülmesiyle ilgili prosedürler literatürde verilmiştir [9].

RPA'yı KPA'ya açmak için izlenecek yol aşağıdaki gibidir ;

- Tüm yerlerin, kendilerinde olabilecek jeton renk sayısı kadar çoğaltılması
- Aynı şekilde tüm geçişlerin, kendi renkleri kadar çoğaltılması

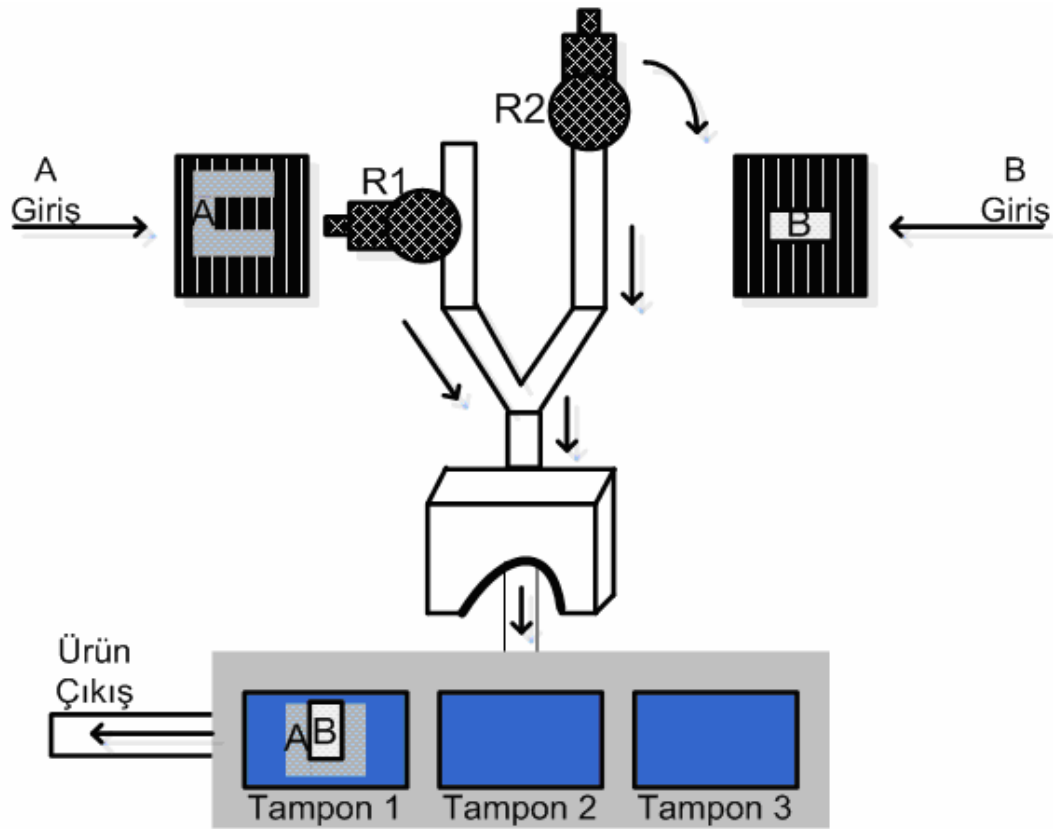
- Son olarak da ateşleme ve ateşleme sonrası davranışa uygun ok ve ağırlıklarının belirlenmesi

Adımlarından sonra RPA'lar kolayca KPA özdeşlerine dönüştürülebilirler.

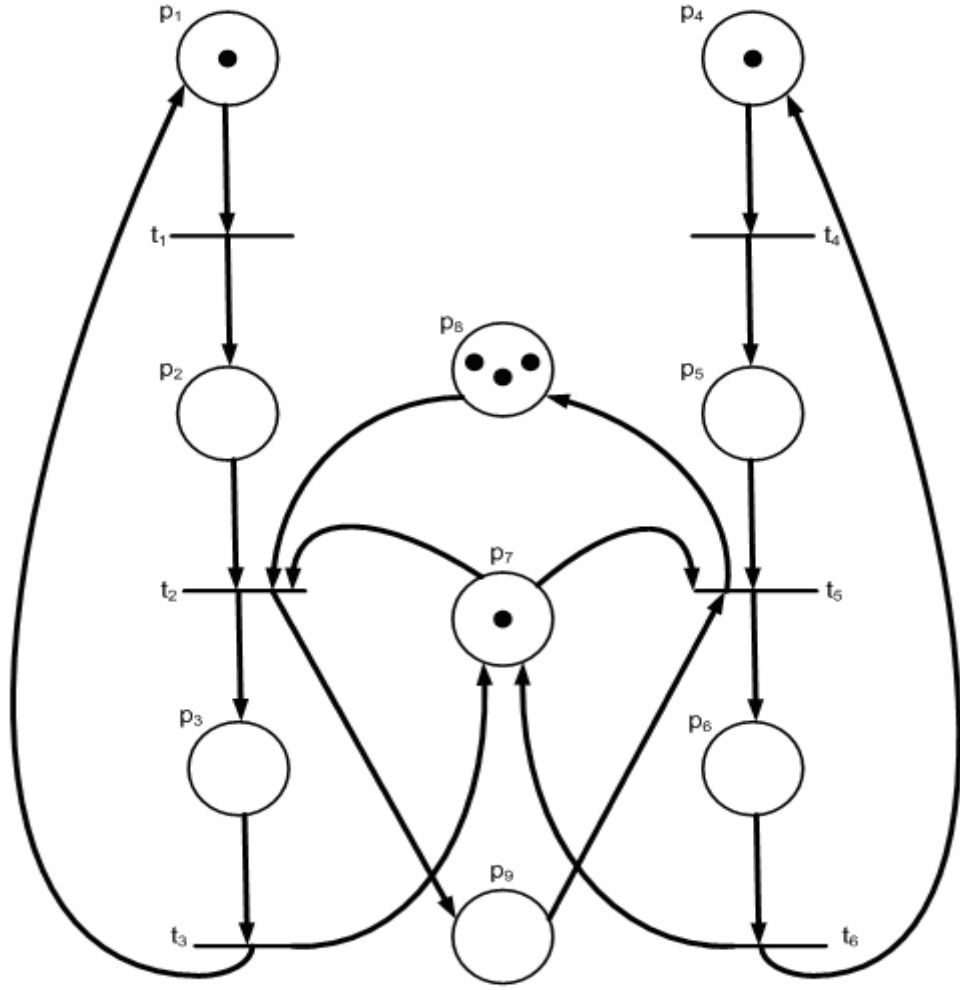
3.4 Klasik Petri Ağı'ndan Renkli Petri Ağına Dönüşüm

Petri Ağları üretim sistemlerinin modellenmesinde oldukça kullanışlıdır [14, 15, 16]. Petri Ağları'nın, gerçek dünyadaki eş zamanlı aktivitelerin, senkronlaştırmaların ve ortak kaynak kullanımının nasıl modellenebileceğini görmek için çok robotlu bir sistemi kısaca inceleyelim.

İki robotlu bir üretim sisteminin Klasik Petri Ağı modeli:



Şekil 3.7 İki robotlu bir üretim sistemi



Şekil 3.8: İki robotlu bir sistemin Klasik Petri Ağı modeli [14]

Çizelge 3.1: Şekil 3.8’deki KPA’daki yerlerin ve geçişlerin anlamları

Yerler	Temsil Ettikleri Anlamlar
$p_1(p_4)$	Robot R_1 (R_2) çalışma alanı dışında beklemede
$p_2(p_5)$	Robot R_1 (R_2) çalışma alanına girmek için beklemektedir
$p_3(p_6)$	Robot R_1 (R_2) çalışma alanında işlem yapmaktadır
p_7	İki robotun aynı anda ortak alanda olmalarını engellemek amaçlı
$p_8(p_9)$	Tampondaki boş/dolu yeri temsil eder
Geçişler	Anlamları
$t_1(t_4)$	Robot R_1 (R_2) çalışma alanına girmek için istek yapar
$t_2(t_5)$	Robot R_1 (R_2) çalışma alanına girer
$t_3(t_6)$	Robot R_1 (R_2) çalışma alanını terk eder

Çizelge 3.1’de gösterildiği gibi R_1 ve R_2 robotları sırasıyla A ve B ürünlerini almakta ve çalışma alanlarına götürerek işleme tabi tutmaktadır. Çakışmaları önlemek için çalışma alanlarında aynı anda sadece bir robot bulunacağı varsayılmıştır. Ayrıca bu örnekte alanların belirli kapasitede tampon bölgelere sahip olduğunu böylece belirli sayıda ürünü tamponlarda bekletilebildiği varsayılmıştır.

Bu modelde p_1, p_2, p_3 ve t_1, t_2, t_3 , R_1 robotunun aktivitelerini (A ürünü için), p_4, p_5, p_6 ve t_4, t_5, t_6 ise R_2 robotunun aktivitelerini (B ürünü için) modellemektedir. t_1 ve t_4 , R_1 ve R_2 robotlarının eş zamanlı yapabileceği aktivitelerdir. Bu geçişlerden biri diğerinden önce veya sonra atışlenebileceği gibi aynı anda da atışlenebilmektedir. Fakat çalışma alanına sadece bir robot girebildiği için bunu sağlamak üzere p_7, p_3, p_6 yerleri ve t_2, t_3, t_5, t_6 geçişleri düzenlenmiştir. t_2 ’nin atışlenmesi t_5 geçişini atışlenemez kılar veya tam tersi. Bu demektir ki belirli bir anda sadece bir robot çalışma alanına girebilir. Daha önceden bahsedilen tampon bölgelerinin kapasitesi 3 olsun, p_8 yeri boş ise tamponda yer yoktur ve t_2 atışlenemez. Böylece R_1 robotunun tampon kapasitesini aşması engellenir. Aynı şekilde R_2 ’de p_9 tamponunda yer yok ise çalışamaz [14].

Şekil 3.8’deki robot örneğinde olduğu gibi birçok alanda bulunabilecek KPA örneklerinde benzer görevde birçok geçiş ve yer mevcuttur. Bir sistemin modellenmesinde RPA’nın sağlayacağı en önemli avantaj ise bu noktada belirginleşmektedir.

KPA’da aynı görevdeki birçok yer ve/veya geçiş birleştirilerek tek bir yer ve/veya geçiş ile gösterilebilmektedir. Tasarımda RPA’nın kullanılmasını doğru kılacak gereksinimler kısaca şu iki başlık ile tanımlanabilir [9].

- Sistemde aynı davranışa sahip elemanların bulunması gereklidir. Renkler bu elemanlar ile ilişkilendirilirler.
- Sistemin, renklere ve geçişlere nasıl atışleneceğini matrissel bir disiplinle tanımlanabilmesi gereklidir.

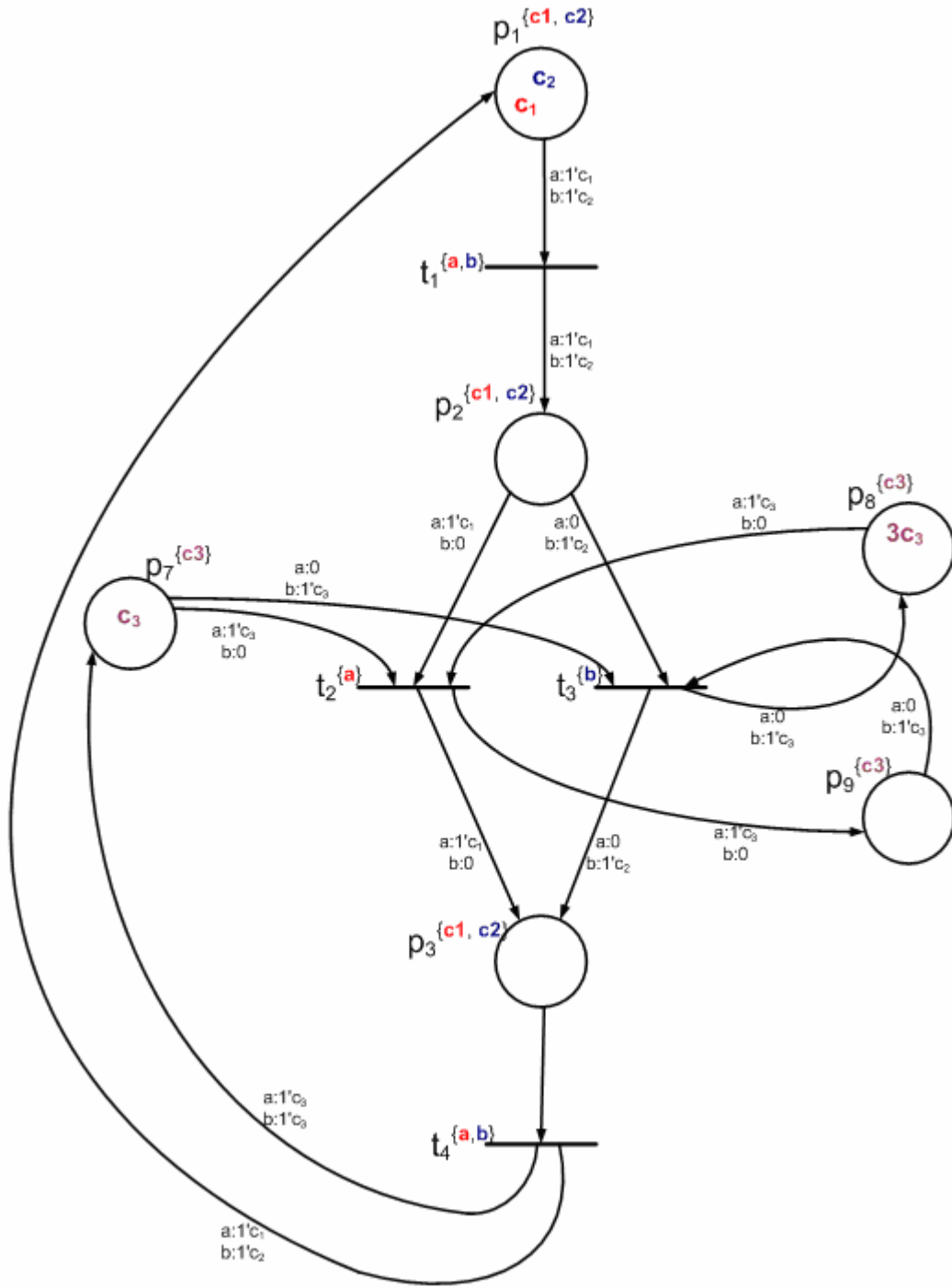
Şekil 3.8’deki KPA, benzer görevde yerlere sahiptir. Benzer yerleri ve benzer geçişleri tek bir yer ve geçiş ile göstererek bu jetonlara ve geçişlere vereceğimiz renklerle yerlerin ne türde renk bilgisi taşıyacağı belirlenir. Daha sonrasında sistem için hiçbir ayırt ediciliği olmayan yerler ve geçişlere de renk bilgisi atanır ve RPA modeli oluşturulur.

Şekil 3.8'deki örnek RPA modeli için A ve B ürünlerini veya R_1 ve R_2 robotlarını birbirinden ayırmak için farklı renkte jetonlar kullanılmıştır. Bunlar c_1 ve c_2 dir. Her iki robotun da aynı işlerini gösteren yerler birleştirilmiştir. Aynı şekilde her iki ürün veya robot davranışı için aynı olan geçişler de birleştirilmiştir.

Aynı görevdeki yerler her iki ürün/robot için aynı olduğundan taşıyacakları renk bilgileri her iki jetonu da içermelidir (c_1 ve c_2). Robotlardan biri için ağın belirli bir bölgesinde diğer robottan farklı bir davranışının söz konusu olduğu bir yer olsaydı bu yerin taşıyacağı renk bilgisi sadece ilişkili robot için verilen renk olacaktır. Benzer şekilde geçişlere de ürünler ile ilişkili renk bilgisi atanır. Ürünlere/robotlara göre davranışları farklı olan geçişler jetonlarla ilişkili renk bilgisi alırlar.

Daha sonra sistem içinde farklılık yaratmayan, bilgi taşımayan, sadece izin verilirliği belirleyen jetonlar için de bir renk bilgisi belirlenir. Bunlar aslında RPA'daki renksiz jetonlardır. Bu jetonlarla ilişkili yerler belirlenir.

Şekil 3.9'da Şekil 3.8'de ki KPA'nın RPA modeli görülmektedir.



Şekil 3.9: Şekil 3.8'de ki KPA'nın RPA modeli

Jeton renkleri dolayısıyla ilişkili yer renkleri $C(P)=\{c_1, c_2, c_3\}$, geçiş renkleri $C(T)=\{a,b\}$ dir. a geçiş rengi c_1 ve c_3 yer renklerine izin veren, b geçiş rengi c_2 ve c_3 yer renklerine izin veren geçiş renkleridir. c_3 renkli jeton her iki renkteki geçişler için izinli olduğundan renksiz bir jeton gibi düşünülebilir.

Şekil 3.9'deki RPA bileşenlerinin anlamları aşağıdaki çizelgede verilmiştir.

Çizelge 3.2: Şekil 3.9'deki RPA elemanlarının anlamları

Yerler	Temsil Ettikleri Anlamlar
p_1	Robot R_1 ve/veya R_2 çalışma alanı dışında beklemede
p_2	Robot R_1 ve/veya R_2 çalışma alanına girmek için beklemektedir
p_3	Robot R_1 ve/veya R_2 çalışma alanında işlem yapmaktadır
p_7	İki robotun aynı anda ortak alanda olmalarını engellemek amaçlı
p_8 / p_9	Tampondaki boş/dolu yeri temsil eder
Geçişler	Anlamları
t_1	Robot R_1 (R_2) çalışma alanına girmek için istek yapar
t_2	Robot R_1 çalışma alanına girer
t_3	Robot R_2 çalışma alanına girer
t_4	Robot R_1 (R_2) çalışma alanını terk eder
Jeton(Yerlerdeki)Renkleri	Taşıdıkları Bilgiler
c_1	R_1 Robotu
c_2	R_2 Robotu
c_3	Uygunlu ve kapasite belirleme amaçlı / Renksiz varsayılabilir
Geçiş Renkleri	Taşıdıkları Bilgiler
a	R_1 Robotu için ateşleme olabilecek
b	R_2 Robotu için ateşleme olabilecek

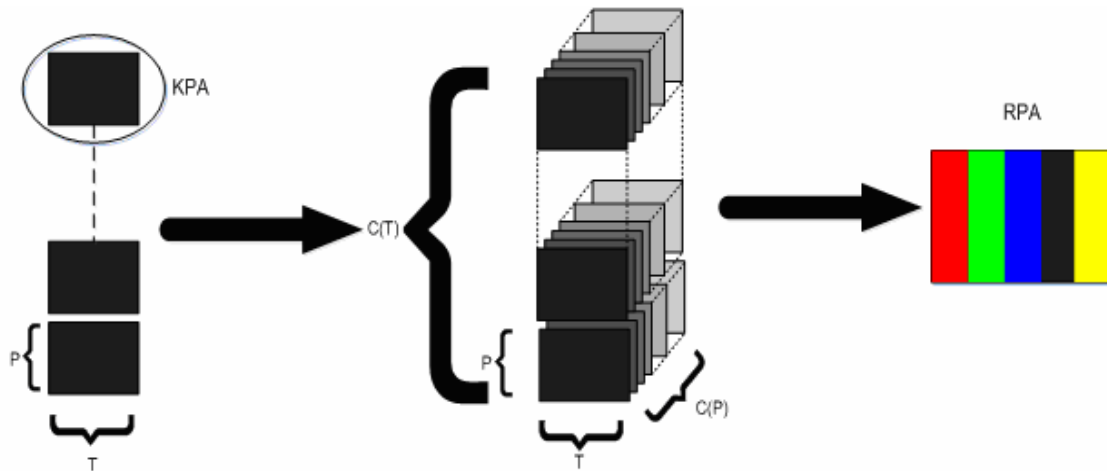
3.5 RPA-KPA Karşılaştırması

Sonlu sayıda elemanlı renk kümesi için RPA, büyük ve karmaşıklığından dolayı kullanışsız KPA modeli ile teorik olarak aynı modelleme gücüne sahiptir. Fakat bu noktada sağladığı avantaj, ağı ufaltarak daha gözlenebilir ve basit bir yapıya kavuşturarak kullanılabilir hale getirmesidir.

RPA'da durum veya işaretleme her yer ile bu yerle ilişkili renkli jetonlar kümesi ile gösterilir. KPA'da olduğu gibi durum değişikliği geçişlerin ateşlenmesi ile gerçekleşir. Renkli jetonların, geçişin ateşlenmesinden sonra geçişin giriş yerinden alınarak çıkış yerine ilave edilmesini göstermek için fazladan tanımlamalara gerek vardır. Bu tanımlamalar giriş ve çıkış matrislerinde geçiş rengi jeton rengi çiftlemesine göre ayrı ayrı ok ağırlıklarının tanımlanmasıyla mümkündür. Böylece her ok geçiş rengine göre bağlı olduğu yerden ilgili renkteki jeton sayısını gösterir hale gelir.

RPA-KPA arasındaki ilişki assembly dili ile günümüzde kullanılan gelişmiş programlama dilleri arasındaki ilişkiye benzetilebilir. Sonlu renk sınıflarına sahip Renkli Petri Ağları KP ağlarından daha iyi bir modelleme gücüne sahiptir.

RPA'dan, RPA'nın eşdeğeri olan KPA modeli uygun bir algoritma ile her zaman inşa edilebilir. Fakat RPAlar hem yoğunlukları ve okunabilirlikleri açısından, hem de analiz evresinde rahatlıkla parametreleştirilebilmelerinden dolayı daha kullanışlıdır.



Şekil 3.10: RPA-KPA ilişkisi

Şekil 3.10’te KPA ile RPA arasındaki ilişki verilmiştir. Şekilden de görülebileceği gibi birbirine benzer özelliklere sahip birçok Klasik Petri Ağı farklı renk bilgileri ile birlikte adeta sıkıştırılarak bir Renkli Petri Ağı ile ifade edilebilmektedir.

3.6 Renkli Petri Ağı’nda Erişilebilirlik Ağacı

Şekil 3.9’daki RPA modeli için sistemin davranışını gösteren giriş ve çıkış matrisleri ve başlangıç işaretlemesini şöyle yazılır.

Giriş Matrisleri:

$$N(p,t)(a,c_1)=\begin{bmatrix} 1000 \\ 0100 \\ 0001 \\ 0000 \\ 0000 \\ 0000 \end{bmatrix} \quad N(p,t)(a,c_2)=\begin{bmatrix} 0000 \\ 0000 \\ 0000 \\ 0000 \\ 0000 \\ 0000 \end{bmatrix} \quad N(p,t)(a,c_3)=\begin{bmatrix} 0000 \\ 0000 \\ 0000 \\ 0100 \\ 0100 \\ 0000 \end{bmatrix}$$

$$N(p,t)(b,c_1)=\begin{bmatrix} 0000 \\ 0000 \\ 0000 \\ 0000 \\ 0000 \\ 0000 \end{bmatrix} \quad N(p,t)(b,c_2)=\begin{bmatrix} 1000 \\ 0010 \\ 0001 \\ 0000 \\ 0000 \\ 0000 \end{bmatrix} \quad N(p,t)(b,c_3)=\begin{bmatrix} 0000 \\ 0000 \\ 0000 \\ 0010 \\ 0000 \\ 0010 \end{bmatrix}$$

Çıkış Matrisleri:

$$\begin{aligned}
 O(p,t)(a,c_1) &= \begin{bmatrix} 0001 \\ 1000 \\ 0100 \\ 0000 \\ 0000 \\ 0000 \end{bmatrix} & O(p,t)(a,c_2) &= \begin{bmatrix} 0000 \\ 0000 \\ 0000 \\ 0000 \\ 0000 \\ 0000 \end{bmatrix} & O(p,t)(a,c_3) &= \begin{bmatrix} 0000 \\ 0000 \\ 0000 \\ 0001 \\ 0000 \\ 0100 \end{bmatrix} \\
 O(p,t)(b,c_1) &= \begin{bmatrix} 0000 \\ 0000 \\ 0000 \\ 0000 \\ 0000 \\ 0000 \end{bmatrix} & O(p,t)(b,c_2) &= \begin{bmatrix} 0001 \\ 1000 \\ 0010 \\ 0000 \\ 0000 \\ 0000 \end{bmatrix} & O(p,t)(b,c_3) &= \begin{bmatrix} 0000 \\ 0000 \\ 0000 \\ 0001 \\ 0010 \\ 0000 \end{bmatrix}
 \end{aligned}$$

Durum Matrisleri:

$$\begin{aligned}
 A(p,t)(a,c_1) &= \begin{bmatrix} -1 & 0 & 0 & 1 \\ 1 & -1 & 0 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} & A(p,t)(a,c_2) &= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} & A(p,t)(a,c_3) &= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 \\ 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \\
 A(p,t)(b,c_1) &= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} & A(p,t)(b,c_2) &= \begin{bmatrix} -1 & 0 & 0 & 1 \\ 1 & 0 & -1 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} & A(p,t)(b,c_3) &= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -1 & 0 \end{bmatrix}
 \end{aligned}$$

Bir RPA'da ($|C(T)| \times |C(P)|$) tane $|P| \times |T|$ boyutunda $A(p,t)(\zeta,c)$, $\forall \zeta \in C(T)$, $\forall c \in C(P)$ matrisi vardır. Renkli Petri Ağı'nda bir geçişin bir geçiş renginde ateşlenmesi sonrası yeni işaretleme

$$M' = M + \begin{bmatrix} (A(p,t)(\varsigma, c_1) \ U(\varsigma))^T \\ \dots \\ \dots \\ (A(p,t)(\varsigma, c_{|C(P)|}) \ U(\varsigma))^T \end{bmatrix}^T$$

Başlangıç İşaretmesi:

$M_0(p_1, c_1)=1$, $M_0(p_1, c_2)=1$, $M_0(p_1, c_3)=0$, $M_0(p_2, c_1)=0$, $M_0(p_2, c_2)=0$, $M_0(p_2, c_3)=0$,
 $M_0(p_3, c_1)=0$, $M_0(p_3, c_2)=0$, $M_0(p_3, c_3)=0$, $M_0(p_7, c_1)=0$, $M_0(p_7, c_2)=0$, $M_0(p_7, c_3)=1$,
 $M_0(p_8, c_1)=0$, $M_0(p_8, c_2)=0$, $M_0(p_8, c_3)=3$, $M_0(p_9, c_1)=0$, $M_0(p_9, c_2)=0$, $M_0(p_9, c_3)=0$,
 $M_0(p_1, c)=[110]$, $M_0(p_2, c)=[000]$, $M_0(p_3, c)=[000]$, $M_0(p_7, c)=[001]$, $M_0(p_8, c)=[003]$,
 $M_0(p_9, c)=[000]$

$$M_0=M_0=\begin{bmatrix} 110 \\ 000 \\ 000 \\ 001 \\ 003 \\ 000 \end{bmatrix} \text{ dir.}$$

Örneğin t_1 geçişin a geçiş rengi için ateşlenmesi sonrası yeni işaretleme bulunsun,

$$U(\varsigma)=U(a)=\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$A(p,t)(a, c_1) \ U(a)=\begin{bmatrix} -1 & 0 & 0 & 1 \\ 1 & -1 & 0 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} -1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$A(p,t)(a,c_2) \ U(a) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$A(p,t)(a,c_3) \ U(a) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 \\ 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$M' = M + \begin{bmatrix} (A(p,t)(\varsigma, c_1) \ U(\varsigma))^T \\ \dots \\ \dots \\ (A(p,t)(\varsigma, c_{|C(P)|}) \ U(\varsigma))^T \end{bmatrix}^T$$

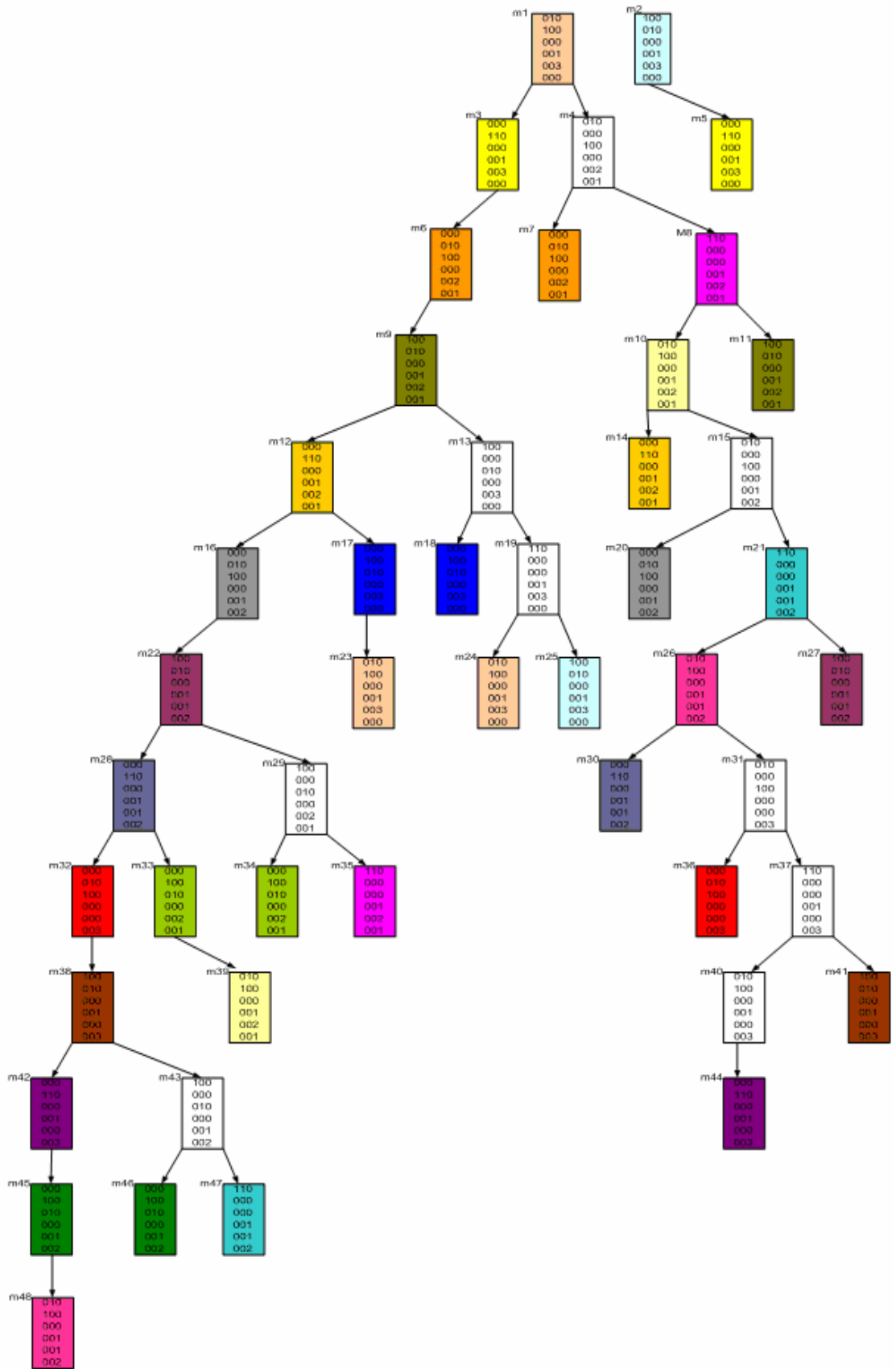
$$\begin{bmatrix} (A(p,t)(\varsigma, c_1) \ U(\varsigma))^T \\ \dots \\ \dots \\ (A(p,t)(\varsigma, c_{|C(P)|}) \ U(\varsigma))^T \end{bmatrix}^T = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T = \begin{bmatrix} -1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$M' = \begin{bmatrix} 110 \\ 000 \\ 000 \\ 001 \\ 003 \\ 000 \end{bmatrix} + \begin{bmatrix} -1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 010 \\ 100 \\ 000 \\ 001 \\ 003 \\ 000 \end{bmatrix}$$

(sarı renkte) ve M_3 ile aynı ateşlemeleri mümkün kılacağından kendinden sonra gelen durumlar M_3 ile aynı olur, bu yüzden M_5 işaretlemesinde oluşabilecek ateşlemelerin hesaplaması durdurulmuştur. Bu Renkli Petri Ağı canlıdır, çünkü her bir düğüm ya kapsanmakta ya da ateşlemelere izin vermektedir.

Şekil 3.12'daki durumlar arası bağlantı grafi $M_0 = \begin{bmatrix} 110 \\ 000 \\ 000 \\ 001 \\ 003 \\ 000 \end{bmatrix}$ başlangıç durumundan elde

edilen erişilebilen durumları göstermektedir.



Şekil 3.12: Birbirlerini kapsayan erişilebilir durumlar

RPA’da erişilebilir durumlar kümesini ve bu küme elemanları arasındaki ilişkiyi gösteren ateşlemelerin elde edilmesi için bir algoritma geliştirilmiştir:

Algoritma 1 : Geliştirilen RPA’nın Erişilebilirlik Ağacı Algoritması

$\varsigma \in C(T)$
 $c \in C(P)$
 $R(M_0) = \{M_0\}$
 $M_i = R(M_0)$
 döngü: $i=0$ 'dan $|R(M_0)|$ 'a kadar
 döngü: $j=0$ 'dan $|T|$ 'ye kadar
 döngü: $k=0$ 'dan $|C(T)|$ 'ye kadar
 $m=0$
 döngü: $l=0$ 'dan $|P|$ 'ye kadar
 döngü: $n=0$ 'dan $|C(P)|$ 'ye kadar
 eger $\left(M(i) \left(p(l), c(n) \right) \geq N \left(p(l), t(j) \right) \left(\varsigma(k), c(n) \right) \right)$ ise
 $m++$
 döngü son
 döngü son
 eger $m = |P| \times |C(P)|$ ise
 $\left(M(i), t(j)(k) \right) = 1$ ve $\left(M(i), t(j)(k) \right) \in IT$
 değil ise
 $\left(M(i), t(j)(k) \right) = 0$ ve $\left(M_i, t(j)(k) \right) \notin IT$
 döngü son
 döngü son
 döngü: $\forall t \in IT(M)$ için
 $M(i)[t > M]$
 eger $M \in R(M_0)$ ise
 $R(M_0) = R(M_0)$
 değil ise
 $R(M_0) = R(M_0) \cup \{M\}$
 döngü son
 döngü son

$R(M_0) = \{\forall M \mid \sigma : \text{atesleme dizisi ve } M_0[\sigma > M]\}$, erişilebilir işaretlemeler kümesi.

$IT = \{\forall (M, t_{(\zeta)}) \mid M \in R(M_0), \forall \zeta \in C(T) \text{ ve } \forall t \text{ için } M[t > M']\}$, izinli geçişler kümesi, erişilebilir işaretleme ile bu işaretlemenin izinli geçişlerini renk bilgisiyle birlikte çiftler halinde gösteren küme.

$IT(M) = \{\forall t_{(\zeta)} \mid \forall \zeta \in C(T) \text{ ve } M[t_{(\zeta)} > M']\}$, M işaretlemesinde izinli geçişler kümesi, Renkli Petri Ağı M işaretlemesinde iken ateşlenebilecek tüm t geçişlerini renkleriyle gösteren küme.

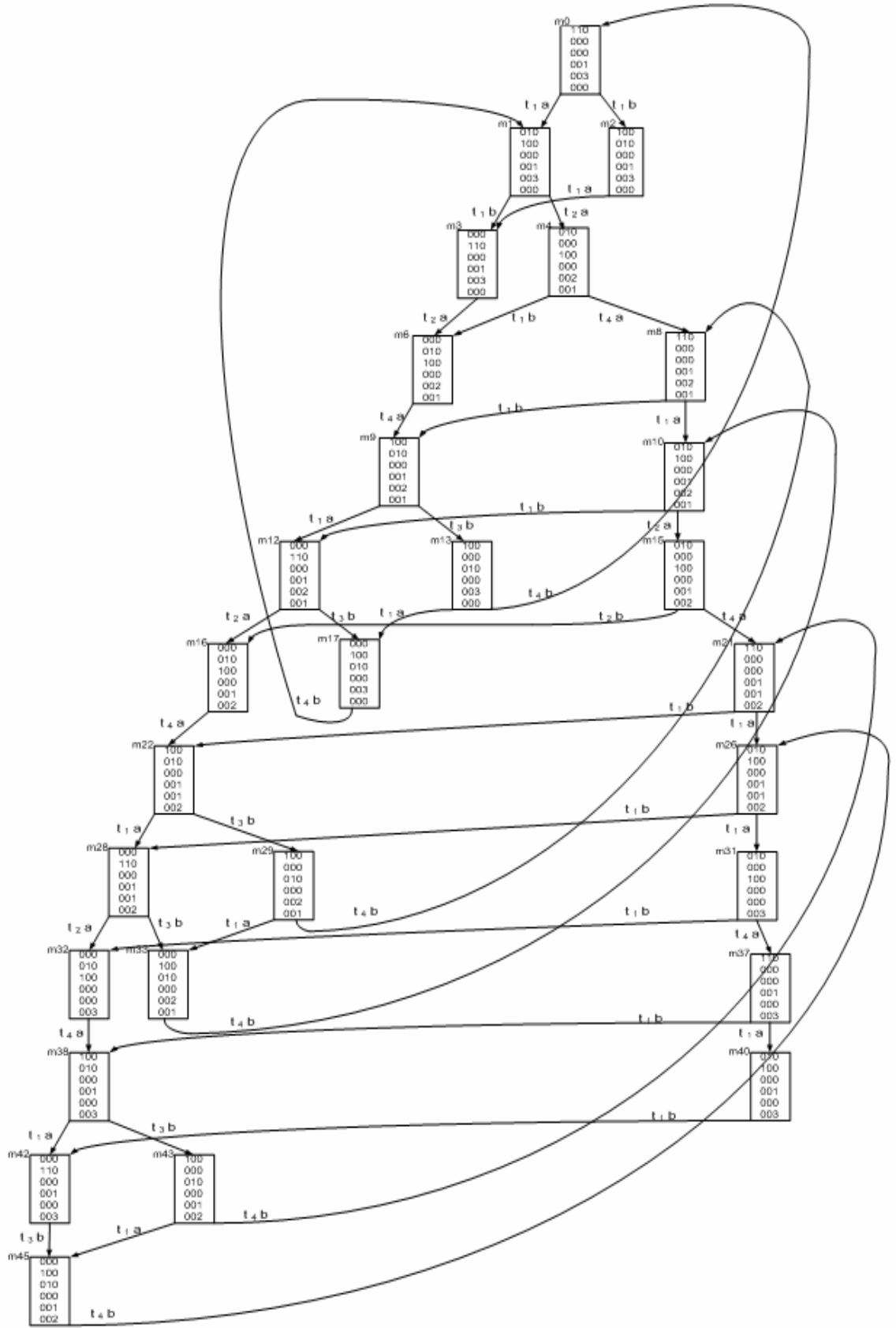
Algoritmanın açıklaması şu şekildedir.

1. Başlangıç işaretlemesi M_0 işlenmek üzere “işlenen” olarak etiketlenir.
2. “İşlenen” işaretlemesinde mümkün kılınan ateşlemeleri bul.
3. Mümkün kılınmış geçiş veya geçişler varsa 4’e geç, yoksa 6. adıma geç
4. “İşlenen” işaretlemesindeyken tüm mümkün kılınmış geçişleri bul.
5. Mümkün kılınmış işaretlemeleri birbirlerinden bağımsız olarak ateşle ve ateşlemeler sonrası oluşan yeni işaretlemeler $R(M_0)$ kümesinde yoksa sırasıyla ekle, varsa o işaretlemeye sebep olan ateşleme ve “İşlenen” i var olan bu işaretleme ile ilişkilendir.
6. $R(M_0)$ kümesinde bir sonraki işlenmemiş işaretleme varsa 7. adıma geç, yoksa 8. adıma geç.
7. Bir sonraki işaretlemeyi “işlenen” olarak etiketle ve 2. adıma dön.
8. Ağaç elde edilmiştir.

Algoritma 1, Şekil 3.7’deki RPA’ya uygulanmış ve başlangıç işaretlemesi

$$M_0 = \begin{bmatrix} 110 \\ 000 \\ 000 \\ 001 \\ 003 \\ 000 \end{bmatrix}$$

olması durumunda erişilebilecek durumları ve aralarındaki bağlantıyı sağlayan geçiş ateşlemelerini gösterecek biçimde Şekil 3.13’de elde edilmiştir.



Şekil 3.13: Şekil 3.9'deki RPA'nın erişilebilirlik ağacı

3.7 Renkli Petri Ağı'nda P Değişmezleri

Renkli Petri Ağları'nda P değişmezleri kullanılarak jetonların renklerine göre bazı yerlerdeki ağırlıklı toplamalarının farklı işaretlemeler altında aynı olduğu gözlenebilir.

Petri Ağı'nın durum denklemi

$$M_k(c) = M_{k-1}(c) + A(\zeta, c) U_k(\zeta) \text{ dir}$$

Burada $A(p, t)(\zeta, c)$ matrisi

$$A(p, t)(\zeta, c) = O(p, t)(\zeta, c) - N(p, t)(\zeta, c) \text{ dir}$$

Ateşleme dizisi σ , $M_x \rightarrow M_y$, yani M_x 'i M_y 'ye götüren sıralı ateşlemeleri göstermektedir ve aşağıdaki denklemle ifade edilebilir. (ζ, c) , $c \in C(P)$ ve $\zeta \in C(T)$, çifti için

$$M_{x+1}(c) = M_x(c) + A(\zeta, c) U_1(\zeta)$$

$$M_{x+2}(c) = M_{x+1}(c) + A(\zeta, c) U_2(\zeta)$$

$$M_{x+2}(c) = M_x(c) + A(\zeta, c) U_1(\zeta) + A(\zeta, c) U_2(\zeta)$$

$$M_{x+2}(c) = M_x(c) + A(\zeta, c) (U_1(\zeta) + U_2(\zeta))$$

...

...

...

$$M_y(c) = M_x(c) + A(\zeta, c) \sigma$$

$$\sigma = (U_1(\zeta) + U_2(\zeta) + \dots + U_{y-x}(\zeta))$$

σ bize ateşlenecek geçişlerin toplamını gösteren fakat ateşlenme sırasının kaybolduğu bir matristir. σ , bazı U ateşleme matrislerinin toplamından oluşur.

Bu fonksiyonun her iki tarafını $Z(\zeta, c)$ matrisi ile çarparsak sonuç değişmez.

$$Z(\zeta, c) M_y(c) = Z(\zeta, c) M_x(c) + Z(\zeta, c) A(\zeta, c) \sigma$$

$Z(\zeta, c)$ öyle seçilir ki $Z(\zeta, c) A(\zeta, c) = 0$ dır.

Dolayısıyla

$$Z(\zeta, c) M_y(c) = Z(\zeta, c) M_x(c)$$

Olur ki $Z(\zeta, c)$, Petri Ağı'ndaki belirli yerlerdeki jetonların, σ ateşleme dizisinden bağımsız olarak belirli sabitlerle çarpımlarının toplamının her zaman sabit kaldığını gösterir. Bu ifadenin diğer bir yorumu, M_0 başlangıç koşulu altında $Z(\zeta, c).M_0(c)=k$, $k \in \mathbb{N}$ sonucu M_0 'dan elde edilen diğer tüm durumlar için de aynıdır. $Z(\zeta, c)$, Renkli Petri Ağları'nda her geçiş rengi- jeton rengi çifti için $(1 \times |P|)$ boyutunda, en az iki elemanı sıfırdan farklı bir matristir ve $\forall \zeta \in C(T)$ ve $\forall c \in C(P)$ için $Z(\zeta, c) A(p, t)(\zeta, c) = 0$ özelliğini sağlamalıdır. Yüksek seviyeli Petri Ağları için değişmezler literatürde incelenmiştir[17].

P değişmezlerinin uygulama alanları;

- Petri Ağları'nın otomat karşılığını bulurken
- Petri Ağı'nın doğruluğunun tespiti için
- Ağın bir yerlerinde jeton azalması dolayısıyla canlılığın ortadan kalkması (ölü kilitleme) durumunun araştırılması
- Ağda jeton sayısının bir yerlerde artarak sonsuza gitmesi, yani sınırlılık özelliğini etkileyecek faktörlerin araştırılması

gibi uygulamalar için P değişmezleri araştırılır.

RPA'larda P değişmezlerinin hesaplanması her geçiş rengi için jeton renklerinin ayrı ayrı hesaplanmasıyla bulunur. Örneğin Şekil 3.9'daki Petri Ağı için geçişlerin a renginde ateşlenmesi halinde c_1 renkli jetonlar için P değişmezleriyle c_2 renkli jetonların P değişmezleri farklı olabilir. Dolayısıyla bir RPA için $|C(T)| \times |C(P)|$ tane P değişmezi hesaplanır. Her bir geçiş-jeton renk çifti için P değişmezi olmayabileceği gibi birden fazla P değişmezi matrisi de elde edilebilir.

3.8 RPA İçin Martinez Silva Algoritması ile P Değişmezlerinin Bulunması

Klasik Petri Ağları'nda P değişmezlerinin elde edilmesi için Martinez-Silva [18,19,20] algoritması geliştirilmiştir. Martinez-Silva Algoritmasının RPA'lar için uygulaması $|C(T)| \times |C(P)|$ defa yani her geçiş-jeton çifti için algoritmanın uygulanması şeklinde olacaktır.

Aşağıda Renkli Petri Ağı'nda bir geçiş rengi-jeton rengi çifti için Martinez-Silva algoritması verilmiştir.

Algoritma 2 : Martinez Silva Algoritması

$A(p,t)(a,c_1) = O(p,t)(a,c_1) - N(p,t)(a,c_1)$
 $I_{|P| \times |P|} = |P| \times |P|$ boyutunda birim matris
 $Q_0 = [I | A(p,t)(a,c_1)], |P| \times |P| + T$ boyutunda matris
 $g, q \in N(\text{dogal sayilar}) \wedge g \neq q$ olan değişkenler
 $\lambda, \beta \geq 1$ olan katsayı
 Lg, Lq ve Lr sirasiyla g, q . ver. satirlari göstermek üzere
döngü $i = |P| + 1$ den $|P| + T|$ ye kadar
 $S = \{ \}$
döngü $g = 1$ den $g = (Q_0$ 'ın sütun sayıya- 1)' e kadar
döngü $q = g + 1$ den $q = (Q_0$ 'ın sütun sayıya)' na kadar
eger $(Q_0(g,i) \neq 0 \wedge Q_0(q,i) \neq 0 \wedge \lambda \times Q_0(g,i) + \beta \times Q_0(q,i) = 0)$ ise
 $\lambda \times Lg + \beta \times Lq$ satiri Q_1 'e eklenir
 $S = S \cup \{g, q\}$
döngü son
döngü son

 $\forall r \notin S$ için
 Lr satirlari da Q_1 'e eklenir
 $Q_0 = Q_1$
döngü son
 I matri sin sifirdan farkli satirlari P deg ismezlerini verir

Martinez-Silva algoritmasının uygulaması için örnek olarak Şekil 3.7’de Renkli Petri Ağı için giriş ve çıkış matrisleri daha önceden verilmiştir. Bu RPA örneği için a geçiş rengi ve c_1 jeton(yer) rengi için algoritmanın uygulanışını adım adım gösterilmiştir.

$$O(p,t)(a,c_1)=\begin{bmatrix} 0001 \\ 1000 \\ 0100 \\ 0000 \\ 0000 \\ 0000 \end{bmatrix}, \quad N(p,t)(a,c_1)=\begin{bmatrix} 1000 \\ 0100 \\ 0001 \\ 0000 \\ 0000 \\ 0000 \end{bmatrix} \text{ idi.}$$

$$t_1 \quad t_2 \quad t_3 \quad t_4$$

$$A(p,t)(a,c_1)=\begin{bmatrix} 0001 \\ 1000 \\ 0100 \\ 0000 \\ 0000 \\ 0000 \end{bmatrix} - \begin{bmatrix} 1000 \\ 0100 \\ 0001 \\ 0000 \\ 0000 \\ 0000 \end{bmatrix} = \begin{bmatrix} -1 & 0 & 0 & 1 \\ 1 & -1 & 0 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{matrix} p_1 \\ p_2 \\ p_3 \\ p_7 \\ p_8 \\ p_9 \end{matrix}$$

$$e_1 \quad e_2 \quad e_3 \quad e_4 \quad e_5 \quad e_6 \quad t_1 \quad t_2 \quad t_3 \quad t_4$$

$$Q_0 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{matrix} p_1 \\ p_2 \\ p_3 \\ p_7 \\ p_8 \\ p_9 \end{matrix}$$



$$Q_0 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{matrix} p_1 \\ p_2 \\ p_3 \\ p_7 \\ p_8 \\ p_9 \end{matrix}$$

$$Q_0 = \begin{array}{c} e_1 \ e_2 \ e_3 \ e_4 \ e_5 \ e_6 \ t_1 \ t_2 \ t_3 \ t_4 \\ \left[\begin{array}{cccccccc} 1 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{array} \right] \begin{array}{l} p_{1+2} \\ p_3 \\ p_7 \\ p_8 \\ p_9 \end{array} \end{array}$$

$$Q_0 = \begin{array}{c} e_1 \ e_2 \ e_3 \ e_4 \ e_5 \ e_6 \ t_1 \ t_2 \ t_3 \ t_4 \\ \left[\begin{array}{cccccccc} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{array} \right] \begin{array}{l} p_{1+2+3} \\ p_7 \\ p_8 \\ p_9 \end{array} \end{array}$$

↙
Z= P Değişmezleri

$Z(a, c_1) \ A(p, t)(a, c_1) = 0$ olmalı

$$[1 \ 1 \ 1 \ 0 \ 0 \ 0] \begin{bmatrix} -1 & 0 & 0 & 1 \\ 1 & -1 & 0 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} = 0$$

$$M_0(p, c_1) = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \text{ için } Z(a, c_1) \ M_0(p, c_1) = [1 \ 1 \ 1 \ 0 \ 0 \ 0] \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = 1$$

Yani P_1, P_2 ve P_3 yerlerindeki jeton sayılarının $Z(a, c_1)$ matrisinin ilgili elemanlarıyla çarpımları toplamı, a geçiş rengi ve c_1 jeton rengi için verilen başlangıç durumuna göre hep sabit ve 1'e eşittir.

Şekil 3.9'deki RPA için tüm geçiş rengi – jeton rengi için Algoritma 2 kullanılarak P değişmezleri bulunmuştur.

$$Z(a, c_1) = [1 \ 1 \ 1 \ 0 \ 0 \ 0]$$

$$Z(a, c_3) = [0 \ 0 \ 0 \ 0 \ 1 \ 1]$$

$$Z(b, c_2) = [1 \ 1 \ 1 \ 0 \ 0 \ 0]$$

$$Z(b, c_3) = [0 \ 0 \ 0 \ 0 \ 1 \ 1]$$

3.9 RPA İçin Satır Eliminasyonu Yöntemi Algoritması ile P Değişmezlerinin Bulunması

Bu algoritma bir matris üzerinde değeri sıfırdan farklı ve mümkün olduğunca en küçük satırın en küçük sütunundaki elemanı pivot olarak atar. Tüm satırlar, bu pivot ile aynı sütundaki elemanlarına bölünür ve pivot ile çarpılarak pivotun satırı ile bu pivotun bulunduğu sütundaki elemanları sıfırlamak üzere toplanır veya farkı alınır.

Bunu yaparken $A(p, t)(\zeta, c)$ matrisin soluna $|P| \times |P|$ boyutunda bir birim matris eklenir ve aynı işlemler bu matrise de uygulanırsa sonuçta P Değişmezleri elde edilir. $A(p, t)(\zeta, c)$ matrisinin olduğu kısımda tüm elemanları sıfır olan satırların bulunduğu eski birim matris kısmı P değişmezlerini verir.

Bir matris üzerinde Satır Eliminasyon Yöntemini gösterilsin. A , $n \times m$ boyutunda matris olsun ve $n=4$, $m=5$ olan A matrisi aşağıdadır.

$$A = \begin{bmatrix} 0 & -3 & -6 & 4 & 9 \\ 1 & 4 & 5 & -9 & -7 \\ -1 & -2 & -1 & 3 & 1 \\ -2 & -3 & 0 & 3 & -1 \end{bmatrix}$$

$$A = \begin{bmatrix} 0 & -3 & -6 & 4 & 9 \\ 1 & 4 & 5 & -9 & -7 \\ -1 & -2 & -1 & 3 & 1 \\ -2 & -3 & 0 & 3 & -1 \end{bmatrix}$$

Pivot sütunu

Pivot eleman

$S_1 \rightarrow S_2$ ile yer değiştirilir

$$A = \begin{bmatrix} 1 & 4 & 5 & -9 & -7 \\ 0 & -3 & -6 & 4 & 9 \\ -1 & -2 & -1 & 3 & 1 \\ -2 & -3 & 0 & 3 & -1 \end{bmatrix}$$

$S_3 \rightarrow S_1 + S_3$, $S_4 \rightarrow 2S_1 + S_4$ yaz.

$$A = \begin{bmatrix} 1 & 4 & 5 & -9 & -7 \\ 0 & -3 & -6 & 4 & 9 \\ 0 & 2 & 4 & -6 & -6 \\ 0 & 5 & 10 & -15 & -15 \end{bmatrix}$$

Olası pivot adayları, S_2 Pivot olarak seçilsin

$S_3 \rightarrow S_2 - S_3 (-3/2)$, $S_4 \rightarrow S_2 - S_4 (-3/5)$ yaz

$$A = \begin{bmatrix} 1 & 4 & 5 & -9 & -7 \\ 0 & -3 & -6 & 4 & 9 \\ 0 & 0 & 0 & -5 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \text{ Satır eliminasyonu uygulanmış matris}$$

A matrisinin soluna $n \times n$ boyutunda birim matris konulsaydı,

$$[I:A]=\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & -3 & -6 & 4 & 9 \\ 0 & 1 & 0 & 0 & 1 & 4 & 5 & -9 & -7 \\ 0 & 0 & 1 & 0 & -1 & -2 & -1 & 3 & 1 \\ 0 & 0 & 0 & 1 & -2 & -3 & 0 & 3 & -1 \end{bmatrix}$$

$$[I:A]=\begin{bmatrix} 0 & 1 & 0 & 0 & 1 & 4 & 5 & -9 & -7 \\ 1 & 0 & 0 & 0 & 0 & -3 & -6 & 4 & 9 \\ 1 & 1.5 & 1.5 & 0 & 0 & 0 & 0 & -5 & 0 \\ 1 & 1.2 & 0 & 0.6 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$



P değişmezleri kısmı

Bu kısım normalize edilirse P değişmezleri $Z(\zeta,c) = [5 \ 6 \ 0 \ 3]$, $\zeta \in C(T)$, $c \in C(P)$ bulunur. Bu P değişmezi örneği RPA'nın sadece bir geçiş rengi-jeton rengi çifti içindir. RPA'daki bütün $A(p,t)(\zeta,c)$ matrisleri için bu işlem yapılarak ağdaki tüm P değişmezleri renklerine göre bulunur.

4.CPN-RPA PROGRAMININ ÖZELLİKLERİ

Bu bölümde, yapılan çalışma kapsamında geliştirilen Renkli Petri Ağları için erişilebilirlik ve P değişmezleri analizi yapan CPN-RPA programının nasıl geliştirildiği, çalışma şekli ve analiz sonuçları incelenecektir.

4.1 Programın Girdileri

CPN-RPA Programı kendi exe'si olan bir programdır. Programa analizi yapılacak RPA'ya ilişkin ağın karakteristiğini veren matrisler girilir. Bu matrisler girdi matrisleri ile çıktı matrisleridir. 3.1 bölümünde bahsedilen bu matrislerin sayısı

$|C(T)| \times |C(P)|$ kadardır ve RPA'nın renklere bağlı davranışını verir. Program bu matrisleri “.txt” uzantılı metin dosyasına girilmiş olarak alır girilmediyse girilmesi için kullanıcıyı zorlar. Matrisler metin dosyalarına alt alta yazılmalıdır. Tüm bu anlatılanlar girdi ve çıktı matrisleri için geçerlidir.

Programın bir diğer girdisi ise M_0 başlangıç işaretlemesidir. M_0 matrisi ayrı bir “.txt” dosyasından programa verilir. Aynı şekilde M_0 matrisi de kullanıcı önceden girmediyse girmesi için zorlanır.

Bu matrislerin girilmesinden sonra program çalıştırıldığında kullanıcıdan geçiş sayısı, yer sayısı, geçiş rengi sayısı ve jeton rengi sayısını almaktadır.

Geliştirilen program 3 ayrı parçadan oluşur, bunlardan ilki erişilebilirlik ağacını, ikincisi Martine-Silva algoritması ile P değişmezlerini sonuncusu ise satır eliminasyon yöntemiyle P değişmezlerini bulur.

Yukarıda bahsedilen girdilerden sonra program bölüm 3.6'da bahsedilen erişilebilirlik ağacını elde eden algoritmaya göre mümkün kılınan geçişleri yani ateşlenebilecek olan geçişleri bir “.txt” metin dosyasına atar daha sonra yine aynı algoritmaya göre erişilebilir durumları sırasıyla numaralandırarak başka bir “.txt” metin dosyasına atar. Bahsedilen ikinci metin dosyası ilk metin dosyasındaki erişilebilir durumların arasındaki ateşlenen geçişi ve rengini verir ayrıca birbirini kapsayan işaretlemeleri gösterir.

Daha sonra 3.8’de anlatılan Martinez-Silva algoritmasına göre P değişmezlerini geçiş rengi jeton rengi çiftine göre bulur. Bulduğu P değişmezleri matrisini ayrıca bir .txt metin dosyasına atar.

P değişmezlerinin bulunmasıyla ilgili satır eliminasyon yöntemi 3.9’da anlatılmıştır. Bu yöntemde ayrıca başka bir programla çalıştırılarak P değişmezleri elde edilir. Bu programında çıktıları “.txt” metin dosyasına atılır. Tüm bu geliştirilen program parçaları için programın çalışma zamanı hesaplanır.

4.2 Programın Geliştirilme Ortamı

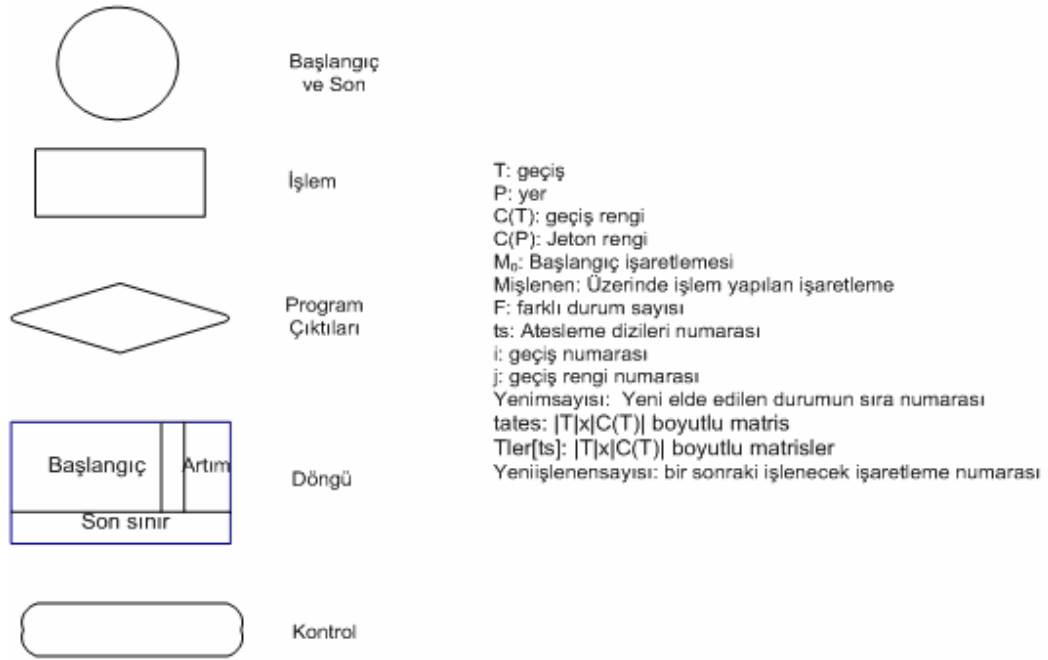
CPN-RPA programının 4.1’de anlatılan 3 parça halindeki hesaplama programları DEVCC++ Versiyon 4.9.9.2 ortamında C programlama dili ile geliştirilmiştir.

Bu 3 parça program tek bir çalıştırma ortamında birleştirilmiştir. Bu işlem Visual Basic 6.0 programı ile gerçekleştirilmiştir. Böylece yazılan 3 program da tek bir exe (çalıştırma aracı) ile çalıştırılmakta ve çıktı almak için kullanıcı ara yüzü tuşları aktif hale gelmektedir. Program, üzerinde Windows 2000, XP, Vista bulunan bilgisayarlarda ilgili program parçaları ve girdi dosyaları ile birlikte sorunsuz çalışmaktadır.

4.3 Programın Çalışma Mantığı

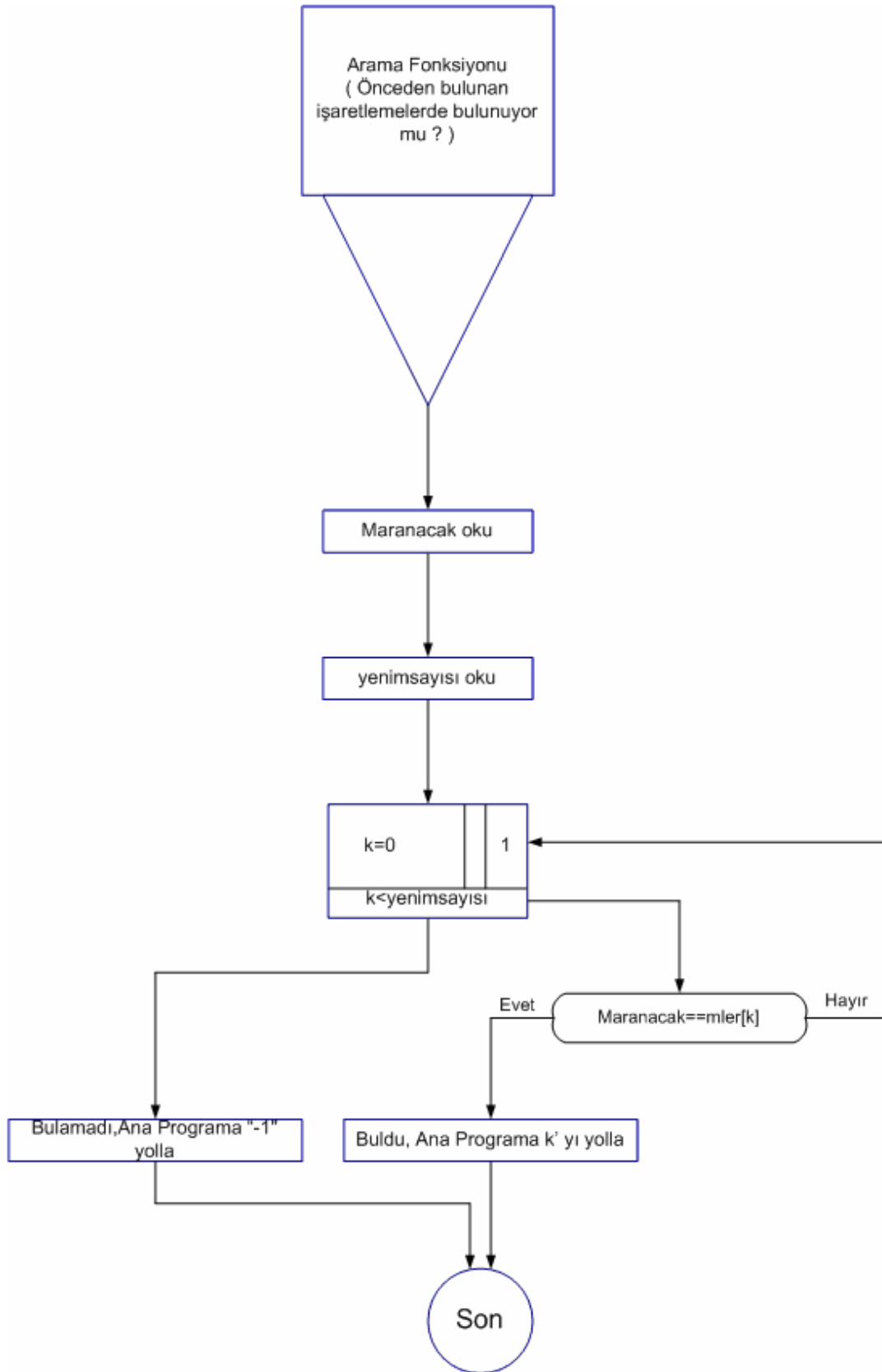
CPN-RPA programı algoritma 1 ve algoritma 2’ye göre giriş ve çıkış matrisleri ve düğüm sayıları ile düğüm renk sayılarını girdi olarak alır ve çıktı olarak da ağıın erişilebilir durumları ile bu durumların birbirleri arasındaki geçiş-geçiş rengi ilişkisini ve P değişmezlerini her jeton rengi için varsa bulur.

Geliştirilen erişilebilirlik analizi programının akış diyagramı Şekil 4.2 ve Şekil 4.3’de verilmiştir. Şekil 4.1’de, Şekil 4.2 ve Şekil 4.3’de kullanılan akış diyagramı elemanları tanıtılmıştır.

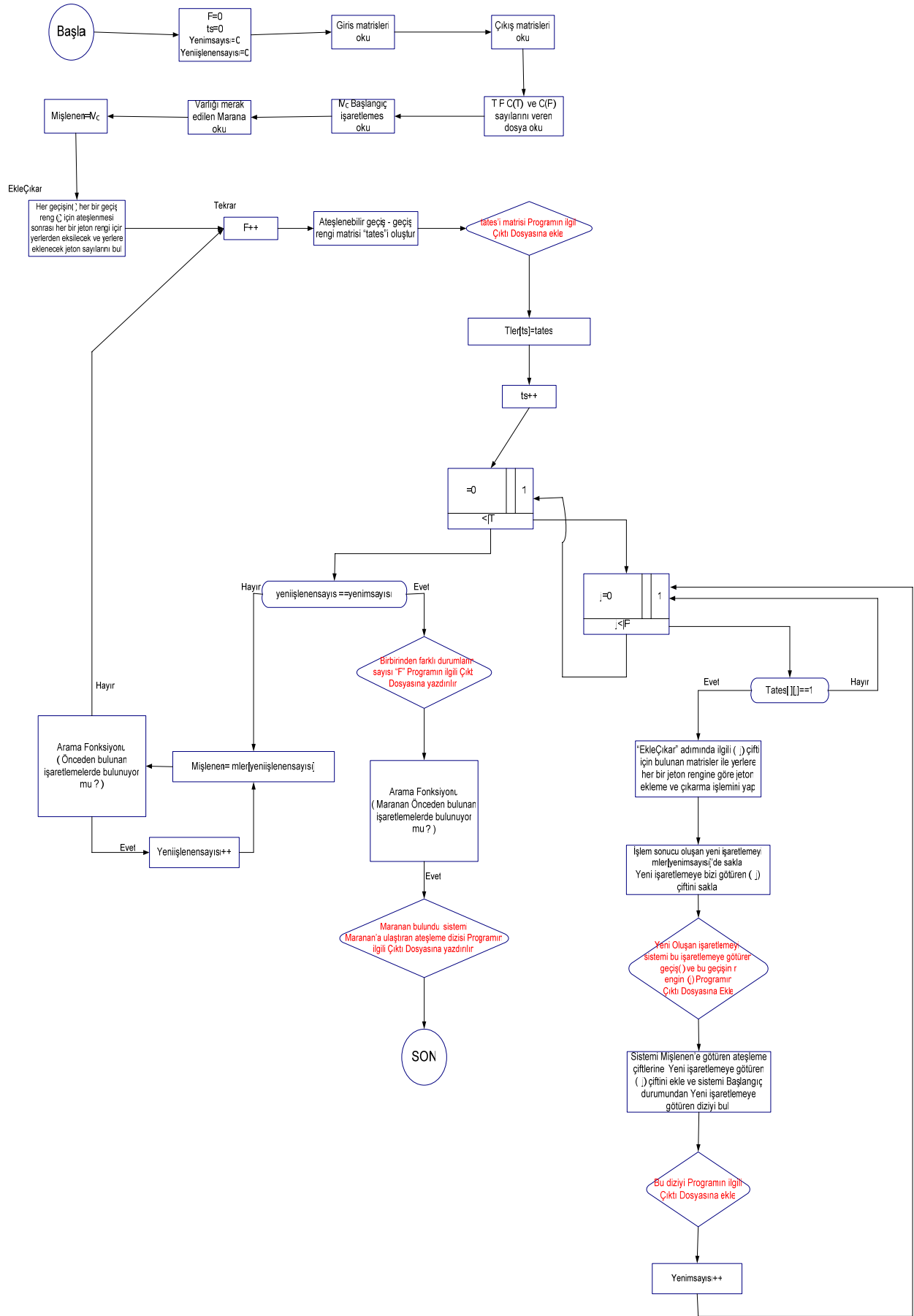


Şekil 4.1: Akış diyagramı elemanları

Şekil 4.2'deki arama fonksiyonu akış diyagramı ile erişilebilirlik ağacında hesaplanan durumlardan birbirlerini kapsayan durumların tespiti için kullanılmıştır. Şekil 4.3'te ise erişilebilirlik ağacı algoritmasının akış diyagramı gösterilmiştir. Bu akış diyagramı verilen RPA modeli için erişilebilir durumları ve aralarındaki bağlantıları vermektedir.



Şekil 4.2: Arama programı akış diyagramı



Şekil 4.3: Erişilebilirlik ağacı akış diyagramı

Geliştirilen yazılım Şekil 3.7’deki RPA modeline uygulanmış ve elde edilen sonuçlar Çizelge 4.1’de verilmiştir. Çizelgede verilen “Örnek 2” ise Şekil 3.5’deki iki robotlu üretim sisteminden iki adet olan bir çalışma alanının modellenmesinin sonuçları verilmektedir. Yani Örnek 2, Şekil 3.5’deki R1 ve R2’den 2 şer adet olan bir sistem için hesaplanmıştır. Sonuçlar Pentium 4, 2.66Ghz CPU hızıyla 512 MB RAM bulunan bilgisayar üzerinde ortalama sonuçlardır.

Çizelge 4.1: Program çalışma zamanı ve erişilen durum sayısı

RPA	Örnek Şekil 3.7	Örnek 2
Geçiş Sayısı	4	4
Geçiş Renk Sayısı	2	4
Yer Sayısı	6	6
Jeton Renk Sayısı	3	6
Hesaplama Süresi (mili saniye)	51	343
Birbirinden Farklı Durum (İşaretleme) Sayısı	28	243

Şekil 3.7’deki RPA’nın P Değişmezleri

$$Z(a,c_1) = [1 \ 1 \ 1 \ 0 \ 0 \ 0]$$

$$Z(a,c_3) = [0 \ 0 \ 0 \ 0 \ 1 \ 1]$$

$$Z(b,c_2) = [1 \ 1 \ 1 \ 0 \ 0 \ 0]$$

$$Z(b,c_3) = [0 \ 0 \ 0 \ 0 \ 1 \ 1]$$

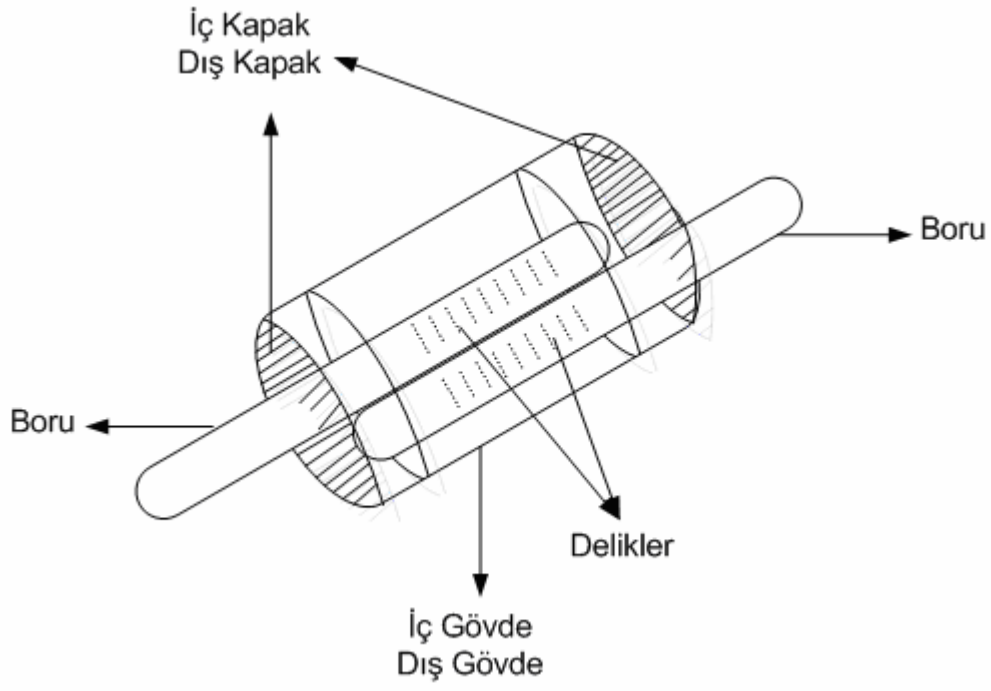
Bu P değişmezleri ortalama 1us hesaplama süresiyle hesaplanmıştır.

4.4 Örnek Çalışma: Susturucu (Egzoz) Üretim Aşamaları

Aşağıdaki Şekil 4.4'te “Susturucu (Egzoz)” parçaları verilmiştir. Bir egzozun üretilmesi için dört adet bileşene ihtiyaç vardır. Bunlar “Dış Gövde”, “İç Gövde”, “İç Boru”, “İç Kapak” ve “Dış Kapak” tır.

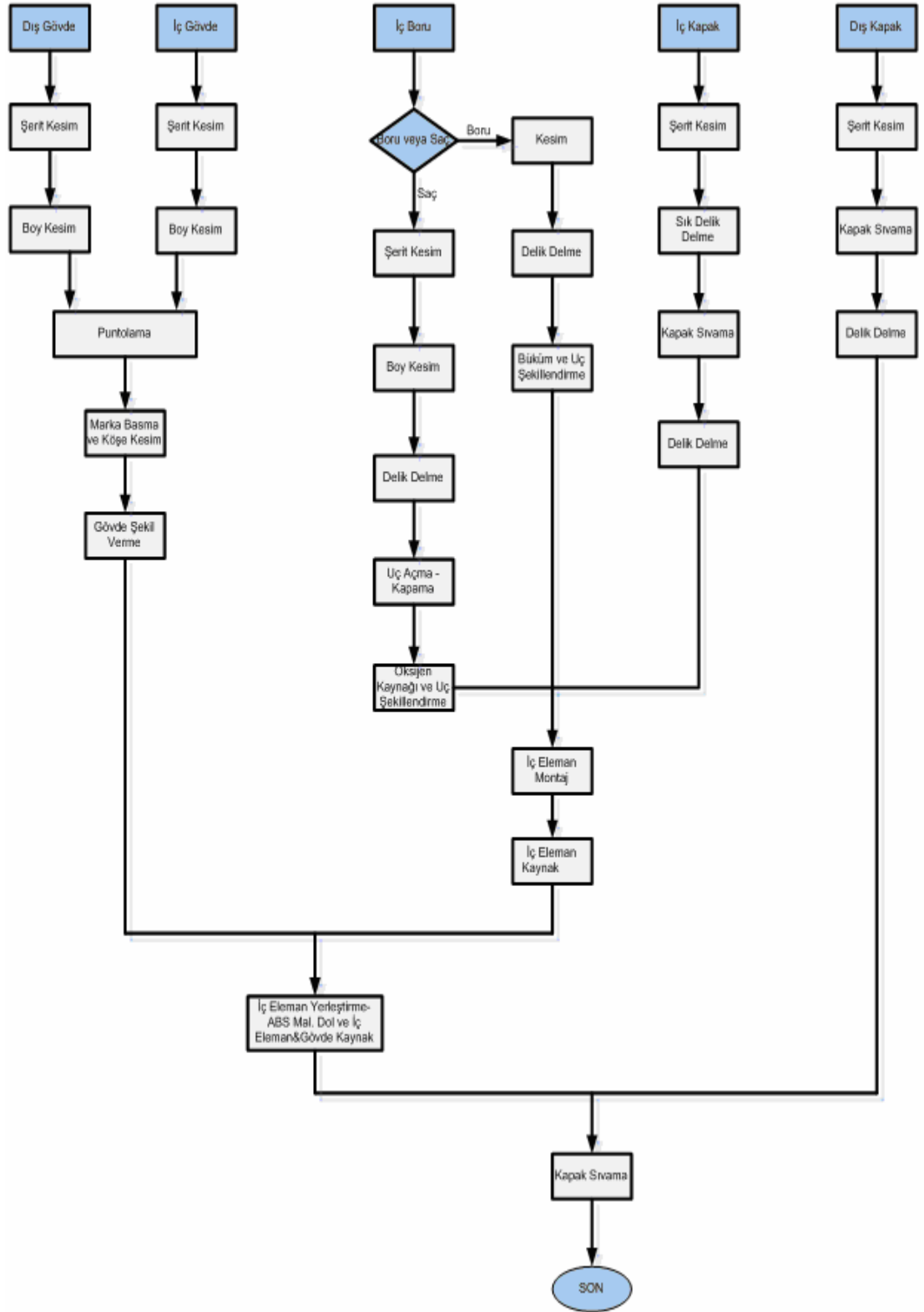
Mevcut olan ham maddeler belirli işlemlere tabi tutulurlar. Kesme, puntolama, delik delme, büküm ve uç şekillendirme, kapak sıvama, uç açma kapama ve bu işlemlerle belirli özellikleri kazanmış parçaların birbirlerine monte edilmesi gibi işlemlerin ardından bir araçta monte edilerek egzoz kullanıma hazır hale gelir. Bahsedilen bu işlemler bu dört ana parça için bazen ortak olarak uygulanır. Örneğin iç gövdeye de dış gövdeye de şerit kesme işlemi uygulanmaktadır. Bunun gibi birçok işlem ortak olarak tüm işlemlere uygulanmaktadır.

Bu üretim safhalarının insan eli değmeden bir elektronik otomasyon sistemi ile gerçekleştirilmesi söz konusudur. Farklı üretim safhalarının ortak alanlarının olduğu bu üretim sistemi bir PLC kullanılarak kontrol edilebilir ve yapısı gereği Renkli Petri Ağı ile en iyi şekilde modellenenebilir. Klasik Petri Ağı ile birçok yer ve geçiş ile modellenenecek olan bu üretim sistemi Renkli Petri Ağı modeli ile daha az yer ve geçişle birlikte renkli jeton ve geçişlerle modellenenebilir. Böylece görsel olarak daha anlaşılır, daha küçük ve programlama mantığıyla daha uyumlu bir modelleme yapılabilir. RPA'daki renkler veri tipleri olarak düşünülebilir. RPA'lar, benzerlikler barındıran birçok farklı modülün bulunduğu yazılımların modellenmesinde de kullanılabilir. Fiziksel bir sistem veya yazılımın modellenmesindeki amaç davranışın gözlenmesi veya aranan durumun sistem tarafından sağlanıp sağlanmadığı sorusuna cevap aramaktır. Ayrıca bir kontrolör tasarımı için veya modelde yapılacak sınırlamaların yansımalarının gözlenmesi içinde Renkli Petri Ağı modellemesi kullanılabilir. Şekil 4.4'te Egzoz'un belli başlı bileşenleri gösterilmiştir.



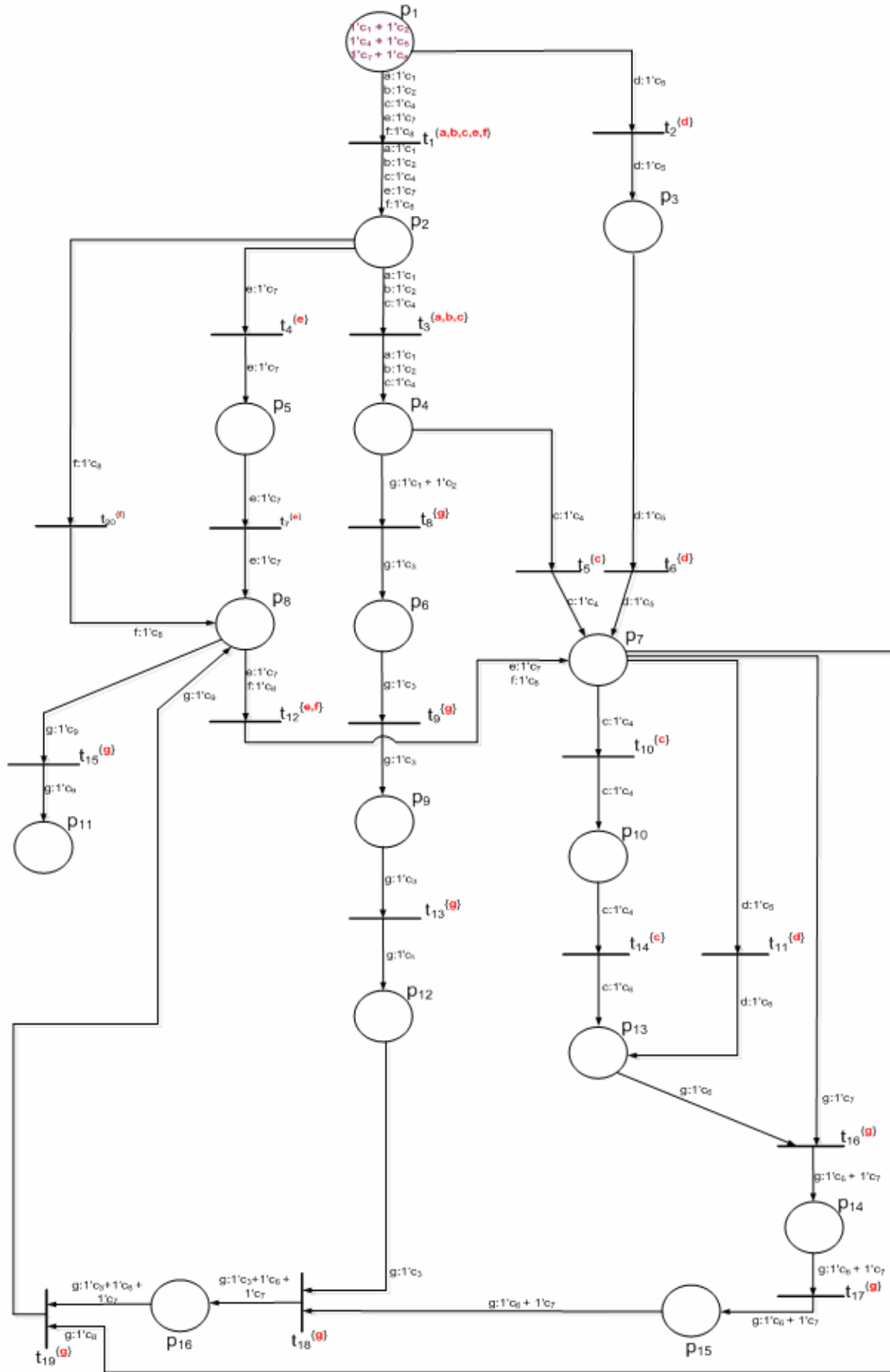
Şekil 4.4: Egzoz bileşenleri

Şekil 4.5’te “Susturucu (Egzoz) Üretim Aşamaları” verilmiştir. Bir egzoz ürününün üretilmesi için uygulanan işlemleri göstermektedir.



Şekil 4.5: Susturucu (Egzoz) üretim aşamaları

Şekil 4.6’da Şekil 4.5’te verilen aşamalara bağlı kalarak bu üretim sisteminin RPA modeli verilmiştir.



Şekil 4.6: Susturucu (Egzoz) üretim aşamaları Renkli Petri Ağı modeli

Çizelge 4.2: Şekil 4.4 Yerlerin anlamları

Yerler	Anlamları
P ₁	Ham malzemeler mevcut ve beklemede
P ₂	Şerit kesildi
P ₃	Kesim yapıldı
P ₄	Boy kesildi
P ₅	Sık delik kesildi
P ₆	Puntolama bitti
P ₇	Delik delindi
P ₈	Kapak sıvand
P ₉	Marka basma ve köşe kesim işlemi bitti
P ₁₀	Uç açma kapama bitti
P ₁₁	Ürün monte edildi, son
P ₁₂	Gövde şekil verme bitti
P ₁₃	İç boru hazır
P ₁₄	İç eleman monte edildi
P ₁₅	İç eleman kaynak yapıldı

Çizelge 4.3: Şekil 4.4 Geçişlerin anlamları

Geçişler	Anlamları
t ₁	Şerit kesim işlemi
t ₂	Kesim işlemi (iç boru kesim işlemi)
t ₃	Boy kesim işlemi
t ₄	Sık delik delme işlemi
t ₅	Delik delme işlemi
t ₆	Delik delme işlemi
t ₇	İç Kapak sıvama
t ₈	Puntolama işlemi
t ₉	Marka basma ve köşe kesim işlemi
t ₁₀	Uç açma kapama işlemi
t ₁₁	Büküm ve uç şekillendirme işlemi
t ₁₂	Delik delme işlemi
t ₁₃	Gövde şekil verme işlemi
t ₁₄	İç eleman monte edildi
t ₁₅	Kapak sıvaması işlemi
t ₁₆	İç eleman montaj işlemi
t ₁₇	İç eleman kaynak
t ₁₈	İç eleman yerleştirme - ABS mal. dol. ve iç eleman&gövde kay. İşlemi
t ₁₉	Kapak sıvaması işlemi
t ₂₀	Dış Kapak sıvama

Çizelge 4.4: Şekil 4.4 Jeton renkleri anlamı

Jeton renkleri	Anlamları
c ₁	Dış gövde
c ₂	İç gövde
c ₃	Hazır olan dış veya iç gövde
c ₄	Sac iç boru
c ₅	Boru (plastik) iç boru
c ₆	Hazır olan iç boru (boru veya sac)
c ₇	İç kapak
c ₈	Dış kapak
c ₉	Hazır olan birleştirilmiş ürün

Çizelge 4.5: Şekil 4.4 Geçiş renkleri anlamı

Geçiş renkleri	Anlamları
a	Dış gövde için
b	İç gövde için
c	Sac iç boru için
d	Boru (plastik) iç boru için
e	İç kapı için
f	Dış kapı için
g	Hazır olan parçalar için

4.4.1 Örnek çalışma sonuçları

Programa girdi olarak başlangıç işaretlemesi, girdi matrisleri, çıktı matrisleri, Renkli Petri Ağı parametreleri (geçiş sayısı, yer sayısı, jeton rengi sayısı, geçiş rengi sayısı) ile varsa ağın erişip erişmediği merak edilen işaretleme verilir.

Çıktı olarak erişilebilir işaretleme ve aralarındaki bağlantı, P değişmezleri, aranılan işaretlemenin yeri ve bu işaretlemeye ulaşmak için uygulanması gereken ateşleme dizisi ile tüm bunları bulmak için programın harcadığı toplam süre verilir

Başlangıç işaretlemesi:

[illegible]

Programın aramasını istediğimiz işaretleme:

$$\text{Maranan} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

ve her geiş_rengi-jeton_rengi çifti için N ve O matrisleri (her geiş_rengi, her jeton_rengi ile ilgili olmadığından çoęu N ve O matrisleri sıfırlardan oluşmaktadır.) girdi olarak verilip program çalıştırılınca çıktı olarak aşağıdaki bilgiler alınmaktadır.

--RENKLI PETRI AGI CIKTI DOSYASI --

GIRDILER

Gecis_sayisi: 20 Gecis_rengi_sayisi: 7

Yer_sayisi: 16 Jeton_rengi_sayisi: 9

Baslangic isaretlemeesi:

110010110

000000000

000000000

000000000

000000000

000000000

000000000

000000000

000000000

000000000

000000000

000000000

000000000

000000000

000000000

000000000

CIKTILAR

Hesaplama suresi (mili saniye)-(saniye) :24531.000000 24.531000

Birbirinden farklı 1062 tane durum vardır.

Aradığınız işaretleme :

010000000

000000100

000000000

100000000

000000000

000000000

000000000

000000010

000000000

000000000

000000000

000000000

000001000

000000000

000000000

000000000

aranan: M1586 isimli işaretlemesidir ve

atesleme dizisi: t1,1 t1,5 t1,6 t2,4 t3,1 t6,4 t11,4 t20,6 ile ulaşılır.

(Atesleme dizisi t1,1..... : t1 geçişin adı, t1,1'deki son 1 ise geçiş renginin sıralamadaki numarasıdır, yani t1,1: t1,a anlamına gelmektedir.)

5. SONUÇLAR VE ÖNERİLER

Klasik Petri Ağı'nda birçok yer ve geçiş ile modellenecek olan bir sistem, RPA modelinde daha az yer ve geçişle birlikte renkli jeton ve geçişlerle modellenebilir. Böylece görsel olarak daha anlaşılır, daha küçük ve programlama mantığıyla daha uyumlu bir modelleme yapılabilir. RPA'lar ile fiziksel bir sistemin çalışması, bir yazılım veya bir kontrolörün tasarımı analiz edilebilir.

Bu çalışmada RPA tanımlanmış ve bileşenleri hakkında bilgiler verilmiş, örnek modellemeler yapılmış, örnekler için erişilebilirlik ağacı oluşturulmuş ve P değişmezleri hesaplanmıştır. RPA'larda erişilebilirlik ağacının elde edilmesi için bir algoritma tasarlanmış ve bu algoritma bir programa dönüştürülerek farklı RPA modelleri için denenmiş, doğru sonuçlar alındığı gözlenmiştir. P değişmezleri için daha önceden geliştirilen farklı yöntemler RPA'lar için uygulanmış, program haline getirilmiş ve yine farklı RPA modelleri için programın doğruluğu denenmiştir. Oluşturulan bu programlar bir araya getirilmiş, RPA'lar için erişilebilirlik ağacı ve P değişmezleri analizi yapan bir program oluşturulmuştur.

Bu çalışma doğrultusunda ilerideki çalışmalar için, P değişmezleri analizinde farklı algoritmalar denenebilir veya geliştirilebilir. Erişilebilirlik ağacı algoritması ayrıntılı incelenerek daha hızlı, farklı bir bakış açısına sahip algoritmalar geliştirilebilir. Ayrıca geliştirilen programın kullanımında Visual Basic arayüzü olmasına karşın verilen girdi ve çıktı matrislerine bağlı olarak RPA'nın grafiksel gösterimini yapan bir program geliştirilerek daha kullanıcı merkezli hale getirilebilir. Literatürde, grafiksel gösterimden yola çıkarak ağın analizini yapan çalışmalar mevcuttur fakat matematik model üzerinden ağın analizi konusunda hala yapılacak çalışmalar vardır. İleriki çalışmalarda bu yönde araştırmalar yapılabilir.

Ayrıca RPA ile süpervayzör kontrolör tasarımları, lojik devre modellemeleri vb. çalışmalar yapılarak uygulamaya dönük araştırmalar da yapılabilir. RPA'ların modelleme gücü kullanılarak birçok elektronik sistemin tasarımı ve analizi

yapılabilir. İleriki çalışmalarda bu gibi uygulamaya yönelik yenilikçi çalışmalarda RPA'lar kullanılarak yeni bir bakış açısıyla modellemeler yapılabilir.

KAYNAKLAR

- [1] **Petri, C.A.** 1962, Kommunikation mit Automaten, *Doctoral thesis*, Technical University Darmstadt, Germany.
- [2] **Jensen, K.**, 1993. *An Introduction to the Theoretical Aspects of Coloured Petri Nets*, *Lecture Notes In Computer Science*; Vol. 803, Springer-Verlag London, UK.
- [3] **Çayır, S.**, 2004. Petri Ağlarında Değişmez Analizi, *Yüksek Lisans Tezi*, İ.T.Ü. Fen Bilimleri Enstitüsü, İstanbul.
- [4] **Cassandras C. G., Lafortune S.**, 2nd ed., 2008,. *Introduction to Discrete Event Systems*, *Kluwer Academic Publishers.*, Dordrecht
- [5] **Ning, L.**, 2002. Power Systems Modeling Using Petri Nets, *Thesis of Doctor Of Philosophy*, Rensselaer Polytechnic Institute. Graduate Faculty of Rensselaer Polytechnic Institute Electric Power Engineering, Troy, New York
- [6] **Murata T.** April 1989. Petri Nets: Properties, Analysis And Applications, *Proceedings of The IEEE*, Vol.77, No.4.
- [7] **Ciardo, G.** 1989 , Analysis of Large Stochastic Petri Net Models. *Doctor of Philosophy Thesis*, Department of Computer Science Duke University
- [8] **Hrúz, B. ve Zhou M.C.**, 2002. *Modeling and Control of Discrete-event Dynamic Systems with Petri Nets and Other Tool*, Springer-Verlag London,UK.
- [9] **Darabi H. and Jafari M. A.**, 1994. Zero-One Programming of Petri Nets to Colored Petri Net Transformation, *Proceedings of the Fourth International Conference on Computer Integrated Manufacturing and Automation Technology*, IEEE, Sayfa 25 – 31.
- [10] **Ribeiro, A.S., Costa E.M.M and Lima II, E. J., C.**, 2008. Flexible Manufatcuting Systems Modeling Using High Level Petri Nets, *ABCM Symposium Series in Mechatronics*, Bahia, USSR, September 2008, Vol. 3 - pp.405-413

[11] **Bause F. ve Kritzinger P.**, 2002. *Stochastic Petri Nets - An Introduction to the Theory* -, Vieweg Verlag 2. Aufl., Germany.

[12] **Mert, V. ,Özkan, H. A. and Aybar, A.**. 19–21 October, 2007 , Modeling and Simulation of a Production System by Using Colored Petri Nets”, *ICCC 2007 5th IEEE International Conference on Computational Cybernetics*; IEEE, Gammarth/Tunisia.

[13] **Fanti, M.P., Giua, A., Seatzu, C.**, Available online 5 April 2006. Monitor design for colored Petri nets: An application to deadlock prevention in railway networks, *Control Engineering Practice* 14, Sayfa 1231–1247

[14] **Zurawski R. and Zhou M.C.**, 1994. Petri Nets and Industrial Applications: A Tutorial, *IEEE Transactions On Industrial Electronics*, Vol. 41. No. 6. pages 567-583

[15] **Recalde, L. and Ezpeleta, J.**, 2004. Petri Nets and Manufacturing Systems: An Examples-Driven Tour, *Lectures on Concurrency and Petri Nets*, Springer Berlin / Heidelberg

[16] **Koh, I. and DiCesare, F.**, August 1991,. Synthesis Rules for Colored Petri Nets and Their Applications to Automated Manufacturing Systems, *Proceedings of The 1991 IEEE International Symposium on Intelligent Control*.

[17] **Dong, J., Marinescu, D.C. , Lin C.**, 1987., *Reachability Graphs and Invariants for Stochastic High Level Petri Nets*. Purdue University, Report No. CSD-TR 686

[18] **Martinez,J., Silva, M.** 1982, A Simple and Fast Algorithm to Obtain All Invariants of a Generalised Petri Net, *Application and Theory of Petri Nets*, Informatik Fachberichte No.52, Springer, 301-310.

[19] **Lin, C. , Chaudhury, A. , Whinston, A. B. and Marinescu. D. C.**, June 1993 Logical Inference of Horn Clauses in Petri Net Models. *IEEE Transactions On Knowledge And Data Engineering*, IEEE, Vol. 5, No. 3

[20] **Svádová, M. , Hanzálek, Z.** , June 2001 Matlab Toolbox For Petri Nets. *22nd International Conference, ICATPN 2001* Newcastle upon Tyne, UK.

EKLER

EK A.1 : Program Kullanımı

EK A.1: Program Kullanımı

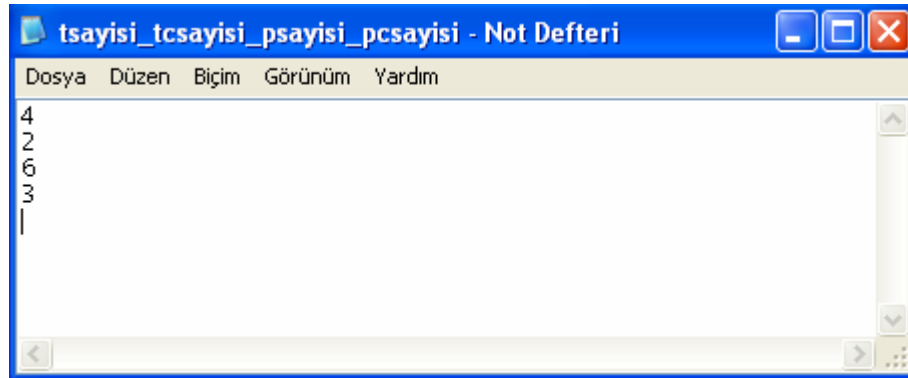
Programın Girdi Dosyaları:

Aşağıdaki metin dosyaları ilgili matrislerle, sırasıyla doldurulur.

1. tsayisi_tcsayisi_psayisi_pcsayisi.txt

Renkli Petri Ağı'nın elemanlarının sayısını verir. Sırasıyla alt alta $|T|$, $|C(T)|$, $|P|$ ve $|C(P)|$ sayıları yazılır.

Örn: $|T|=4$, $|C(T)|=2$, $|P|=6$, $|C(P)|=3$ için

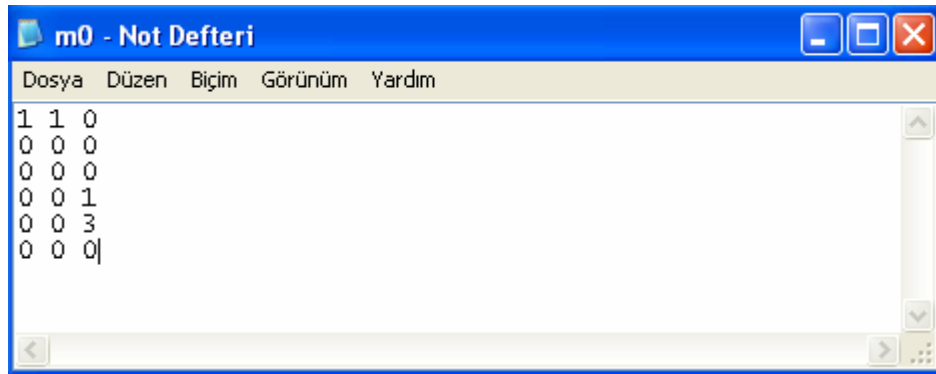


Şekil A.1: Parametre girdileri dosyası

2. m0.txt

Başlangıç işaretlemesi (ilk işaretleme)'nin bulunduğu metin dosyasıdır. Satırlar yerleri sütunlar jeton renklerini gösterir yani satır sayısı kadar yer(veya tersi), sütun sayısı kadar jeton rengi(veya tersi) bulunur.

Örn: $P=\{p_1, p_2, p_3, p_4, p_5, p_6\}$ ve $C(P)=\{c_1, c_2, c_3\}$



Şekil A.2: Başlangıç işaretlemesi dosyası

3. maranan.txt (isteğe bağlı)

Kullanıcının aradığı işaretlemeyi gösterir. Elde edilecek erişilebilirlik ağacında kullanıcının aradığı işaretlemeyi belirtir. Boyutları $m0$ matrisinin boyutları ile aynıdır.

Örn: $P=\{p_1, p_2, p_3, p_4, p_5, p_6\}$ ve $C(P)=\{c_1, c_2, c_3\}$



Şekil A.3: Aranılan işaretleme dosyası

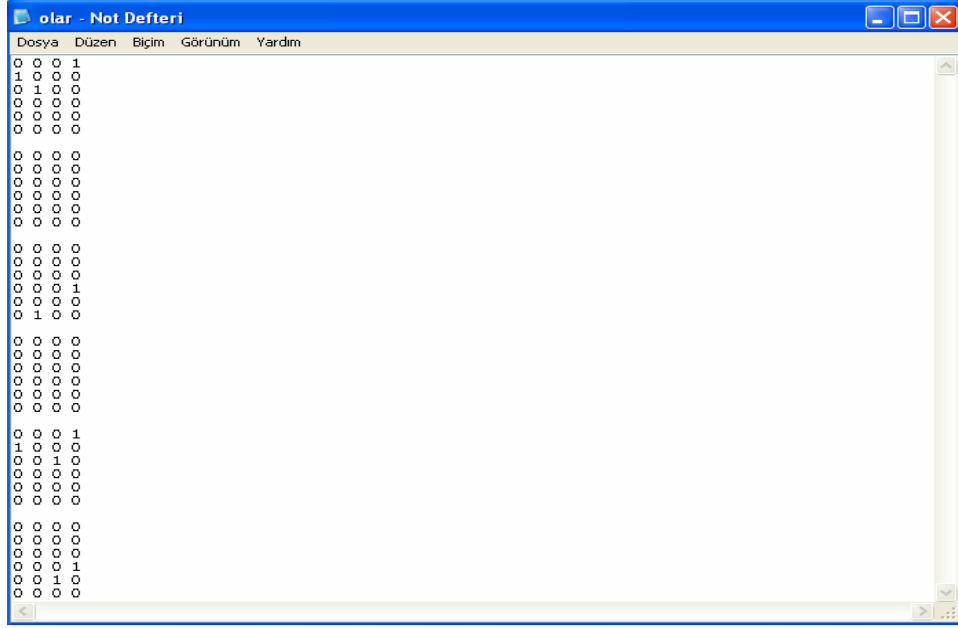
4. olar.txt

Renkli Petri Ağında (geçiş rengi sayısı x jeton rengi sayısı) kadar ($|C(T)| \times |C(P)|$ tane) $O(p,t)(\zeta,c)$ (Çıkış Matrisi) matrisi vardır ($\zeta \in C(T), c \in C(P)$). Bu $O(p,t)(\zeta,c)$ matrisleri $|P| \times |T|$ boyutundadır.

Örn: $|T|=4, |C(T)|=2, |P|=6, |C(P)|=3, C(P)=\{c_1, c_2, c_3\}, C(T)=\{a, b\}$ için

1. matris : a geçiş rengi ve c_1 jeton rengi için
2. matris : a geçiş rengi ve c_2 jeton rengi için
3. matris : a geçiş rengi ve c_3 jeton rengi için
4. matris : b geçiş rengi ve c_1 jeton rengi için
5. matris : b geçiş rengi ve c_2 jeton rengi için
6. matris : b geçiş rengi ve c_3 jeton rengi için

Not: her matris arasında bir satır boşluk bırakılarak yazılmalıdır.



Şekil A.4: Çıktı matrisleri dosyası

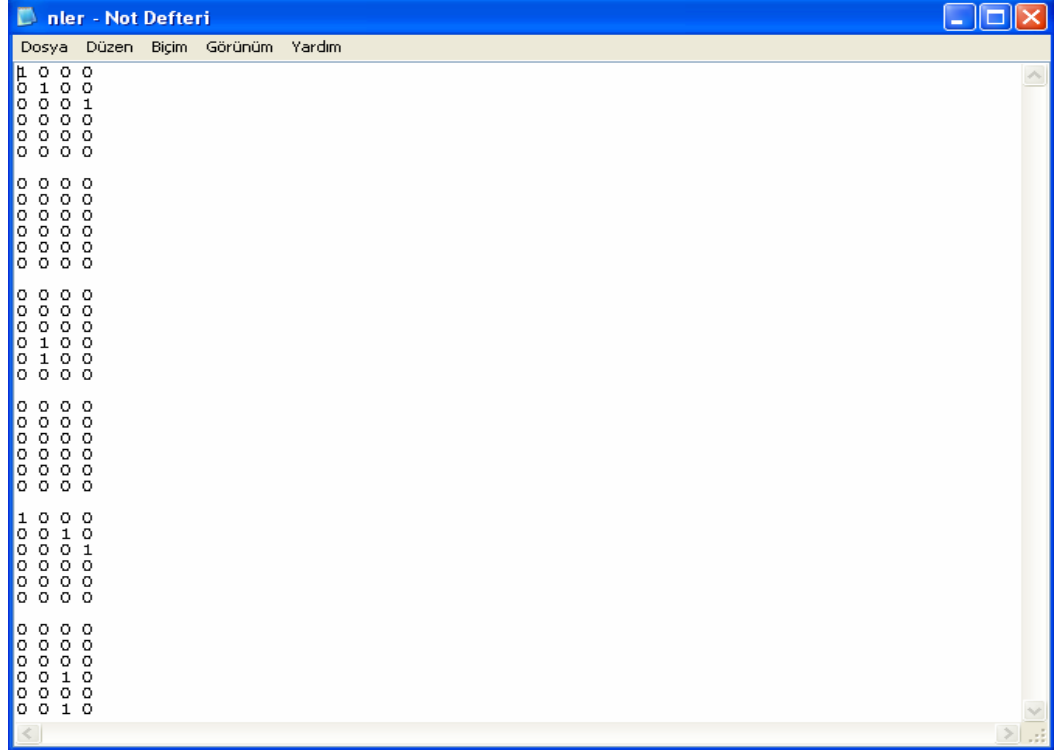
5. nler.txt

Renkli Petri Ağında geçiş rengi sayısı x jeton rengi sayısı kadar ($|C(T)| \times |C(P)|$ tane) $N(p,t)(\zeta,c)$ (Giriş Matrisi) matrisi vardır ($\zeta \in C(T), c \in C(P)$). Bu $N(p,t)(\zeta,c)$ matrisleri $|P| \times |T|$ boyutundadır.

Örn: $|T|=4, |C(T)|=2, |P|=6, |C(P)|=3, C(P)=\{c_1, c_2, c_3\}, C(T)=\{a, b\}$ için

1. matris : a geçiş rengi ve c_1 jeton rengi için
2. matris : a geçiş rengi ve c_2 jeton rengi için
3. matris : a geçiş rengi ve c_3 jeton rengi için
4. matris : b geçiş rengi ve c_1 jeton rengi için
5. matris : b geçiş rengi ve c_2 jeton rengi için
6. matris : b geçiş rengi ve c_3 jeton rengi için

Not: her matris arasında bir satır boşluk bırakılarak yazılmalıdır.



Şekil A.5: Girdi matrisleri dosyası

Programın Çıktı Dosyaları:

Aşağıdaki metin dosyaları programın çıktı dosyalarıdır.

1. erisebilirler.txt

Verilen başlangıç durumundan erişilebilen durumları, bu durumlara hangi durumdan hangi ateşleme sonucunda varılacağı yazar. Dosyanın en altında ise birbirinden farklı durum sayısı yazar.

2. ok.txt

Her bir durum için ateşlenebilecek geçiş ve rengini verir. Bir durum başka bir durum tarafından kapsanıyorsa bunu belirtir.

3. atdizi.txt

Her durum için başlangıç işaretlemesinden itibaren sırasıyla hangi ateşlemelerle gelineceğini belirtir.

4. pdegler.txt

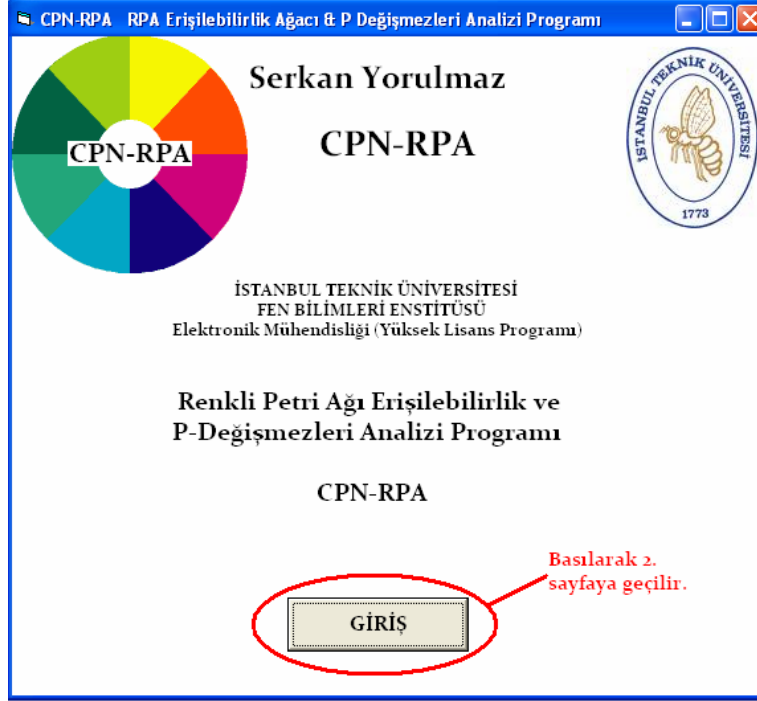
Her bir $(\zeta \in C(T), c \in C(P))$ (ζ, c) çifti için varsa P değişmezlerini veren $1 \times |P|$ boyutunda matrislerdir.

5. cikti.txt

Programın çıktılarını özetleyen dosyadır.

Program Ara Yüzü Kullanımı:

1. Programın ilk açılış sayfası aşağıdaki şekilde görüldüğü gibidir ve “Giriş” tuşuna basılarak programın 2. sayfası açılır.



Şekil A.6: Arayüz programı giriş sayfası

2. Daha sonra açılan 2. sayfada eğer yukarıda bahsedilen girdi dosyaları girilmişse “EVET” e basılır ve geçiş sayısı, geçiş renk sayısı, yer sayısı ve yer renk sayısı kutuları doldurulur.

Şekil A.7: Arayüz programı veri giriş sayfası

3. Eğer girdi dosyaları girilmemişse “HAYIR” tuşuna basılır ve girilmesi gereken girdi dosyaları açılır, ayrıca bir uyarı ekranıyla birlikte istenirse “CPN-RPA v1.0 Kullanım Klavuzu” açılabilir.

Şekil A.8: Arayüz programı açıklama sayfası

4. Geçiş sayısı, geçiş renk sayısı, yer sayısı ve yer renk sayısı kutuları doldurulduktan sonra “TAMAM” tuşuna basılır ve artık program çalıştırılmaya hazırdır. Daha sonra aktif hale gelen “Erişilebilirlik Ağacı ve P Değişmezleri HESAPLAMA İŞLEMİNİ BAŞLAT” tuşuna basarak program çalıştırılır.

CPN-RPA RPA Erişilebilirlik Ağacı & P Değişmezleri Analizi Programı

Serkan Yorulmaz

Erişilebilirlik Ağacı ve P-Değişmezleri Tespiti İçin Gerekli Bilgilerin Girilmesi

Giriş, Çıkış ve Mo (Başlangıç işaretlemesi) Matrislerini ilgili metin dosyalarına girdiniz mi?

EVET HAYIR

Aşağıdaki kutuları doğru şekilde doldurunuz ve ardından "Tamam" tuşuna basınız.

Geçiş Sayısı 4 Geçiş Renk Sayısı 2 Yer Sayısı 6 Yer Renk Sayısı 3

TAMAM

Erişilebilirlik Ağacı ve P-Değişmezleri HESAPLAMA İŞLEMİ BAŞLAT

Şekil A.9: Arayüz programı girişlerin yapılması sayfası

CPN-RPA RPA Erişilebilirlik Ağacı & P Değişmezleri Analizi Programı

Serkan Yorulmaz

Erişilebilirlik Ağacı ve P-Değişmezleri Tespiti İçin Gerekli Bilgilerin Girilmesi

Giriş, Çıkış ve Mo (Başlangıç işaretlemesi) Matrislerini ilgili metin dosyalarına girdiniz mi?

EVET HAYIR

Aşağıdaki kutuları doğru şekilde doldurunuz ve ardından "Tamam" tuşuna basınız.

Geçiş Sayısı 4 Geçiş Renk Sayısı 2 Yer Sayısı 6 Yer Renk Sayısı 3

TAMAM

Erişilebilirlik Ağacı ve P-Değişmezleri HESAPLAMA İŞLEMİ BAŞLAT

Şekil A.10: Arayüz programı analizin başlatılması sayfası

5. Son pencerede programlar çalışmış ve hesaplamalar yapılmıştır. İstenilen dosyalar açılarak gözlenebilir, yeni hesaplanacak ağ için program başa alınabilir veya program kapatılabilir.



Şekil A.11: Arayüz programı hesaplama sonrası çıktı sayfası

Serkan Yorulmaz

1985 yılında İstanbul’da doğdu. İlköğretim eğitimini İstanbul Bahçelievler İ.Ö.Okulunda, lise eğitimini İstanbul Pertevniyal Anadolu Lisesinde tamamladı.

2003-2007 yılları arasında Yıldız Teknik Üniversitesi’nde, Elektronik ve Haberleşme Mühendisliği bölümündeki eğitimini üçüncülükle bitirdi ve Elektronik ve Haberleşme Mühendisi ünvanı aldı. 2007 yılında İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Elektronik Mühendisliği yüksek lisans programına başladı.



Ad Soyad: Serkan Yorulmaz

Doğum Yeri ve Tarihi: İstanbul / 28.03.1985

Adres: Gültepe Mahallesi Ülkü Sokak No:9/6 Sefaköy İstanbul

Lisans Üniversite: Yıldız Teknik Üniversitesi

Yayın Listesi:

- Yorulmaz S., Kırıcı M., Behavioral and Structural Analysis of Automated Systems by Using Colored Petri Nets. *International Symposium on Innovations in Intelligent Systems and Applications*, June 29-July 1, 2009. TRABZON, TÜRKİYE. (Kabul edilmiştir.)

