

Valéria Maria Volpe

**Um Estudo Sobre a Modelagem de
Sistemas Usando Redes de Petri Coloridas**



UNIVERSIDADE ESTADUAL DE MARINGÁ
CENTRO DE CIÊNCIAS EXATAS
DEPARTAMENTO DE MATEMÁTICA

DISCIPLINA DE MATEMÁTICA

PROFESSOR: DR. JOÃO CARLOS

ALUNO: JOÃO CARLOS DE SOUZA
MATRÍCULA: 1910000772

1910000772



UM ESTUDO SOBRE A MODELAGEM DE
SISTEMAS USANDO REDES DE PETRI
COLORIDAS

Valéria Maria Volpe

Dissertação de Mestrado

Pós-Graduação em Matemática Aplicada e Computacional
MAC-020



unesp



UNIVERSIDADE ESTADUAL PAULISTA

Instituto de Biociências, Letras e Ciências Exatas
DEPARTAMENTO DE CIÊNCIAS DE COMPUTAÇÃO E ESTATÍSTICA

**UM ESTUDO SOBRE A MODELAGEM DE SISTEMAS USANDO
REDES DE PETRI COLORIDAS**

Valéria Maria Volpe

Dissertação de Mestrado



772/93

Pós-Graduação em Matemática Aplicada e Computacional

19



UM ESTUDO SOBRE A MODELAGEM DE SISTEMAS USANDO REDES DE PETRI COLORIDAS

Valéria Maria Volpe

Orientador Prof. Dr. Norian Marranghello

Dissertação apresentada à UNESP-IBILCE como parte dos requisitos para a obtenção do título de Mestre em Ciências Matemáticas - Área de Concentração Matemática Aplicada e Computacional.

São José do Rio Preto - SP - 1997.



**Dedico este trabalho
à minha
mãe
e minha avó
Tizira.**

Agradecimentos

Agradeço a **Deus** pela realização deste trabalho.

Agradeço ao meu orientador **Prof. Dr. Norian Marranghello** que gentilmente me recebeu como orientanda e sempre me ajudou nas horas mais difíceis, com sua amizade e compreensão.

Agradeço ao meu amigo **Carlos Magnus Carlson Filho** que sempre que pode me ajudou na pesquisa bibliográfica e todas as vezes me ofereceu seu apoio, sua colaboração e sua amizade.

Agradeço aos meus irmãos **Vanessa (Chu), Júlio (Peu) e Viviane (Vivi)** e ao meu pai **Adelmar** que sempre me apoiaram durante o desenvolvimento deste trabalho e em toda a minha caminhada.

Agradeço às minhas amigas **Silmara e Mara** que aceitaram a presença quase constante dos livros e artigos nos nossos churrascos de fins de semana.

Agradeço a todos das **FIRP - Faculdades Integradas Riopretense**, em especial ao Diretor Geral **Dr. Halim Atique Jr.** e à Diretora de Ensino, Pesquisa e Extensão **Profª Dra. Maria Aparecida Cória-Sabini** que sempre apoiaram o desenvolvimento deste trabalho, autorizando a minha participação em Congressos, Colóquios, ..., e que tem sempre apoiado o desenvolvimento da pesquisa nas FIRP. Também ao meu amigo **Fernando Dosualdo Moreira (Fer)** que sempre me socorreu com aquelas dicas infalíveis.

Enfim agradeço a todos que colaboraram direta ou indiretamente pela realização deste trabalho.



Um Estudo sobre a Modelagem de Sistemas usando Redes de Petri Coloridas

RESUMO : O objetivo deste trabalho é investigar as possibilidades de modelagem e simulação de sistemas por meio da linguagem para a descrição e a análise de processos, chamada **rede de Petri colorida**. Nesta dissertação, inicialmente, discutem-se os aspectos mais relevantes da teoria das duas principais classes de redes de Petri, quais sejam, as ordinárias e as de alto nível. Na sequência da dissertação são apresentadas as principais propriedades das **redes de Petri coloridas** e os métodos mais importantes de análise desta que, atualmente, é a representante de uso mais difundido dentre as **redes de Petri de alto nível**. Durante este trabalho de mestrado, foi desenvolvido um programa para analisar modelos descritos em redes de Petri coloridas, o qual foi denominado *CPNETVMV*. Este programa foi escrito na linguagem de programação *PASCAL 7.0* e desenvolvido para ambiente *MS-DOS*, pois, neste caso, a meta era criar um programa que qualquer pessoa pudesse usar em seu microcomputador pessoal, para estudos iniciais sobre redes de Petri coloridas. Portanto, a parte final desta dissertação apresenta as estruturas usadas neste programa, bem como a descrição da simulação de alguns modelos de sistemas, cuja análise foi utilizada para a verificação do comportamento do *CPNETVMV*. Uma discussão das potencialidades e limitações deste programa e sugestões para seu aprimoramento encerram esta monografia.

On system modeling using colored Petri nets

ABSTRACT : The goal of this work is to investigate the possibilities of systems modeling and simulation using **colored Petri nets**, a language for the description and analysis of processes. Initially, the most relevant aspects of the theory of the two main Petri net classes, i. e., ordinary and high-level, will be discussed. Thereafter, the main colored Petri net properties and the most important analysis methods of this net model, which presently is the most widely used representative of high-level Petri nets, will be presented. *CPNETVMV*, a program for the analysis of models described through colored Petri nets was developed during this work. This program is written in the *PASCAL 7.0* programming language under the *MS-DOS* environment. In this case, our goal was to create a program that could be used by anybody in her/his own personal computer during the initial phase of her/his studies on colored Petri nets. In the last part of this dissertation the data structures used in the program will be presented as well as the simulation of some systems models will be described. The analysis of such simulations were used in the assesement of *CPNETVMV* behavior. A discussion of the potencialities and limitations of this program, and sugestions for its enhancement conclude this monograph.

ÍNDICE

CAPÍTULO I	1
1 - INTRODUÇÃO ÀS REDES DE PETRI	1
1.1 - CARACTERÍSTICAS DE REDES	2
 CAPÍTULO II	9
2 - DEFINIÇÃO FORMAL DAS REDES DE PETRI.....	9
2.1 - REPRESENTAÇÃO DAS REDES DE PETRI.....	9
2.1.1 - Representação por Multi-Conjuntos.....	9
2.1.2 - Representação por Grafos.....	12
2.1.3 - Representação por Matriz.....	15
2.2 - MARCAÇÃO DE REDES DE PETRI.....	19
2.3 - REGRAS DE DISPARO DE REDES DE PETRI	23
2.4 - REDE DE PETRI PURA	25
2.5 SEQUÊNCIA DE DISPARO DE UMA REDE DE PETRI.....	26
2.6 - ESTADO DE UMA REDE DE PETRI	26
 CAPÍTULO III	28
3 - CLASSES DAS REDES DE PETRI	28
3.1- REDE DE PETRI ORDINÁRIA	28
3.1.1 - Rede Condição/Evento	28
3.1.2 - Rede Lugar/Transição.....	32
3.2 - REDES DE PETRI DE ALTO NÍVEL.....	37
3.2.1 - Redes de Petri Coloridas	37



CAPÍTULO IV	40
4 - PROPRIEDADES DAS REDES DE PETRI.....	40
4.1 - PROPRIEDADES COMPORTAMENTAIS	40
4.1.1 - Segurança (Safeness).....	40
4.1.2 - Limitação (Boundedness).....	41
4.1.3 - Vivacidade (liveness)	43
4.1.4 - Alcançabilidade (Reachability)	45
4.1.5 - Cobertura (Coverability)	46
4.1.6 - Reiniciabilidade (Reversibility).....	46
4.2 - PROPRIEDADES ESTRUTURAIS	47
4.2.1 Limitação Estrutural.....	48
4.2.2 - Conservação	48
CAPÍTULO V	51
5 - ANÁLISE DAS REDES DE PETRI.....	51
5.1 - MÉTODOS DE ANÁLISE DAS REDES DE PETRI	51
5.1.1 - Árvore de alcançabilidade	51
5.1.2 - Algoritmo de Construção da Árvore de Cobertura	53
5.1.3 - Equação fundamental das redes de Petri	55
5.2 - CÁLCULO DE INVARIANTES.....	56
5.2.1 - Invariantes de lugares das redes de Petri.....	56
5.2.2 - Invariantes de transição das redes de Petri	59
CAPÍTULO VI	61
6 - REDES DE PETRI COLORIDAS	61
6.1 - PRINCIPAIS MOTIVOS PARA A UTILIZAÇÃO DAS REDES DE PETRI COLORIDAS ..	61
6.2 - FUNCIONAMENTO DAS REDES DE PETRI COLORIDAS	62
6.3 - DEFINIÇÃO FORMAL DAS REDES DE PETRI COLORIDAS (CP-NETS)	65



6.4 - COMPORTAMENTO DINÂMICO DAS REDES DE PETRI COLORIDAS.....	67
6.5 - PROPRIEDADES COMPORTAMENTAIS DAS REDES DE PETRI COLORIDAS.....	69
6.5.1 - Limitação.....	70
6.5.2 - Vivacidade.....	70
6.5.3 - Reiniciabilidade.....	71
6.6 - ANÁLISE DAS REDES DE PETRI COLORIDAS.....	72
6.6.1 - Árvore de alcançabilidade de uma rede de Petri colorida.....	72
6.6.2 - Invariantes de lugar de uma rede de Petri colorida.....	73
 CAPÍTULO VII.....	 75
7 - UM PROGRAMA PARA ANÁLISE DE MODELOS DE SISTEMAS USANDO REDES DE PETRI COLORIDAS.....	 75
7.1 - DEFINIÇÃO DAS ESTRUTURAS CONSTRUÍDAS.....	75
7.1.1 - Estrutura dinâmica das matrizes.....	76
7.1.2 - Estrutura dinâmica da lista de marcações alcançáveis.....	81
7.2 - ORGANIZAÇÃO DO PROGRAMA.....	82
 CAPÍTULO VIII.....	 84
8 - ESTUDOS DE CASOS EM REDES DE PETRI COLORIDAS.....	84
8.1 - ANÁLISE DO MODELO DO SISTEMA DO JANTAR DOS FILÓSOFOS USANDO REDES DE PETRI COLORIDA.....	 84
8.2 - ANÁLISE DO MODELO DE UM SISTEMA DE MANUFATURA BASEADO NA FILOSOFIA KANBAN USANDO DOIS CARTÕES.....	 87
8.2.1 - Componentes do sistema kanban de dois cartões.....	88
8.2.2 - Fluxo do sistema kanban de dois cartões.....	89
8.2.3 - Representação do sistema kanban por CP-net.....	90



CAPÍTULO IX 95

9 - CONCLUSÃO..... 95

REFERÊNCIAS BIBLIOGRÁFICAS 97



CAPÍTULO I

1 - Introdução às Redes de Petri

O objetivo deste trabalho é estudar a representação de sistemas reais por meio de redes de Petri. Para atingir tal objetivo as redes de Petri foram utilizadas na modelagem de alguns sistemas. Os modelos resultantes foram simulados por meio de um programa desenvolvido no transcurso deste trabalho.

Neste capítulo, as redes de Petri serão apresentadas informalmente com o intuito de introduzir natural e suavemente alguns de seus conceitos fundamentais.

Redes de Petri é o nome genérico de uma teoria para modelagem de sistemas concorrentes. As ferramentas desenvolvidas com base nesta teoria auxiliam na simulação e na análise dos sistemas modelados.

As redes de Petri foram desenvolvidas por Carl Adam Petri no ano de 1962 na sua dissertação de Doutorado "Kommunikation mit Automaten" na Instituto de Matemática Aplicada da Universidade de Darmstadt na antiga Alemanha Ocidental. O trabalho de Carl Adam Petri atraiu a atenção de A. W. Holt que, em conjunto com vários outros pesquisadores desenvolveram muito da teoria, notação e representação das redes de Petri. A partir do desenvolvimento da teoria de redes de Petri várias contribuições científicas levaram ao desenvolvimento de várias de redes, onde as que mais se destacam são as redes de Petri de alto nível.

Em muitas áreas de estudos, fenômenos não são estudados diretamente, mas através de seu modelo. Um modelo é uma representação de sistema, muitas vezes em termos matemáticos, através do qual pode-se perceber as características importantes do objeto ou sistema estudado. As características físicas dos fenômenos, muitas vezes podem ser descritas por equações ou inequações. As redes de Petri aplicam-se sistemas, os quais



podem ser representados em termos matemáticos e cuja modelagem torna a compreensão do sistema mais fácil [Pet-81].

1.1 - Características de Redes

Muitos sistemas possuem partes independentes ou interagindo com outras partes, de forma simples ou complexa. Cada parte tem seu próprio estado inicial que é a informação necessária para descrever as ações futuras do sistema. O estado de cada uma das partes pode mudar no decorrer do tempo. O conceito de estado é muito importante para a modelagem de sistemas. Um estado de um sistema descreve o seu comportamento em um dado momento. Isto, por sua vez, representa as características atuais do sistema e pode servir para indicar se o sistema está se comportando corretamente ou não [Pet-81].

É possível representar uma rede de Petri por um grafo composto por dois tipos de nós, designados por lugares e transições. Os lugares são representados por círculos ou elipses enquanto as transições são representadas por retângulos ou barras. Os lugares e as transições são ligados por arcos direcionados, que representam a direção do fluxo da rede [Mac-96]. Tais elementos são representados na Figura 1.1.

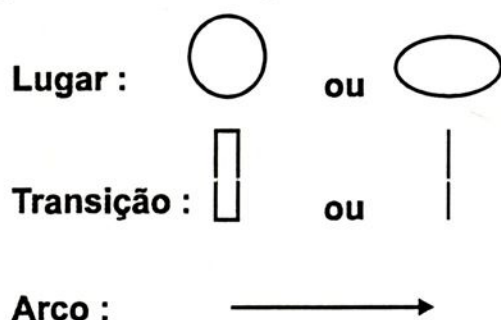


Figura 1.1 - Elementos para representar graficamente uma rede de Petri

Um lugar representa a parte passiva do sistema, uma transição representa a parte ativa do sistema. Isto significa que os lugares têm como objetivo armazenar informações de cada estado do sistema, as transições, por sua vez, representam as ações e são responsáveis pelas mudanças nos estados do sistema [Mac-96].

Para o melhor entendimento de uma rede de Petri, apresenta-se como exemplo uma rede representando os períodos de um dia. O dia pode ser dividido em três períodos : MANHÃ, TARDE e NOITE. Para que a MANHÃ se transforme em TARDE é necessário que ocorra o ENTARDECER, para que a TARDE passe a ser NOITE é necessário que ocorra o ANOITECER, assim como para que a NOITE de origem à MANHÃ é necessário que ocorra o AMANHECER [Mac-96]. A Figura 1.2 apresenta este ciclo sob a forma de uma rede de Petri.

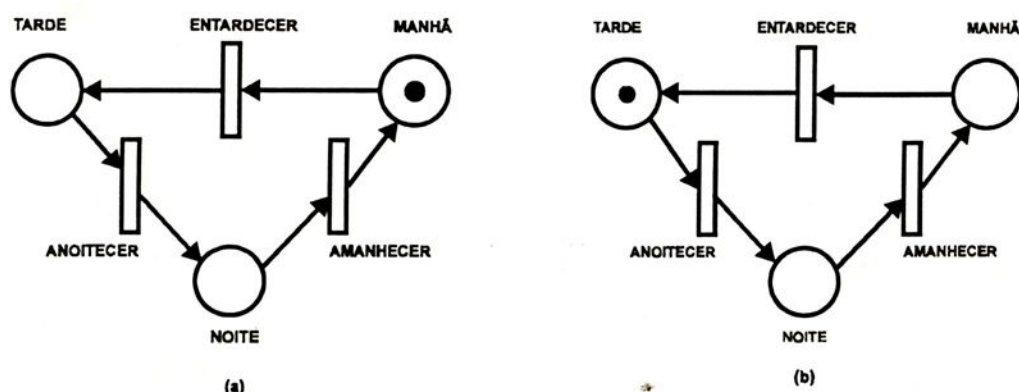


Figura 1.2 - Uma rede de Petri representando os Períodos do Dia [Mac-96].

Observe que na Figura 1.2(a) o estado da rede é MANHÃ, isto significa que o período do dia é MANHÃ, após ocorrer a transição ENTARDECER o estado da rede mudou para TARDE, como mostra a Figura 1.2(b). A “ficha preta” no lugar MANHÃ representa uma informação na rede, chamada de *marca*, o *estado inicial* da rede é chamado de *marcação inicial* da rede de Petri. Assim, pode-se dizer que uma rede de Petri possui um *estado inicial*, representado pela *marcação inicial* da rede.

Um lugar de entrada de uma transição é um lugar cujo arco que liga este lugar a uma transição tem origem neste lugar e destino na transição.

Um lugar de saída de uma transição é um lugar cujo arco que liga este lugar a uma transição tem origem nesta transição e destino neste lugar.

Só é possível gerar um novo estado para a rede se uma transição estiver habilitada para um disparo. Um disparo de uma transição habilitada significa eliminar todas as *marcas* dos lugares de entrada de uma transição habilitada e adicionar novas *marcas* aos seus lugares de saída. Uma transição

esta habilitada para ser disparada se existir *marcas* em todos os lugares de entrada de uma transição [Mac-96].

Uma pré-condição de uma transição é o conjunto cujos elementos são os lugares ligados à transição em questão por arcos que tem origem nos lugares e destino na transição.

Uma pós-condição de uma transição é o conjunto cujos elementos são os lugares ligados à transição em questão por arcos que tem origem na transição e destino nos lugares.

Pode-se ter vários arcos ligando um lugar a uma transição e vice-versa. Nesta situação, para que uma transição esteja habilitada é necessário que, em todos os lugares de entrada desta transição, exista um número de *marcas* maior ou igual à quantidade de arcos que ligam os nós lugares aos nós transições. Assim, quando a transição está habilitada, o seu disparo remove, dos lugares de entrada, a quantidade de *marcas* de acordo com o número de arcos e são adicionadas novas *marcas* aos lugares de saída de acordo com a quantidade de arcos que ligam a transição aos lugares de saída [Mac-96]. A Figura 1.3(a) representa uma rede de Petri com vários arcos ligando os lugares às transições e vice-versa. A Figura 1.3(b) representa o disparo da transição t_1 habilitada.

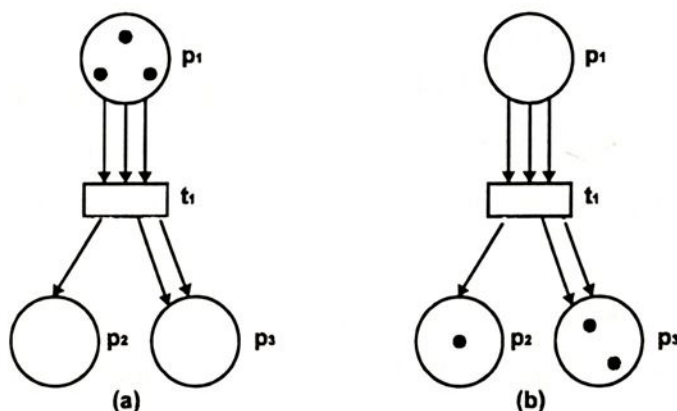


Figura 1.3 - (a) Uma rede de Petri com múltiplos arcos ligando os lugares às transições e vice-versa. (b) O disparo da transição t_1 habilitada.

Os arcos múltiplos podem ser substituídos por um único arco, mas com um valor que indica a quantidade de arcos que ligam os lugares às transições e vice-versa. Quando existir apenas um arco ligando um lugar a uma transição

na rede original, tal arco não terá nenhum valor inscrito na rede simplificada, ou seja, arcos sem inscrição têm valor unitário. A Figura 1.4 representa uma rede de Petri com valores inscritos nos arcos para indicar múltiplas ligações entre os nós da rede. Os valores atribuídos aos arcos são chamado de peso dos arcos [Mac-96].

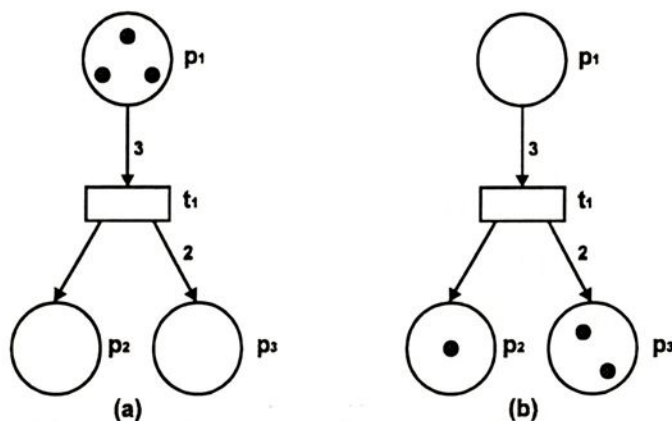


Figura 1.3 - (a) Uma rede de Petri com arcos rotulados ligando os lugares às transições e vice-versa. (b) O disparo da transição t_1 habilitada.

A representação das redes de Petri por meio de grafos torna-se interessante para facilitar a compreensão do seu funcionamento, mas a partir do momento que se deseja analisar as redes de Petri usando alguma ferramenta auxiliar, como por exemplo um computador, a representação por grafo torna-se difícil de ser usada. Uma alternativa para solucionar este problema é representar as redes de Petri usando matrizes, onde a representação gráfica é traduzida para a representação matricial e vice-versa. Esta representação permite a utilização de métodos formais de análise [Mac-96].

Tomando as linhas da matriz para representar as transições e suas colunas os lugares, usando a relação existente entre estes elementos, podem ser construídas matrizes que representam os arcos, que ligam os lugares às transições e vice-versa. Separando os lugares de entrada das transições dos lugares de saída constroem-se duas matrizes, que são denominadas matrizes de Entrada (I) e Saída (O). Quando existir um arco ligando um lugar a uma transição será atribuído, à posição da matriz correspondente ao cruzamento da

linha que representa a transição e com a coluna que representa o lugar, o peso do arco, caso não exista nenhum arco este valor será zero [Mac-96].

Para entender melhor esta representação matricial, como exemplo, suponha as seguintes atribuições, usando a Figura 1.2 :

Matriz de Entrada

Não existe nenhum arco ligando o lugar MANHÃ à transição AMANHECER então o elemento $I_{1,1}$ da matriz de entrada recebe o valor zero, ou seja,

$$I_{1,1} = \text{lugar de entrada MANHÃ na transição AMANHECER} = 0.$$

Assim como, existe um arco ligando o lugar NOITE à transição AMANHECER então o elemento $I_{1,3}$ da matriz de entrada recebe o peso do arco, neste caso o valor um, ou seja,

$$I_{1,3} = \text{lugar de entrada NOITE na transição AMANHECER} = 1$$

O mesmo acontece com a matriz de saída, ou seja

Matriz de Saída

Existe um arco ligando o lugar MANHÃ à transição AMANHECER, o elemento $O_{1,1}$ da matriz de saída recebe o peso do arco que liga o lugar à transição, ou seja,

$$O_{1,1} = \text{lugar de saída MANHÃ para a transição AMANHECER} = 1$$

Assim como não existe arco ligando o lugar TARDE à transição AMANHECER, o elemento $O_{1,2}$ da matriz de saída recebe o valor zero, ou seja,

$$O_{1,2} = \text{lugar de saída TARDE para a transição AMANHECER} = 0$$

Desta forma, a representação matricial da Figura 1.2 é dada por :

$$I = \begin{bmatrix} \text{MANHA} & \text{TARDE} & \text{NOITE} \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{matrix} \text{AMANHECER} \\ \text{ENTARDECER} \\ \text{ANOITECER} \end{matrix}$$

$$O = \begin{bmatrix} \text{MANHA} & \text{TARDE} & \text{NOITE} \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{matrix} \text{AMANHECER} \\ \text{ENTARDECER} \\ \text{ANOITECER} \end{matrix}$$

Define-se uma matriz de Incidência (D) associada a uma rede de Petri como sendo a matriz calculada a partir da diferença entre a matriz de Saída e a matriz de Entrada. Para o caso apresentado na Figura 1.2, a matriz de incidência é :

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

As características da matriz de incidência são [Gom-96] :

- O elemento D_{ij} representa a variação no número de *marcas* no lugar p_j quando é disparada a transição habilitada t_i ;
- A soma das entradas na linha k representa a diferença entre o número de *marcas* dos estados final e inicial de uma rede de Petri provocada pelo disparo da transição t_k , ou seja, suas linha estão associadas às transições;
- As colunas estão associadas aos lugares e representam seus ganhos ou perdas de *marcas* em função do disparo de cada transição.

As marcações de uma rede de Petri são representadas por um conjunto de vetores, onde os elementos dos vetores representam a quantidade de *marcas* presentes nos lugares, a cada estado da rede. Assim, chamar-se-ão de μ_0 a marcação inicial de uma rede de Petri, μ_1 a marcação gerada após o disparo de uma das transições habilitadas, a partir da marcação inicial μ_0 , e assim sucessivamente. Novas marcações podem ser geradas desde que pelo menos uma transição esteja habilitada. Este processo se repete enquanto existir uma transição habilitada a partir da marcação inicial da rede ou de suas sucessoras [Pet-81]. A Figura 1.5(b) representa a nova marcação gerada a partir do disparo da transição MONTA PACOTE :

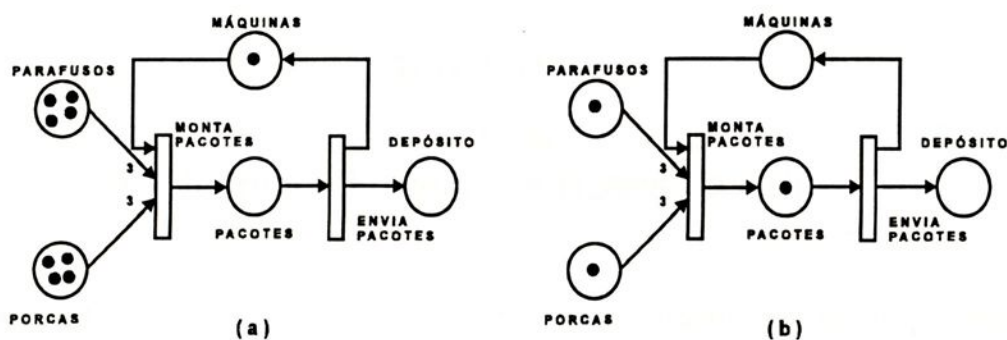


Figura 1.5 - redes de Petri representando uma linha de montagem e empacotamento de em conjunto de porcas e parafusos. 1.5(a) Marcação Inicial $\mu_0 = (4, 4, 1, 0, 0)$. 1.5(b) Marcação gerada após o disparo da transição Monta Pacote, $\mu_1 = (1, 1, 0, 1, 0)$ [Mac-96].

Deste modo, a execução de uma rede de Petri gera uma seqüência de marcações, com base numa seqüência de disparos de transições habilitadas a partir da marcação inicial μ_0 .

Várias técnicas de análise de redes de Petri podem ser usadas para caracterização de um conjunto de propriedades relativas ao modelo/sistema que estiver sendo estudado. Estas técnicas serão discutidas no capítulo V.

Através da análise das redes de Petri pode-se obter informações importantes sobre a estrutura e o comportamento do sistema modelado, as quais podem indicar a necessidade de alterações no sistema real, para a sua melhoria e seu aprimoramento.

CAPÍTULO II

2 - Definição Formal das Redes de Petri

As redes de Petri podem ser definidas formalmente de três formas diferentes, cada uma das quais tem como objetivo apresentar os conceitos fundamentais das redes de Petri, visando o entendimento exato de suas características.

Redes de Petri é uma técnica de especificação de sistemas que possibilita a sua representação matemática. Também, possui mecanismos de análise, que permitem a verificação das propriedades do sistema modelado e sua correção. As redes de Petri permitem modelar sistemas paralelos, concorrentes, assíncronos e não determinísticos [Mac-96].

Neste trabalho serão mostradas as representações de redes de Petri e discutidas quando a utilização de uma representação é mais apropriada que a outra. Assim como serão apresentadas algumas técnicas de análise de modelos descritos por redes de Petri.

2.1 - Representação das redes de Petri

Seu formalismo pode ser representado por técnicas matemáticas do tipo:

- Representação por Multi-Conjuntos.
- Representação por Grafos
- Representação por Matrizes

2.1.1 - Representação por Multi-Conjuntos

Usando a representação das redes de Petri por multi-conjuntos serão apresentadas as técnicas de modelagem utilizadas por esta representação, assim como as técnicas de análise.



Definição 2.1 - Multi-Conjunto : A teoria de multi-conjuntos é uma extensão da teoria de conjuntos. Um multi-conjunto, assim como um conjunto, é uma coleção de elementos sobre um domínio qualquer. Mas, diferentemente do conjunto, o multi-conjunto permite múltiplas ocorrências de elementos. Na teoria de conjuntos, um elemento pertence a um conjunto ou não pertence a um conjunto. Na teoria de multi-conjuntos, um elemento pode pertencer a um multi-conjunto nenhuma vez, uma vez ou um número qualquer de vezes [Pet-81].

Definição 2.2 - Redes de Petri por multi-conjunto: Uma estrutura de uma rede de Petri R é uma quádrupla $R = (P, T, I, O)$. $P = \{p_1, p_2, p_3, \dots, p_n\}$, onde P é o conjunto de lugares da rede de Petri R e $n \geq 0$ é a cardinalidade do conjunto P . $T = \{t_1, t_2, t_3, \dots, t_m\}$, onde T é o conjunto de transições da rede de Petri R e $m \geq 0$ é a cardinalidade do conjunto T . O conjunto de lugares e o conjunto de transições são disjuntos, ou seja, $P \cap T = \emptyset$. $I : T \rightarrow P^\infty$ é a função de entrada, um mapeamento do multi-conjunto de transições para o multi-conjunto de lugares. $O : T \rightarrow P^\infty$ é a função de saída, um mapeamento do multi-conjunto de transições para o multi-conjunto de lugares [Pet-81].

Um elemento qualquer do multi-conjunto de lugares P é denotado por p_i , $i = 1, 2, \dots, n$ e um elemento qualquer do multi-conjunto de transições T é denotado por t_j , $j = 1, 2, \dots, m$.

Um lugar p_i é um lugar de entrada de uma transição t_j se $p_i \in I(t_j)$, um lugar p_i é um lugar de saída de uma transição t_j se $p_i \in O(t_j)$. As entradas e saídas de uma transição são multi-conjuntos de lugares. A multiplicidade de um lugar de entrada p_i para uma transição t_j é o número de ocorrências de p_i no multi-conjunto de entrada da transição t_j , $\#(p_i, I(t_j))$. Assim como, A multiplicidade de um lugar de saída p_i para uma transição t_j é o número de ocorrências de p_i no multi-conjunto de saída da transição t_j , $\#(p_i, O(t_j))$.



Definição 2.3 - Extensões das Funções de entrada e saída : Seja a função de entrada I dada como $I : P \rightarrow T^\infty$ e a função de saída O dada como $O : P \rightarrow T^\infty$ tal que [Pet-81] :

$$\#(t_j, I(p_i)) = \#(p_i, O(t_j))$$

$$\#(t_j, O(p_i)) = \#(p_i, I(t_j))$$

A notação $\#(t_j, I(p_i))$ indica a multiplicidade do lugar de entrada p_i para a transição t_j , ou seja, o número de ocorrências do lugar p_i no multi-conjunto de entrada da transição t_j . Assim como a notação $\#(p_i, O(t_j))$ indica a multiplicidade do lugar de saída p_i para a transição t_j , ou seja, o número de ocorrências do lugar p_i no multi-conjunto de saída da transição t_j .

Veja alguns exemplos de redes de Petri descritas como multi-conjunto :

Exemplo 1 : Seja R a rede de Petri que representa os períodos do dia, então tem-se:

$$R_{\text{períodos_do_dia}} = (P, T, I, O)$$

o conjunto de Lugares é :

$$P = (\text{MANHÃ}, \text{TARDE}, \text{NOITE}),$$

o conjunto de Transições é :

$$T = (\text{AMANHECER}, \text{ENTARDECER}, \text{ANOITECER})$$

as funções de Entrada e Saída para cada um dos Lugares são :

$$I(\text{AMANHECER}) = \text{NOITE}$$

$$O(\text{AMANHECER}) = \text{MANHÃ}$$

$$I(\text{ENTARDECER}) = \text{MANHÃ}$$

$$O(\text{ENTARDECER}) = \text{TARDE}$$

$$I(\text{ANOITECER}) = \text{TARDE}$$

$$O(\text{ANOITECER}) = \text{NOITE}$$

Exemplo 2 : Seja R a rede de Petri que representa uma linha de produção e empacotamento de porcas e parafusos, então tem-se :

$$R_{\text{linha_de_produção}} = (P, T, I, O)$$

o conjunto de Lugares é :

$$P = \{\text{PORCAS}, \text{PARAFUSOS}, \text{PACOTE}, \text{MÁQUINA}, \text{DEPÓSITO}\}$$



conjunto de Transições é :

$$T = \{\text{MONTA PACOTE, ENVIA PACOTE}\}$$

as funções de Entrada e Saída para cada um dos Lugares são :

$$I(\text{MONTA PACOTE}) = \{\text{PORCAS, PARAFUSOS, MÁQUINA}\}$$

$$O(\text{MONTA PACOTES}) = \{\text{PACOTE}\}$$

$$I(\text{ENVIA PACOTE}) = \{\text{PACOTE}\}$$

$$O(\text{ENVIA PACOTE}) = \{\text{DEPÓSITO, MÁQUINA}\}$$

Exemplo 3 : Seja R uma rede de Petri qualquer [Pet-81], então tem-se :

$$R = (P, T, I, O)$$

conjunto de Lugares é :

$$P = \{p_1, p_2, p_3, p_4, p_5\}$$

conjunto de Transições é :

$$T = \{t_1, t_2, t_3, t_4, t_5\}$$

a função de Entrada e Saída para cada um dos Lugares são :

$$I(t_1) = \{p_1\}$$

$$O(t_1) = \{p_2, p_3\}$$

$$I(t_2) = \{p_3\}$$

$$O(t_2) = \{p_3, p_5, p_5\}$$

$$I(t_3) = \{p_2, p_3\}$$

$$O(t_3) = \{p_2, p_4\}$$

$$I(t_4) = \{p_4, p_5, p_5, p_5\}$$

$$O(t_4) = \{p_4\}$$

$$I(t_5) = \{p_2\}$$

$$O(t_5) = \{p_5\}$$

2.1.2 - Representação por Grafos

A representação mais comum das redes de Petri é em forma de grafos.

A utilização de um grafo permite uma percepção do comportamento do modelo claramente mais intuitiva.

Definição 2.4 - *Grafo direcionado bipartido* : Seja $G = (V, A)$ um grafo, onde V é seu o conjunto de nós e A o seu conjunto de arestas. Um grafo G é dito um grafo bipartido porque possui dois tipos de conjuntos de nós P e T , onde o conjunto de nós $V = P \cup T$ e $P \cap T = \emptyset$. Toda aresta de G une um nó de P a um nó de T . Um grafo G é dito um grafo direcionado se as arestas são pares



ordenados de vértices (v, w) onde v é o nó origem w o nó destino. Assim, cada aresta tem uma única direção, de v para w . Isto é :

$$A = \{\forall a_i \in A \mid a_i = (v, w) \rightarrow [v \in P, w \in T] \vee [w \in P, v \in T]\}.$$

Definição 2.5 - Redes de Petri por Grafos: Seja G a representação gráfica de uma rede de Petri, onde G é um grafo direcionado bipartido. $G = (V, A)$, onde $V = \{v_1, v_2, v_3, \dots, v_n\}$ é o multi-conjunto dos vértices e $A = \{a_1, a_2, a_3, \dots, a_m\}$ é o multi-conjunto dos arcos direcionados, $a_i = (v_j, v_k)$, com $v_j, v_k \in V$. O conjunto V é particionado em dois conjuntos disjuntos P e T , que representam os lugares e as transições da rede de Petri, onde $V = P \cup T$ e $P \cap T = \emptyset$, e para cada arco direcionado $a_i \in A$, se $a_i = (v_j, v_k)$, então $v_j \in P$ e $v_k \in T$ ou então $v_k \in P$ e $v_j \in T$ [Pet-81].

Veja alguns exemplos de redes de Petri descritas como Grafos:

Exemplo 1 : Seja R uma rede de Petri que representa as quatro estações do ano [Rei-85], então tem-se :

$$R_{\text{estações_do_ano}} = (P, T, A)$$

conjunto de Lugares é :

$$P = \{\text{Outono, Inverno, Primavera, Verão}\}$$

conjunto das Transições é

$$T = \{\text{Início do Inverno, Início da Primavera, Início do Verão, Início do Outono}\}$$

e as relações existentes entre os lugares P e as transições T representadas pelo conjunto A são :

$$A = \{(\text{Primavera, Início do Verão}), (\text{Início do Verão, Verão}), (\text{Verão, Início do Outono}), (\text{Início do Outono, Outono}), (\text{Outono, Início do Inverno}), (\text{Início do Inverno, Inverno}), (\text{Inverno, Início da Primavera}), (\text{Início da Primavera, Primavera})\}$$

O grafo da rede de Petri $R_{\text{estações_do_ano}} = (P, T, A)$ é mostrado na Figura 2.1

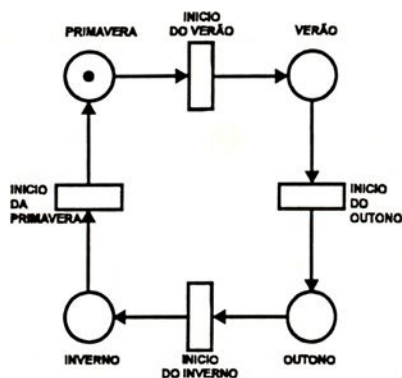


Figura 2.1 - Representação por grafo da Rede de Petri representando as quatro estações do ano [Rei-85].

Exemplo 2 : Seja R uma rede de Petri que representa um sistema com um PRODUTOR, um CONSUMIDORES e um DEPÓSITO [Gom-96]. Uma interpretação da rede de Petri R pode ser associada a um sistema onde os lugares p_1 e p_2 modelam as atividades do produtor, os lugares p_4 e p_5 as do consumidor, enquanto p_3 se encontra associado ao processo que armazenará temporariamente os “produtos produzidos”. Desta forma, a presença de uma marca em p_1 modela a disponibilidade de “produtos produzidos” prontos para serem armazenados. Enquanto que a marca em p_2 indica que o sistema está em condições de “produzir”. As transições t_1 e t_2 representam as ações “produzir” e “transferir para o armazém”, respectivamente. Uma marca em p_3 e uma marca em p_5 indicam que o consumidor está em condições de “consumir o produto armazenado”. Uma marca em p_4 indica “produto consumido”. As transições t_3 e t_4 representam as ações “consumir produto” e “consumir”, respectivamente. Portanto tem-se:

$$R_{\text{PRODUTOR_CONSUMIDOR}} = (P, T, A)$$

conjunto de Lugares é :

$$P = \{p_1, p_2, p_3, p_4, p_5\}$$

conjunto das Transições é

$$T = \{t_1, t_2, t_3, t_4\}$$

e as relações existentes entre os lugares P e as transições T são representadas pelo conjunto A, como segue :

$$A = \{(p_1, t_2), (t_2, p_2), (t_2, p_3), (p_2, t_1), (t_1, p_1), (p_3, t_3), (t_3, p_4), (p_4, t_4), (t_4, p_5), (p_5, t_3)\}$$

O grafo da rede de Petri $R_{\text{PRODUTOR_CONSUMIDOR}} = (P, T, A)$ é mostrado na Figura 2.2.

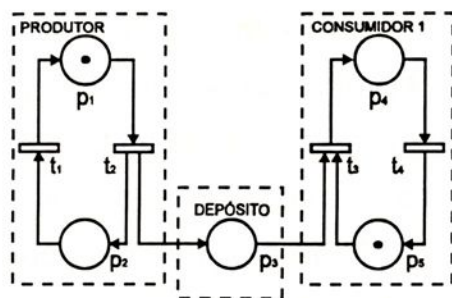


Figura 2.2 - Rede de Petri representando um sistema com um PRODUTOR, um CONSUMIDORES e um DEPÓSITO. [Gom-96]

2.1.3 - Representação por Matriz

A representação matricial das redes de Petri é mais utilizada para a implementação computacional de suas teorias, possibilitando a análise das propriedades comportamentais. Esta representação é fortemente usada para se estudar as redes de Petri utilizando a computação como ferramenta de análise, pois os resultados obtidos nos estudos da álgebra matricial fortalecem-nas.

Definição 2.6 - Pré-condições : As pré-condições de uma transição t_j formam um conjunto cujos elementos são lugares p_i ligados a transição t_j por arcos com origem em p_i e término em t_j . Formalmente, sejam $p_i \in P$ e $t_j \in T$, se $p_i \in I(t_j)$ então p_i é uma pré-condição de t_j . O conjunto das pré-condições de uma transição t_j é denotado por $\bullet t_j$.

Definição 2.7 - Pós-condições : As pós-condições de uma transição t_j formam um conjunto cujos elementos são lugares p_i ligados a transição t_j por arcos com origem em t_j e término em p_i . Formalmente, sejam $p_i \in P$ e $t_j \in T$, se $p_i \in O(t_j)$ então p_i é uma pós-condição de t_j . O conjunto das pós-condições de uma transição t_j é denotado por $t_j^\bullet$.

Definição 2.8 - Redes de Petri por Matrizes : Considere uma matriz com m linhas representando as transições e n colunas representando os lugares de uma rede de Petri. O valor do elemento na i -ésima coluna da j -ésima linha é igual ao peso do arco que liga o lugar p_i à transição t_j . Para tanto considera-se como sendo um arco de peso zero se o lugar p_i não tiver ligação com a transição t_j . Se p_i pertencer ao conjunto de entrada de t_j o arco que os une contribuirá com o recíproco do valor de seu peso. Se p_i pertencer ao conjunto de saída de t_j o arco que os une contribuirá com o valor absoluto de seu peso. Formalmente, a representação de uma rede de Petri por matrizes é uma quádrupla $R = (P, T, I, O)$, onde P é um conjunto finito de lugares, T é um conjunto finito de transições. $I : P \times T \rightarrow \mathbb{N}^+$ é uma matriz de entrada. $O : P \times T \rightarrow \mathbb{N}^+$ é uma matriz de saída [Mac-96].

A seguir serão apresentados alguns exemplos da representação matricial de redes de Petri :

Exemplo 1 : Seja R a rede de Petri que representa as quatro estações do ano vista na seção anterior então tem-se :

$$R_{\text{estações_do_ano}} = (P, T, I, O)$$

conjunto de Lugares é :

$$P = \{\text{Outono, Inverno, Primavera, Verão}\}$$

conjunto das Transições é

$$T = \{\text{Início do Inverno, Início da Primavera, Início do Verão, Início do Outono}\}$$

as Pré-condições para cada uma das Transições são :

$$I(\text{INÍCIO DO VERÃO}) = \text{PRIMAVERA}$$

$$I(\text{INÍCIO DO OUTONO}) = \text{VERÃO}$$

$$I(\text{INÍCIO DO INVERNO}) = \text{OUTONO}$$

$$I(\text{INÍCIO DO PRIMAVERA}) = \text{INVERNO}$$



a Matriz das Pré-condições é :

$$I = \begin{bmatrix} \text{PRIMAVERA} & \text{VERAO} & \text{OUTONO} & \text{INVERNO} \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{matrix} \text{INICIO DO VERA} \\ \text{INICIO DO OUTONO} \\ \text{INICIO DO INVERNO} \\ \text{INICIO DA PRIMAVERA} \end{matrix}$$

as Pós-condições para cada uma das Transições são :

O(INÍCIO DO VERÃO) = VERÃO

O(INÍCIO DO OUTONO) = OUTONO

O(INÍCIO DO INVERNO) = INVERNO

O(INÍCIO DO PRIMAVERA) = PRIMAVERA

a Matriz das Pós-condições é :

$$O = \begin{bmatrix} \text{PRIMAVERA} & \text{VERAO} & \text{OUTONO} & \text{INVERNO} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{matrix} \text{INICIO DO VERA} \\ \text{INICIO DO OUTONO} \\ \text{INICIO DO INVERNO} \\ \text{INICIO DA PRIMAVERA} \end{matrix}$$

Exemplo 2 : Seja R a rede de Petri que representa uma linha de produção e empacotamento de porcas e parafusos então tem-se :

$$R_{\text{linha_de_produção}} = (P, T, I, O)$$

o conjunto de Lugares é :

$$P = \{\text{PORCAS, PARAFUSOS, PACOTE, MÁQUINA, DEPÓSITO}\}$$

conjunto de Transições é :

$$T = \{\text{MONTA PACOTE, ENVIA PACOTE}\}$$

as Pré-condições para cada uma das Transições são :

$$I(\text{MONTA PACOTE}) = \{\text{PORCAS, PARAFUSOS, MÁQUINA}\}$$

$$I(\text{ENVIA PACOTE}) = \{\text{PACOTE}\}$$

a Matriz das Pré-condições é :

$$I = \begin{bmatrix} \text{PARAFUSOS} & \text{PORCAS} & \text{PACOTES} & \text{MAQUINA} & \text{DEPOSITO} \\ 3 & 3 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{matrix} \text{MONTA PACOTE} \\ \text{ENVIA PACOTE} \end{matrix}$$

as Pós-condições para cada uma das Transições são :

$$O(\text{MONTA PACOTES}) = \{\text{PACOTE}\}$$

$$O(\text{ENVIA PACOTE}) = \{\text{DEPÓSITO, MÁQUINA}\}$$

a Matriz das Pós-condições é :

$$O = \begin{bmatrix} \text{PARAFUSOS} & \text{PORCAS} & \text{PACOTES} & \text{MAQUINA} & \text{DEPOSITO} \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} \begin{matrix} \text{MONTA PACOTE} \\ \text{ENVIA PACOTE} \end{matrix}$$

Exemplo 3 : Seja R a rede de Petri que representa os períodos do dia então tem-se:

$$R_{\text{períodos_do_dia}} = (P, T, I, O)$$

o conjunto de Lugares é :

$$P = (\text{MANHÃ, TARDE, NOITE}),$$

o conjunto de Transições é :

$$T = (\text{AMANHECER, ENTARDECER, ANOITECER})$$

as Pré-condições para cada uma das Transições são :

$$I(\text{AMANHECER}) = \text{NOITE}$$

$$I(\text{ENTARDECER}) = \text{MANHÃ}$$

$$I(\text{ANOITECER}) = \text{TARDE}$$

a Matriz das Pré-condições é :

$$I = \begin{bmatrix} \text{MANHA} & \text{TARDE} & \text{NOITE} \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{matrix} \text{AMANHECER} \\ \text{ENTARDECER} \\ \text{ANOITECER} \end{matrix}$$

as Pós-condições para cada uma das Transições são :

$$O(\text{AMANHECER}) = \text{MANHÃ}$$

$$O(\text{ENTARDECER}) = \text{TARDE}$$

$$O(\text{ANOITECER}) = \text{NOITE}$$

a Matriz das Pós-condições é :

$$O = \begin{bmatrix} \text{MANHA} & \text{TARDE} & \text{NOITE} \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{matrix} \text{AMANHECER} \\ \text{ENTARDECER} \\ \text{ANOITECER} \end{matrix}$$

2.2 - Marcação de redes de Petri

Uma marcação de uma rede de Petri indica a presença das *marcas* em seus lugares. O número e a posição das *marcas* se alteram durante a execução da rede.

Definição 2.9 - *Marcação de uma rede de Petri* : Uma marcação μ de uma rede de Petri $R = (P, T, I, O)$ é uma função de um conjunto de lugares para o conjunto de inteiros não negativos, ou seja, $\mu : P \rightarrow \mathbb{N}^+$ [Pet-81].

Definição 2.10 - *Representação da Marcação de uma rede de Petri por Vetor* : Uma marcação μ de uma rede de Petri $R = (P, T, I, O)$ é um vetor n dimensional $\mu = (\mu_1, \mu_2, \mu_3, \dots, \mu_n)$, onde n é a cardinalidade do conjunto P e cada $\mu_i \in \mathbb{N}^+$, $i = 1, 2, 3, \dots, n$ [Pet-81].

Vetor μ indica a quantidade de *marcas* em cada um dos lugares da rede de Petri. O número de *marcas* do lugar p_i é dado por μ_i , $i = 1, 2, 3, \dots, n$, denotado por $\mu(p_i) = \mu_i$ [Pet-81].

Definição 2.11 - *Rede de Petri Marcada* : Se $M = (R, \mu)$ é uma rede de Petri marcada, $R = (P, T, I, O)$ representa a estrutura da rede e μ a sua marcação. Alternativamente pode-se escrever $M = (P, T, I, O, \mu)$ [Pet-81].

Em um grafo as *marcas* são representadas por pequenos pontos pretos nos círculos ou elipses que representam os lugares de uma rede de Petri.

Em matrizes e multi-conjuntos as *marcas* são representadas por vetores como na **Definição 2.10**, ou seja, usa-se a representação por vetor para indicar a quantidade de *marcas* em cada um dos lugares da rede.

Veja alguns exemplos de redes marcadas :

Exemplo 1 : Seja $R = (P, T, I, O, \mu)$ uma rede de Petri que modela o problema do jantar dos filósofos proposto por Dijkstra. Suponha a existência de três filósofos que podem estar comendo, pensando ou com fome. Os filósofos

estão sentados em volta de uma mesa, cada filósofo tem um prato à sua frente, um garfo à sua direita e outro garfo à sua esquerda, mas seus garfos são compartilhados com os filósofos que estão a seu lado. Assim quando um filósofo estiver comendo os outros não podem estar comendo. Conseqüentemente, devem estar pensando ou com fome esperando a liberação dos garfos. Para facilitar a compreensão da modelagem deste problema observe o quadro abaixo [Mac-96] :

Lugares	Transições
p ₁ - PENSANDO1	t ₁ - COMEÇA PENSAR1
p ₂ - COM FOME1	t ₂ - COMEÇA SENTIR FOME1
p ₃ - COMENDO1	t ₃ - TENTA COMER1
p ₄ - PENSANDO2	t ₄ - COMEÇA PENSAR2
p ₅ - COM FOME2	t ₅ - COMEÇA SENTIR FOME2
p ₆ - COMENDO2	t ₆ - TENTA COMER2
p ₇ - PENSANDO3	t ₇ - COMEÇA PENSAR3
p ₈ - COM FOME3	t ₈ - COMEÇA SENTIR FOME3
p ₉ - COMENDO3	t ₉ - TENTA COMER3
p ₁₀ - GARFO1	
p ₁₁ - GARFO2	
p ₁₂ - GARFO3	

Então tem-se :

Por Multi-Conjunto :

$$R_{jantar_dos_filósofos} = (P, T, I, O, \mu)$$

o conjunto de Lugares é :

$$P = \{p_1, p_2, p_3, p_4, p_5, p_6, p_7, p_8, p_9, p_{10}, p_{11}, p_{12}\}$$

o conjunto de Transições é :

$$T = \{t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8, t_9\}$$

as funções de Entrada e Saída para cada um dos Lugares são :

$I(t_1) = \{p_3\}$	$O(t_1) = \{p_1, p_{10}, p_{12}\}$
$I(t_2) = \{p_1\}$	$O(t_2) = \{p_2\}$
$I(t_3) = \{p_2, p_{10}, p_{12}\}$	$O(t_3) = \{p_3\}$
$I(t_4) = \{p_6\}$	$O(t_4) = \{p_4, p_{11}, p_{10}\}$
$I(t_5) = \{p_4\}$	$O(t_5) = \{p_5\}$



$$I(t_6) = \{p_5, p_{11}, p_{10}\}$$

$$O(t_6) = \{p_6\}$$

$$I(t_7) = \{p_9\}$$

$$O(t_8) = \{p_7, p_{12}, p_{11}\}$$

$$I(t_8) = \{p_7\}$$

$$O(t_8) = \{p_8\}$$

$$I(t_9) = \{p_8, p_{12}, p_{11}\}$$

$$O(t_9) = \{p_9\}$$

Vetor Marcação μ da rede é :

$$\mu = (0, 0, 1, 1, 0, 0, 0, 1, 0, 0, 1, 0)$$

Por Grafo :

$$R_{jantar_dos_filósofos} = (P, T, A, \mu)$$

o conjunto de Lugares é :

$$P = \{p_1, p_4, p_7, p_2, p_5, p_8, p_3, p_6, p_9, p_{10}, p_{11}, p_{12}\}$$

conjunto de Transições é :

$$T = \{t_1, t_4, t_8, t_2, t_5, t_8, t_3, t_6, t_9\}$$

e as relações existentes entre os lugares P e as transições T são representadas pelo conjunto A como segue :

$$A = \{(t_1, p_1), (t_1, p_{10}), (t_1, p_{12}), (p_1, t_2), (t_2, p_2), (p_2, t_3), (t_3, p_3), (t_3, p_{10}), (t_3, p_{12}), (p_3, t_1), (t_4, p_4), (t_4, p_{11}), (t_4, p_{10}), (p_4, t_2), (t_5, p_5), (p_5, t_6), (t_6, p_6), (t_6, p_{10}), (t_6, p_{11}), (t_6, p_4), (t_8, p_7), (t_8, p_{12}), (t_8, p_{11}), (p_7, t_8), (t_8, p_8), (p_8, t_9), (t_9, p_9), (t_9, p_{12}), (t_9, p_{11}), (t_9, p_8)\}$$

A Figura 2.3 apresenta o grafo da rede de Petri $R_{jantar_dos_filósofos} = (P, T, A)$

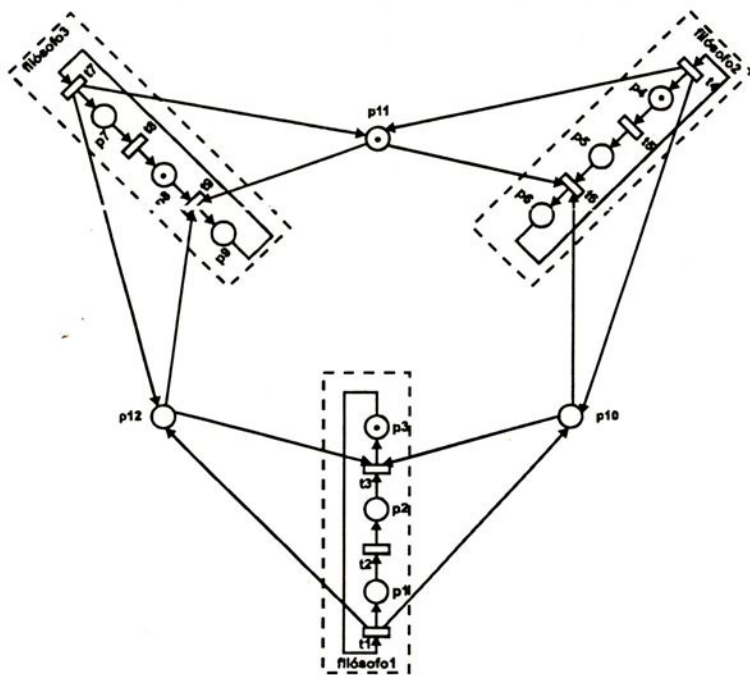


Figura 2.3 - Rede de Petri representando o Problema do Jantar dos Filósofos. Representação por Grafo [Mac-96].

O vetor marcação μ da rede é :

$$\mu = (0, 0, 1, 1, 0, 0, 0, 1, 0, 0, 1, 0)$$

Por matriz :

$$R_{\text{jantar_dos_filósofos}} = (P, T, I, O, \mu)$$

o conjunto de Lugares é :

$$P = \{p_1, p_2, p_3, p_4, p_5, p_6, p_7, p_8, p_9, p_{10}, p_{11}, p_{12}\}$$

o conjunto de Transições é :

$$T = \{t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8, t_9\}$$

as Pré-condições para cada uma das Transições são :

$$I(t_1) = \{p_3\}$$

$$I(t_4) = \{p_6\}$$

$$I(t_8) = \{p_9\}$$

$$I(t_2) = \{p_1\}$$

$$I(t_5) = \{p_4\}$$

$$I(t_8) = \{p_7\}$$

$$I(t_3) = \{p_2, p_{10}, p_{12}\}$$

$$I(t_6) = \{p_5, p_{11}, p_{10}\}$$

$$I(t_9) = \{p_8, p_{12}, p_{11}\}$$

a Matriz das Pré-condições é :

$$I = \begin{bmatrix} p_1 & p_2 & p_3 & p_4 & p_5 & p_6 & p_7 & p_8 & p_9 & p_{10} & p_{11} & p_{12} \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \end{bmatrix} \begin{matrix} t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \\ t_6 \\ t_7 \\ t_8 \\ t_9 \end{matrix}$$

as Pós-condições para cada uma das Transições são :

$$O(t_1) = \{p_1, p_{10}, p_{12}\}$$

$$O(t_4) = \{p_4, p_{11}, p_{10}\}$$

$$O(t_8) = \{p_7, p_{12}, p_{11}\}$$

$$O(t_2) = \{p_2\}$$

$$O(t_5) = \{p_5\}$$

$$O(t_8) = \{p_8\}$$

$$O(t_3) = \{p_3\}$$

$$O(t_6) = \{p_6\}$$

$$O(t_9) = \{p_9\}$$

a Matriz das Pós-condições é :

$$O = \begin{bmatrix} p_1 & p_2 & p_3 & p_4 & p_5 & p_6 & p_7 & p_8 & p_9 & p_{10} & p_{11} & p_{12} \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \begin{matrix} t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \\ t_6 \\ t_7 \\ t_8 \\ t_9 \end{matrix}$$

O vetor marcação μ da rede é :

$$\mu = (0, 0, 1, 1, 0, 0, 0, 1, 0, 0, 1, 0)$$

2.3 - Regras de Disparo de redes de Petri

Para executar uma rede de Petri é necessário que alguma transição esteja habilitada. A execução de uma rede consiste em controlar o número e a distribuição de *marcas* na rede. Uma transição habilitada indica que é possível dispará-la. O disparo de uma transição indica a remoção de *marcas* dos lugares de entrada e o acréscimo de novas *marcas* aos lugares de saída desta transição. Uma transição está habilitada se o número de *marcas* em seus lugares de entrada é, no mínimo, igual ao peso dos arcos que unem tais lugares à transição.

Definição 2.12 - *Transição habilitada em uma rede de Petri* : Uma transição $t_j \in T$ em uma rede Marcada $R = (P, T, I, O, \mu)$ está habilitada se para todo $p_i \in P$, $\mu(p_i) \geq \#(p_i, I(t_j))$ [Pet-81].

O disparo de uma transição habilitada remove todas as *marcas* dos lugares de entrada e deposita novas *marcas* nos lugares de saída, de acordo com o peso dos arcos que unem as transições aos lugares de saída. Pesos superiores a um produzem múltiplas *marcas*.

A Figura 2.4 representa uma transição habilitada para ser disparada pois $\mu(\text{FINAL DO ENVIO}) = 1 \geq 1 = \#(\text{FINAL DO ENVIO}, I(t_1))$ e $\mu(\text{FINAL DO RECEBIMENTO}) = 1 \geq 1 = \#(\text{FINAL DO RECEBIMENTO}, I(t_5))$.

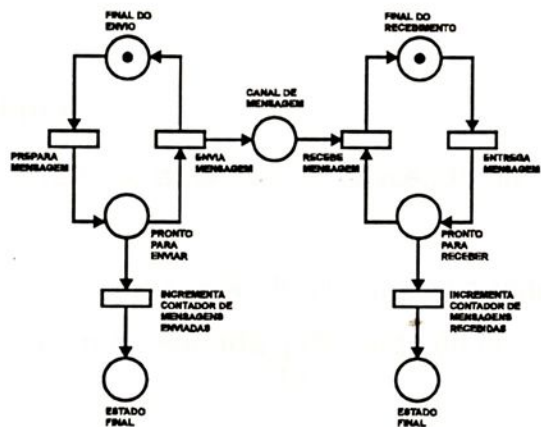


Figura 2.4 - Versão de um sistema SEND/RECEIVE com Estado Final. As transições PREPARA MENSAGEM e ENTREGA MENSAGEM estão habilitadas para serem disparadas [Rei-85].

Definição 2.13 - Geração de Novas Marcações nas redes de Petri : Uma transição $t_j \in T$ em uma rede Marcada $R = (P, T, I, O, \mu_0)$ pode disparar quando estiver habilitada. O disparo da transição habilitada t_j gera uma nova marcação μ_1 definida por

$$\mu_1(p_i) = \mu_0(p_i) - \#(p_i, I(t_j)) + \#(p_i, O(t_j))$$

onde μ_0 é a Marcação da rede, $p_i \in P$, é um lugar de entrada de t_j . O cálculo de μ_k é chamado de equação fundamental das redes de Petri, para μ_k sendo uma nova marcação gerada a partir do disparo de qualquer $t_j \in T$ [Pet-81].

A Figura 2.5 representa uma nova marcação para a estrutura da rede de Petri representada na Figura 2.4, após o disparo da transição PREPARA MENSAGEM.



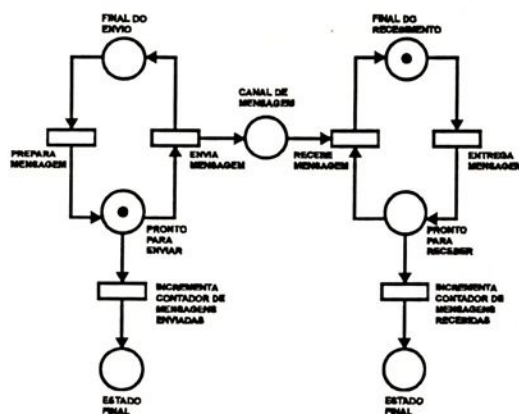


Figura 2.5 - Versão de um sistema SEND/RECEIVE com Estado Final. A transição PREPARA MENSAGEM foi disparada gerando uma nova marcação para a rede de Petri da Figura 2.4 [Rei-85].

2.4 - Rede de Petri Pura

Este trabalho será todo desenvolvido baseado na definição de redes de Petri puras.

Uma rede de Petri é pura quando um lugar pertencente a rede não for, ao mesmo tempo, um lugar de entrada e de saída de uma transição.

Definição 2.14 - Redes de Petri puras : Dada uma rede de Petri marcada $R = (P, T, I, O, \mu_0)$, onde μ_0 é a marcação inicial da rede. R é dita uma rede de Petri pura se $(p_i, I(t_j)) \cap (p_i, O(t_j)) = \emptyset$ [Pet-81].

Veja a rede de Petri da Figura 2.6(a) representando uma rede não pura, pois o lugar p_1 é um lugar de entrada e, ao mesmo tempo, um lugar de saída de t_1 . Mas a rede de Petri da Figura 2.6(b) representa uma rede de Petri pura, pois não existe nenhum lugar de saída que é também lugar de entrada de uma transição.

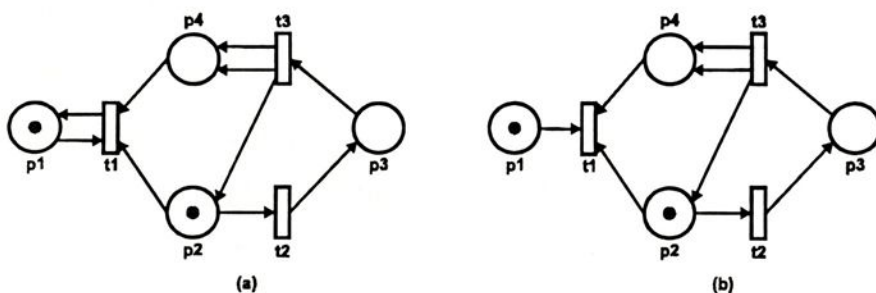


Figura 2.6 - (a) Rede de Petri não pura. (b) Rede de Petri pura.

2.5 Seqüência de Disparo de uma rede de Petri

Uma seqüência de disparos de uma rede de Petri representa a execução de uma rede, ou seja, após o disparo de uma transição habilitada, uma nova marcação é gerada, esta nova marcação poderá habilitar outras transições, o disparo destas outras transições, conseqüentemente, geram outras marcações, e assim por diante. Uma seqüência de disparos de uma rede é usada para analisar as propriedades desta rede.

Definição 2.15 - *Seqüência de Disparo de uma rede de Petri* : Dada uma rede de Petri marcada $R = (P, T, I, O, \mu_0)$, onde μ_0 é a marcação inicial da rede, se existir uma transição qualquer habilitada em R , o seu disparo gera uma marcação μ_1 , caso μ_1 habilite outras transições, o disparo de uma dessas transições gera uma outra marcação μ_2 , assim por diante. A seqüência ocorrida é uma seqüência de marcações, denotada por [Jen-94]

$$\mu_0[t_0 > \mu_1[t_1 > \mu_2[t_2 > \mu_3 \dots \mu_{n-1}[t_{n-1} > \mu_n, \text{ onde } \mu_i[t_i > \mu_{i+1}$$

significa que o disparo da transição t_j a partir da marcação μ_i conduz o modelo do sistema à marcação μ_{i+1} , para $i = 0, 1, 2, \dots, n$.

2.6 - Estado de uma rede de Petri

O estado de uma rede de Petri é definido por sua marcação. O disparo de uma transição representa a troca de estado do sistema modelado, pois ocorre a troca na marcação da rede.

Definição 2.16 - *Função Próximo Estado* : Seja $R = (P, T, I, O, \mu_0)$ uma rede de Petri marcada, com marcação inicial μ_0 e transição $t_j \in T$ habilitada, a função próximo estado $\delta : \mathbb{N}^n \times T \rightarrow \mathbb{N}^n$ é definida se e somente se

$$\mu_0(p_i) \geq \#(p_i, I(t_j))$$

para todo $p_i \in P$. Se $\delta(\mu_0, t_j)$ é definido, então $\delta(\mu_0, t_j) = \mu_1$, onde

$$\mu_1(p_i) = \mu_0(p_i) - \#(p_i, I(t_j)) + \#(p_i, O(t_j))$$



para todo $p_i \in P$. Onde N^n é o conjunto de todas as marcações, n é a cardinalidade do conjunto P [Pet-81].

A troca de estado da rede, após o disparo de uma transição, é definida pela função próximo estado. Aplicando esta função em uma marcação μ_0 , com o disparo de uma transição habilitada t_j , produz-se uma nova marcação (novo estado) μ_1 [Pet-81].

O estado inicial das redes de Petri é representado pela marcação inicial da rede. A marcação inicial das redes de Petri é representada pela presença de *marcas* nos lugares da rede, antes de iniciar a execução desta rede. Normalmente, μ_0 é a notação usada para indicar a marcação inicial de uma rede de Petri.

A seqüência de marcações $(\mu_0, \mu_1, \mu_2, \mu_3, \dots, \mu_n)$ é resultante da execução da rede de Petri após o disparo da seqüência de transições $(t_j, t_k, t_s, \dots, t_n)$.

Definição 2.17 - Marcação Imediatamente Alcançável: Seja $R = (P, T, I, O, \mu_0)$ uma rede de Petri marcada, com μ_0 a marcação inicial da rede. Uma marcação μ_1 é imediatamente alcançável de μ_0 se existir uma transição $t_j \in T$, habilitada tal que $\delta(\mu_0, t_j) = \mu_1$ [Pet-81].

As teorias citadas neste capítulo serão usadas nos próximos capítulos, onde serão definidas classes de redes de Petri, serão descritos métodos para sua análise e serão estudados vários sistemas modelados em redes de Petri, assim como será discutido o projeto desenvolvido.



CAPÍTULO III

3 - Classes das redes de Petri

As redes de Petri podem ser agrupadas em duas classes diferentes : redes de Petri Ordinárias e redes de Petri de Alto Nível. Neste capítulo serão mostradas as principais diferenças entre estas duas classes de redes de Petri.

3.1- Rede de Petri Ordinária

As redes de Petri ordinárias caracterizam-se pelo tipo de suas *marcas*. Nesta classe de rede, o tipo da *marca* pertence ao conjunto dos números inteiros não negativos $\mathbb{N}^+$.

Na classe das redes de Petri ordinária, os tipos de redes que merecem destaque são :

- Rede Condição/Evento
- Rede Lugar/Transição

as quais serão descritas a seguir, neste capítulo.

3.1.1 - Rede Condição/Evento

As redes Condição/Evento são as redes mais simples que se tem, pois os lugares destas redes podem armazenar no máximo uma *marca* e os arcos que ligam os lugares às transições são unitários.

Uma rede de Petri Condição/Evento concentra dois conceitos primitivos: eventos e condições. Os eventos são ações que são descritas pelas transições da rede. A ocorrência destes eventos é controlada pelos estados da rede. O estado de uma rede pode ser descrito como um conjunto de condições representadas pelos lugares da rede. Uma condição é um predicado ou uma descrição lógica de um estado da rede. Tais condições podem ser seguras (verdadeiras) ou não seguras (falsas) [Pet-81].



Uma vez que eventos são ações, eles podem ocorrer. Para que um evento ocorra é necessário que certas condições sejam seguras e sejam definidas como as pré-condições do evento. A ocorrência deste evento tornará suas pré-condições não seguras e gerará outras condições seguras, definidas como pós-condições do evento [Pet-81].

Definição 3.1 - *Rede Condição/Evento* : A rede de Petri $R = (C, E, A)$ é chamada rede Condição/Evento se e somente se:

- C é o conjunto de condições da rede
- E é o conjunto de eventos da rede
- C e E são conjuntos disjuntos, ou seja, $C \cap E = \emptyset$
- A representa a direção do fluxo da rede, ou seja, a ligação entre as condições e os eventos.

Graficamente, uma condição segura (verdadeira) em uma rede Condição/Evento é representada por uma *marca* no lugar que representa esta condição.

Algebricamente a marcação da rede é representada por um vetor, como foi definido no capítulo II. As condições seguras têm valor 1 enquanto as não seguras têm valor 0, para cada elemento do vetor que as representam. Assim, $\mu = (\mu_1, \mu_2, \mu_3, \dots, \mu_n)$ é a marcação da rede, onde n é a cardinalidade do conjunto C, o valor de μ_i é 0 ou 1, para cada uma das condições pertencentes a C, $i = 1, 2, \dots, n$. A notação μ_0 representa a marcação inicial da rede.

Quando todas as pré-condições de um evento são seguras pode-se realizar a troca de estado da rede. Neste caso, uma *marca* é removida das pré-condições e uma nova *marca* é adicionada às pós-condições. Isto representa a execução de uma rede Condição/Evento.

Uma rede Condição/Evento pode ser representada por um grafo, assim como por um multi-conjunto ou pela representação matricial, onde as condições representam os lugares e os eventos representam as transições da rede. Também pode-se dizer que as pré-condições representam os lugares de



entrada das transições assim como as pós-condições representam os lugares de saída das transições, como foi visto no capítulo II.

Veja o exemplo de uma máquina de venda de latas de refrigerantes usando uma rede Condição/Evento.

As condições para a rede são :

- a. A máquina está ociosa esperando para trabalhar (receber a ficha, selecionar o refrigerante e envia-lo);
- b. Uma ordem de trabalho esperada é recebida (chegada da ficha e escolha do refrigerante);
- c. A máquina esta trabalhando sobre a ordem recebida (consumindo a ficha e selecionando o refrigerante);
- d. A ordem é finalizada (refrigerante enviado).

Os eventos são :

- e₁. Chegada de uma ordem (Receber uma ficha e a escolha do refrigerante);
- e₂. A máquina inicia o trabalho sobre a ordem (consome a ficha, seleciona o refrigerante);
- e₃. A máquina finaliza a ordem (envia o refrigerante);
- e₄. A máquina é liberada.

As pré e pós-condições para cada um dos eventos dados são definidas de acordo com a tabela abaixo :

Evento	Pré-condições	Pós-condições
e ₁	nenhuma	b
e ₂	a, b	c
e ₃	c	d, a
e ₄	d	nenhuma

A representação da máquina de venda de latas de refrigerante por **multi-conjunto** é dada pela rede $R = (C, E, A)$, onde A é a ligação dos lugares às transições e das transições para os lugares. Sendo assim tem-se $I : C \rightarrow E$, representa o conjunto das pré-condições, $O : E \rightarrow C$, representa o conjunto das pós-condições, e a rede Condição/Evento é representada por :



$$R_{\text{máquina_de_vendas_de_refrigerantes}} = (C, E, I, O).$$

O conjunto das condições é representado por :

$$C = \{a, b, c, d\}$$

O conjunto dos eventos é representado por :

$$E = \{e_1, e_2, e_3, e_4\}$$

Para o conjunto das pré-condições e pós-condições tem-se :

$I(e_1) = \emptyset$	$O(e_1) = \{b\}$
$I(e_2) = \{a, b\}$	$O(e_2) = \{c\}$
$I(e_3) = \{c\}$	$O(e_3) = \{a, d\}$
$I(e_4) = \{d\}$	$O(e_4) = \emptyset$

Sua marcação inicial é $\mu_0 = (1, 0, 0, 0)$.

A representação da máquina de venda de latas de refrigerante por **grafo** é dada pela rede

$$R_{\text{máquina_de_vendas_de_refrigerantes}} = (C, E, A).$$

O conjunto das condições é representado por :

$$C = \{a, b, c, d\}$$

O conjunto dos eventos é representado por :

$$E = \{e_1, e_2, e_3, e_4\}$$

A relação existente entre as condições e os eventos é dada por

$$A = \{(e_1, b), (b, e_2), (a, e_2), (e_2, c), (c, e_3), (e_3, d), (e_3, a), (d, e_4)\}$$

Sua marcação inicial é $\mu_0 = (1, 0, 0, 0)$.

A Figura 3.1 representa graficamente uma máquina de vendas de latas de refrigerantes baseado no modelo descrito acima :

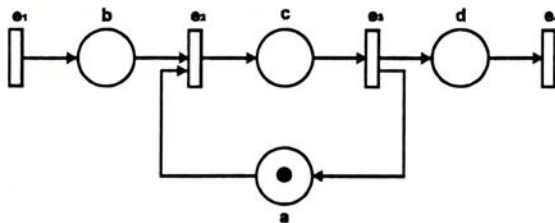


Figura 3.1 - Representação gráfica de uma máquina de vendas de latas de refrigerantes usando uma rede de Petri Condição/Evento [Pet-81].

Na representação **matricial** da máquina de venda de latas de refrigerante, I é a matriz de entrada, O a matriz de saída e a rede é

$$R_{\text{máquina_de_vendas_de_refrigerantes}} = (C, E, I, O).$$

O conjunto das condições é representado por :

$$C = \{a, b, c, d\}$$

O conjunto dos eventos é representado por :

$$E = \{e_1, e_2, e_3, e_4\}$$

A matriz de entrada da rede é dada por :

$$I = \begin{bmatrix} a & b & c & d \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{matrix} e_1 \\ e_2 \\ e_3 \\ e_4 \end{matrix}$$

A matriz de saída da rede é dada por :

$$O = \begin{bmatrix} a & b & c & d \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{matrix} e_1 \\ e_2 \\ e_3 \\ e_4 \end{matrix}$$

Sua marcação inicial é $\mu_0 = (1, 0, 0, 0)$.

3.1.2 - Rede Lugar/Transição

As redes Lugar/Transição, em contraste com as redes Condição/Evento, podem acumular *marcas* nos lugares assim como os seus arcos podem ter pesos maiores ou iguais a 1. Um estado atual da rede Lugar/Transição é representado pela distribuição de *marcas* nos lugares da rede. O número de *marcas* em cada um dos lugares pode ser maior ou igual a zero.

Os lugares de entrada de uma transição são todos os lugares cujos arcos têm origem no lugar e destino na transição. O conjunto de todos os lugares de entrada de uma transição é chamado de pré-condição da transição e a notação usada é $\cdot t$. Assim como os lugares de saída de uma transição são aqueles cujos arcos têm origem na transição e destino no lugar. O conjunto de todos os lugares de saída de uma transição é chamado pós-condição de uma transição e a notação usada é t^* .

O disparo de uma transição corresponde à remoção das *marcas* dos lugares de entrada desta transição e à criação de novas *marcas* nos lugares de saída da transição.

A execução de uma rede de Petri Lugar/Transição é representada pela mudança de estado da rede, sempre que uma transição é disparada. Após o disparo de várias transições tem-se uma seqüência de marcações da rede.

Uma transição t_j está habilitada para ser disparada se cada lugar pertencente a *t_j têm o número de *marcas* maior ou igual ao peso do arco que o conecta a t_j . Quando t_j é disparada, as *marcas* presentes nos lugares de entrada de t_j são removidas e novas *marcas* são adicionadas em todos os lugares pertencentes a t_j^* , de acordo com os pesos dos arcos.

Definição 3.2 - *Rede Lugar/Transição* : Seja $R = (P, T, A, \mu)$ uma rede é chamada rede de Petri Lugar/Transição se e somente se :

- P é o conjunto de lugares da rede
- T é o conjunto de transições da rede
- P e T são conjuntos disjuntos, ou seja, $P \cap T = \emptyset$
- A representa a direção do fluxo da rede, ou seja, a ligação entre os lugares e as transições
- μ representa a marcação da rede [Rei-85].

Em uma rede Lugar/Transição os arcos são rotulados com o seu peso. O peso de um arco representa múltiplos arcos ligando um lugar a uma transição, assim como uma transição a um lugar.

Definição 3.3 - *Regras de disparo para uma rede Lugar/Transição* : Seja $R = (P, T, A, \mu)$ uma rede Lugar/Transição então :

- μ é uma marcação da rede R se e somente se para todo lugar $p_i \in P$, $\mu(p_i) \geq 0$



- Uma transição $t_j \in T$ está habilitada para um disparo se para todo lugar $p_i \in {}^*t_j$, $\mu(p_i)$ for maior ou igual ao peso dos arcos que ligam p_i a t_j
- Uma nova marcação é gerada pela remoção das *marcas* dos lugares $p_i \in {}^*t_j$, e pela adição de novas *marcas* em todos os lugares $p_i \in t_j^*$, de acordo com o peso dos arcos que os ligam [Rei-85].

Pode-se representar uma rede de Petri Lugar/Transição usando a representação por multi-conjunto, assim como a representação por grafo e a representação matricial.

Veja o exemplo de um processo Leitor/Escritor de um Sistema Operacional usando uma rede Lugar/Transição cuja representação será dada por multi-conjunto, por grafo e matricialmente para que se possa entender melhor o funcionamento do sistema.

As definições das pré e pós-condições para cada uma das transições são dadas pela tabela abaixo :

transição	pré-condição	pós-condição
t_1	p_1 : processo inativo	p_2 : processo pronto para ler
t_2	p_2 : processo pronto para ler p_6 : sincronização	p_3 : processo lendo
t_3	p_3 : processo lendo	p_1 : processo inativo p_6 : sincronização
t_4	p_1 : processo inativo	p_4 : processo pronto para escrever
t_5	p_4 : processo pronto para escrever p_6 : sincronização	p_5 : processo escrevendo
t_6	p_5 : processo escrevendo	p_1 : processo inativo p_6 : sincronização

A representação por **multi-conjunto** é dada por :

A rede de Petri $R = (P, T, I, O, \mu)$, onde $I : P \rightarrow T$, representa o conjunto das pré-condições e $O : T \rightarrow P$, representa o conjunto das pós-condições. μ é a marcação inicial da rede. . Sendo assim tem-se a rede

$$R_{\text{processo_Leitor/Escritor}} = (P, T, I, O, \mu)$$



O conjunto de lugares é representado por :

$$P = \{p_1, p_2, p_3, p_4, p_5, p_6\}$$

O conjunto de transições é representado por :

$$T = \{t_1, t_2, t_3, t_4, t_5, t_6\}$$

Para o conjunto das pré-condições e pós-condições tem-se :

$$I(t_1) = \{p_1\}$$

$$O(t_1) = \{p_2\}$$

$$I(t_2) = \{p_2, p_6\}$$

$$O(t_2) = \{p_3\}$$

$$I(t_3) = \{p_3\}$$

$$O(t_3) = \{p_1, p_6\}$$

$$I(t_4) = \{p_1\}$$

$$O(t_4) = \{p_4\}$$

$$I(t_5) = \{p_4, 5p_6\}$$

$$O(t_5) = \{p_5\}$$

$$I(t_6) = \{p_5\}$$

$$O(t_6) = \{p_1, 5p_6\}$$

Sua marcação inicial é $\mu_0 = (1, 0, 0, 0, 0, 0)$.

A representação de um processo Leitor/Escritor usando **grafo** de um Sistema Operacional é dada pela rede de Petri Lugar/Transição

$$R_{\text{processo_Leitor/Escritor}} = (P, T, A, \mu).$$

O conjunto de lugares é representado por :

$$P = \{p_1, p_2, p_3, p_4, p_5, p_6\}$$

O conjunto de transições é representado por :

$$T = \{t_1, t_2, t_3, t_4, t_5, t_6\}$$

A relação existente entre as condições e os eventos é dada por

$$A = \{(p_1, t_1), (t_1, p_2), (p_2, t_2), (t_2, p_3), (p_3, t_3), (t_3, p_1), (t_3, p_6), (p_6, t_2), (5p_6, t_5), (p_1, t_4), (t_4, p_4), (p_4, t_5), (t_5, p_5), (p_5, t_6), (t_6, p_1), (t_6, 5p_6)\}$$

A marcação inicial é $\mu_0 = (1, 0, 0, 0, 0, 0)$

A Figura 3.2 representa graficamente o funcionamento de um processo Leitor/Escritor de um Sistema Operacional modelado usando uma rede de Petri Lugar/Transição, baseado no sistema definido acima :



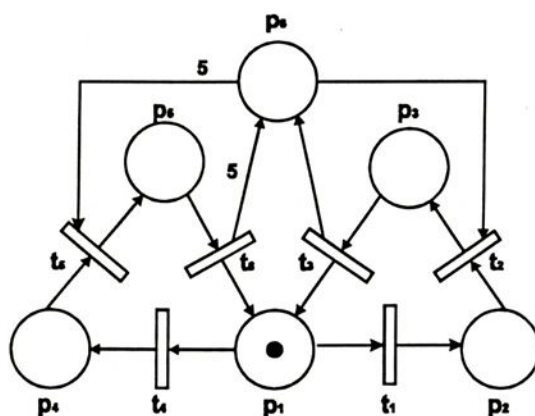


Figura 3.2 - Um sistema de processo Leitor/Escritor de um Sistema Operacional modelado por rede de Petri Lugar/Transição [Rei-85].

Na representação **matricial** de um processo Leitor/Escritor de um Sistema Operacional, onde I é a matriz de entrada e O a matriz de saída, a rede de Petri Lugar/Transição é dada por

$$R_{\text{processo_Leitor/Escritor}} = (P, T, I, O).$$

O conjunto de lugares é representado por :

$$P = \{p_1, p_2, p_3, p_4, p_5, p_6\}$$

O conjunto de transições é representado por :

$$T = \{t_1, t_2, t_3, t_4, t_5, t_6\}$$

A matriz de entrada da rede é dada por :

$$I = \begin{bmatrix} p_1 & p_2 & p_3 & p_4 & p_5 & p_6 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 5 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \begin{matrix} t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \\ t_6 \end{matrix}$$

A matriz de saída da rede é dada por :

$$O = \begin{bmatrix} p_1 & p_2 & p_3 & p_4 & p_5 & p_6 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 5 \end{bmatrix} \begin{matrix} t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \\ t_6 \end{matrix}$$

Sua marcação inicial é $\mu_0 = (1, 0, 0, 0, 0, 0)$.

Embora as redes de Petri Ordinárias sejam uma técnica de descrição formal poderosa que possibilita a modelagem de sistemas concorrentes de forma natural, a sua aplicação para sistemas reais tem se mostrado inadequados, pois uma grande dificuldade que se encontra é a falta de expressividade das redes de Petri Ordinárias na tradução do problema ao modelo das redes de Petri. Como foi mostrado nos exemplos dados até aqui, nas redes de Petri Ordinárias existe apenas um tipo de *marca*, o que não permite a diferenciação de recursos em um lugar, sendo necessário lugares distintos para expressar recursos similares. As redes Ordinárias não possibilitam ainda níveis de refinamento do modelo, ou seja, não há hierarquização. Sendo assim, um grande número de extensões às redes de Petri, são denominadas redes de Petri de Alto Nível [Mac-96].

3.2 - Redes de Petri de Alto Nível

Modelar redes em larga escala torna-se impossível sem o conceito e a notação oferecida pelas redes de alto nível. Uma das características das redes de alto nível é que suas *marcas* representam estruturas de dados e não meramente valores lógicos ou inteiros positivos, como nas redes Ordinárias. O uso de *marcas*, que são estruturas de dados, permite a representação resumida das redes de sistemas complexos. Durante muitos anos uma variedade de redes de alto nível tem sido propostas e colocadas em uso. Entre os diversos modelos em uso atualmente, um modelo que se destaca tanto na teoria como na aplicação é o das Redes de Petri Coloridas, as quais serão discutidas a seguir.

3.2.1 - Redes de Petri Coloridas

O principal objetivo das redes de Petri Coloridas é a redução do tamanho do modelo descrito pelas redes Ordinárias, permitindo que *marcas* individualizadas (cores) representem diferentes processos ou recursos em uma mesma rede. Nesta classe de rede os arcos são rotulados com uma ou mais das possíveis cores ou ainda com variáveis. Para que uma transição desta



rede esteja habilitada é necessário que os lugares de entrada desta transição tenham *marcas* do tipo (cor) associado aos arcos que interligam estes lugares à transição. O disparo desta transição retira as *marcas* das cores associadas aos arcos, dos lugares de entrada e adiciona novas *marcas* aos lugares de saída, estas são *marcas* das cores iguais às cores associadas aos arcos que interligam a transição aos lugares de saída [Mac-96].

Para que isto possa ficar mais claro veja a Figura 3.3 que representa uma rede de Petri Colorida. Na Figura 3.3(a), a transição t_1 não pode disparar pois o lugar de entrada p_1 não possui uma cor a (vermelha) rotulada no arco que liga o lugar de entrada p_1 à transição t_1 , já a transição t_2 pode disparar, pois os lugares de entrada p_2 e p_3 possuem as cores inscritas nos arcos que os ligam à transição t_2 , respectivamente. A Figura 3.3(b) representa o disparo da transição t_2 para as cores nos arcos que ligam os lugares de entrada a t_2 .

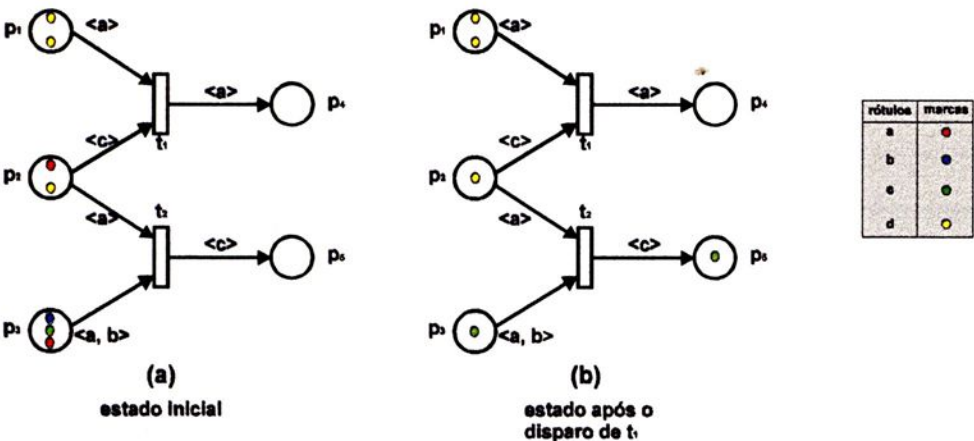


Figura 3.3 - Rede de Petri Colorida [Mac-96].

Maiores detalhes sobre as rede de Petri Coloridas serão dados no capítulo VI onde este assunto será discutido mais detalhadamente.

Por enquanto, é necessário que se saiba apenas os conceitos básicos, para identificar as principais diferenças entre as redes Ordinárias e as redes de alto nível.

Como dito anteriormente, as redes Ordinárias armazenam apenas um tipo simples de *marcas* e, como consequência disto, a modelagem de sistemas complexos torna-se difícil. No caso das redes de alto nível esta modelagem

pode ser feita com mais facilidade. Sendo assim, torna-se vantajoso o uso das redes de Petri de alto nível para a modelagem de sistemas complexos.



CAPÍTULO IV

4 - Propriedades das redes de Petri

As redes de Petri não se restringem apenas a uma ferramenta que possibilita a modelagem de sistemas. Foi desenvolvido, em paralelo, uma série de métodos que permitem a análise de um grande número de propriedades dos sistemas modelados. As propriedades das redes de Petri podem ser divididas em dois grandes grupos : propriedades que dependem da marcação, denominadas propriedades comportamentais, e as propriedades que não dependem da marcação, denominadas propriedades estruturais, estas dependem apenas da topologia da rede [Mac-96].

4.1 - Propriedades Comportamentais

Estas propriedades conduzem para respostas sobre o comportamento do sistema modelado usando redes de Petri. Várias técnicas tem sido desenvolvidas para analisar as redes de Petri, mas muitos problemas ainda encontram-se em aberto. O objetivo da análise das redes de Petri é determinar as características do sistema modelado.

4.1.1 - Segurança (*Safeness*)

Uma das mais importantes propriedades é a segurança. Um lugar em uma rede de Petri é seguro se o número de marcas neste lugar nunca excede a uma única marca. Uma rede de Petri é segura se todos os seus lugares são seguros [Pet-81].

Definição 4.1 - Lugar seguro : Seja $p_i \in P$ um lugar qualquer de uma rede de Petri $R = (P, T, I, O, \mu)$, onde μ é a marcação inicial da rede R . p_i é um lugar seguro se para toda marcação $\mu_i \in R$, $\mu_i(p_i) \leq 1$ [Mac-96].



Definição 4.2 - Rede de Petri segura : Seja $R = (P, T, I, O, \mu)$ uma rede de Petri, onde μ é a marcação inicial da rede. $R = (P, T, I, O, \mu)$ é uma rede segura se para toda marcação $\mu_i \in R$ e para todo $p_i \in P$ tem-se $\mu_i(p_i) \leq 1$. Portanto uma rede de Petri é segura se todos os lugares $p_i \in P$ são seguros [Pet-81].

A Figura 4.1 representa uma rede segura, pois todos os seus lugares não acumulam mais que uma marca.

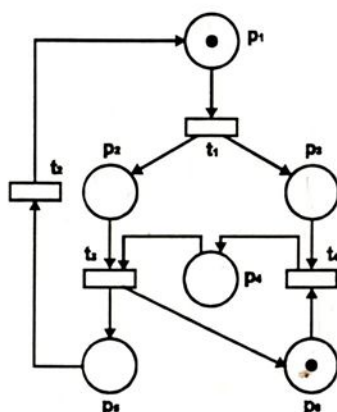


Figura 4.1 - Rede de Petri Segura (Safe) [Mac-96].

As redes de Petri Condição/Evento são seguras por definição, uma vez que todos os seus lugares só podem armazenar uma marca, este foi o motivo pelo qual os lugares desta rede, são representados como uma condição, pois uma condição é verdadeira ou falsa [Pet-81].

4.1.2 - Limitação (*Boundedness*)

Um lugar em uma rede de Petri é limitado quando a sua marcação não exceder k , onde k é um número inteiro positivo. Uma rede de Petri é limitada quando todos os seus lugares são limitados.

Definição 4.3 - Lugar limitado : Seja $R = (P, T, I, O, \mu)$ uma rede de Petri com marcação inicial μ . Seja $p_i \in P$ um lugar qualquer da rede R . p_i é dito um lugar limitado se a quantidade de marcas em p_i não exceder k , onde $k \in \mathbb{N}^+$ é um

número inteiro positivo, ou seja, $\mu_i(p_i) \leq k$ para $\mu_i \in R$. p_i é dito k -limitado [Pet-81].

Definição 4.4 - Lugar k -seguro : Seja $R = (P, T, I, O, \mu)$ uma rede de Petri com marcação inicial μ . Seja $p_i \in P$ um lugar qualquer da rede R . p_i é dito um lugar k -seguro se p_i é um lugar k -limitado para $k \in \mathbb{N}^+$ [Pet-81].

Definição 4.5 - Rede de Petri limitada : Seja $R = (P, T, I, O, \mu)$ uma rede de Petri com marcação inicial μ . Seja $p_i \in P$ um lugar qualquer da rede R . R é uma rede limitada se $\mu_i(p_i) \leq k$ para todo $\mu_i \in R$, onde $k \in \mathbb{N}^+$ é um número inteiro positivo. R é dita uma rede k -limitada. Portanto uma rede é limitada se todos os seus lugares são limitados [Pet-81].

A Figura 4.2 representa uma rede de Petri limitada pois o lugar p_8 permite que o lugar p_3 acumule, no máximo, 4 marcas. Após uma breve análise desta rede pode-se concluir que os lugares p_1, p_2, p_4, p_5, p_6 e p_7 podem armazenar 0 ou 1 marca, estes lugares são lugares seguros e também, são 1-limitados, enquanto os lugares p_3 e p_8 podem armazenar entre 0 e 4 marcas e são 4-limitados. Sendo assim, esta rede é dita 4-limitada e todos os seus lugares são limitados.

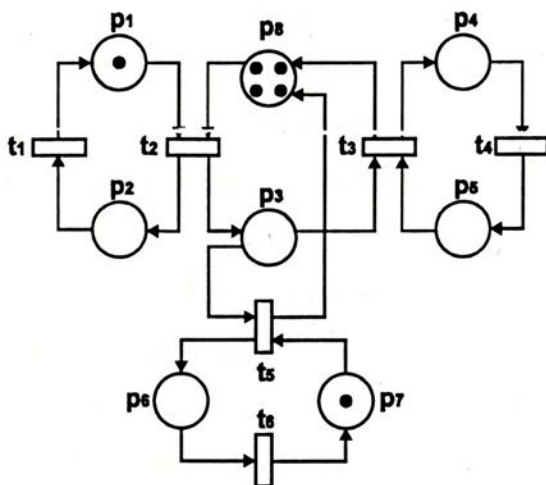


Figura 4.2 - Rede de Petri limitada. O número de marcas em qualquer lugar é limitado [Gom-96].

As redes seguras são casos especiais de redes limitadas.



Um lugar k-seguro é chamado simplesmente de seguro.

4.1.3 - Vivacidade (*liveness*)

Um dos problemas mais estudados na ciência da computação tem sido o *deadlock*. Para as redes de Petri, a ausência de *deadlock* está fortemente ligada à vivacidade. *Deadlock* para uma rede de Petri é a impossibilidade do disparo de qualquer transição da rede. Um *deadlock* em uma rede de Petri ocorre quando uma transição nunca é disparada. Uma transição é viva se ela não está em *deadlock*. Uma rede de Petri é viva se todas as suas transições não estão em *deadlock* [Pet-81].

Definição 4.6 - *Transição potencialmente disparável* : Seja $R = (P, T, I, O, \mu)$ uma rede de Petri, onde μ é sua marcação inicial. Seja $t_j \in T$ uma transição qualquer, t_j é dita potencialmente disparável em uma marcação μ se existir uma marcação $\mu_i \in R$, onde t_j é habilitada para esta marcação μ_i [Mac-96].

Definição 4.7 - *Transição viva* : Seja $R = (P, T, I, O, \mu)$ uma rede de Petri, onde μ é a marcação da inicial. Seja $t_j \in T$ uma transição qualquer, t_j é dita viva em uma marcação μ se t_j é potencialmente disparável em todas as marcações $\mu \in R$, ou seja, t_j é viva se esta não é passível de *deadlock* [Mac-96].

Definição 4.8 - *Rede de Petri viva* : Seja $R = (P, T, I, O, \mu)$ uma rede de Petri, onde μ é a marcação inicial. Seja $t_j \in T$ uma transição qualquer, R é uma rede de Petri viva se toda transição t_j é uma transição viva [Mac-96].

A propriedade de vivacidade é fundamental na análise de sistemas reais, mas sua observação é cara quando se consideram sistemas de grande porte. Estudos resultaram na estratificação deste conceito, caracterizando-os em níveis de vivacidade [Pet-81]. Tais níveis são definidos a seguir, considerando-se uma rede de Petri $R = (P, T, I, O, \mu_0)$, com marcação inicial μ_0 e sendo $t_j \in T$ uma transição qualquer :



- Nível 0 : Uma transição t_j é viva para o nível 0 se ela nunca pode ser disparada.
- Nível 1 : Uma transição t_j é viva para o nível 1 se ela é potencialmente disparável, isto é, se existir uma marcação $\mu_i \in R$, tal que t_j é habilitada em μ_i .
- Nível 2 : Uma transição t_j é viva para o nível 2 se para todo inteiro não negativo n existe uma seqüência de disparos na qual t_j ocorre pelo menos n vezes.
- Nível 3 : Uma transição t_j é viva para o nível 3 se em uma seqüência infinita de disparos, t_j pode ser disparada um número infinito de vezes.
- Nível 4 : Uma transição t_j é viva para o nível 4 se para toda marcação $\mu_i \in R$, existe uma seqüência de disparos na qual t_j está habilitado [Pet-81].

Uma transição viva para o nível 0 é dita morta. Uma transição viva para o nível 4 é simplesmente viva. Uma rede de Petri é viva para qualquer nível i se toda transição é viva para o nível i .

Veja a Figura 4.3 onde estão representadas uma rede de Petri com *deadlock* (a), pois se forem disparadas as transições t_1 e em seguida t_4 não tem-se marcas suficientes para disparar qualquer outra transição, e uma rede de Petri viva (b), pois não existe uma seqüência de disparo onde a rede entre em *deadlock*.

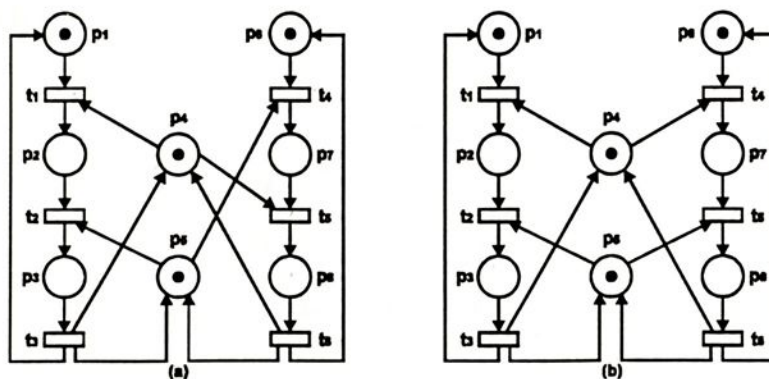


Figura 4.3 - (a) Rede com *deadlock*, não Viva [Pet-81]. (b) Rede Viva [Mac-96].

4.1.4 - Alcançabilidade (*Reachability*)

Alcançabilidade é fundamental para os estudos de propriedades dinâmicas de qualquer sistema. Para as redes de Petri, a alcançabilidade indica a possibilidade de uma certa marcação ser atingida pelo disparo de um número finito de transições, a partir da marcação inicial. Sendo assim, uma marcação μ_i é dita acessível a partir da marcação inicial μ_0 , se existir uma seqüência de transições que, disparadas, levam à marcação μ_i . O problema da alcançabilidade pode ser caracterizado como o problema de saber se uma marcação qualquer μ_i pertence a uma rede de Petri [Gom-96].

Definição 4.9 - Marcação Alcançável : Seja $R = (P, T, I, O, \mu_0)$ uma rede de Petri, onde μ_0 é a marcação inicial. Seja $s = \mu_0 [t_{j_0} > \mu_1 [t_{j_1} > \mu_2 [t_{j_2} > \dots \mu_i [t_{j_i} > \dots \mu_{n-1} [t_{j_{n-1}} > \mu_n$ uma seqüência de marcações de R , ou seja, $s \in R$. μ_k é uma marcação alcançável em R , a partir de μ_0 , se $\mu_k \in R$ [Mac-96].

Definição 4.10 - Alcançabilidade : Seja $R = (P, T, I, O, \mu_0)$ uma rede de Petri, onde μ_0 é a marcação inicial. Seja $s = \mu_0 [t_{j_0} > \mu_1 [t_{j_1} > \mu_2 [t_{j_2} > \dots \mu_i [t_{j_i} > \dots \mu_{n-1} [t_{j_{n-1}} > \mu_n$ uma seqüência de marcações de R , ou seja, $s \in R$. O conjunto de todas as marcações possíveis obtidas a partir da marcação inicial μ_0 na rede R é denotado por $A(R) = \{\mu_k \in \mathbb{N}^m \mid \exists s \in R \mid \mu_0[s > \mu_k\}$ [Mac-96].

Veja a Figura 4.4, cuja marcação inicial é $\mu_0 = (1, 0, 0, 0)$, por exemplo a marcação $\mu_k = (0, 0, 0, 1)$ é uma marcação alcançável, a partir de μ_0 , pois pode ser atingida pelos disparos das transições t_1, t_2 e t_4 ou então pelos disparos das transições t_3 e t_4 .

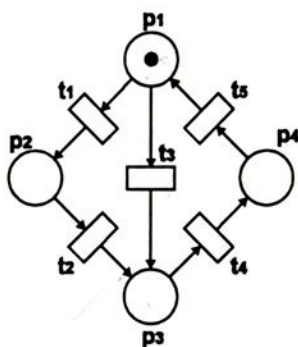


Figura 4.4 - Rede de Petri com marcação inicial $\mu_0 = (1, 0, 0, 0)$ e marcação alcançável $\mu_k = (0, 0, 0, 1)$ [Mac-96].

Muitos problemas podem ser analisados em termos do problema de alcançabilidade. Por exemplo pode-se saber se uma determinada transição esta em *deadlock* ou não, a partir de uma certa marcação.

4.1.5 - Cobertura (*Coverability*)

A propriedade de cobertura está fortemente ligada às propriedades de alcançabilidade e vivacidade. Quando deseja-se saber se uma marcação qualquer μ_i é obtida a partir de uma determinada marcação μ_k , tem-se um problema denominado cobertura de uma marcação μ_k .

Definição 4.11 - *Cobertura de uma marcação* : Seja $R = (P, T, I, O, \mu_0)$ uma rede de Petri, onde μ_0 é sua marcação inicial e μ_i é uma marcação qualquer, se existir uma marcação alcançável $\mu_k \in R$, tal que $\mu_k \geq \mu_i$ então μ_i é dita coberta por μ_k . Assim $\mu_k(p_j) \geq \mu_i(p_j)$, $\forall p_j \in P$ [Mac-96].

4.1.6 - Reiniciabilidade (*Reversibility*)

Uma rede de Petri é definida como reiniciável se para cada marcação alcançável, no conjunto de marcações acessíveis da rede, a marcação inicial pode ser atingida novamente. Para algumas aplicações é suficiente garantir o retorno a alguma marcação, diferente da marcação inicial, denominado marcação *home*, pelo que o conceito de reiniciabilidade é convertido na marcação *home*.

Definição 4.12 - *Home-State* : Seja $R = (P, T, I, O, \mu)$ uma rede de Petri, onde μ é a marcação inicial. Seja uma marcação $\mu_i \in R$. μ_i é denominada *home-state* se μ_i é uma marcação alcançável a partir de uma seqüência de marcações μ_k , para toda $\mu_k \in R$ [Mac-96].



Definição 4.13 - Rede de Petri reiniciável: Seja $R = (P, T, I, O, \mu)$ uma rede de Petri, onde μ é a marcação inicial. Se existir uma marcação qualquer $\mu_i \in R$, onde $\mu_i = \mu$, a rede é dita reiniciável [Mac-96].

Veja a Figura 4.5, em que estão representadas duas redes de Petri. A Figura 4.5(a) apresenta uma rede reiniciável, pois se as de transições t_1 , t_2 e t_4 forem disparadas nesta ordem, a marcação alcançada será igual à marcação inicial. A rede mostrada na Figura 4.5(b) não é reiniciável pois não é possível atingir novamente a marcação inicial e não existe uma marcação *home*.

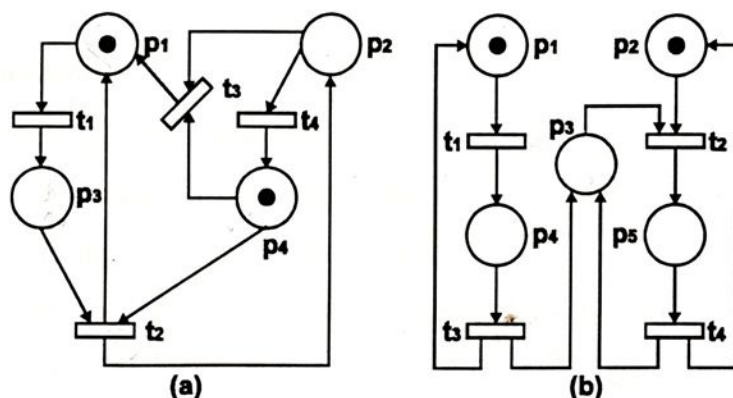


Figura 4.5 - (a) Rede de Petri reiniciável. (b) Rede de Petri não reiniciável, e sem marcação *home* [Mac-96].

Observe ainda que a rede da Figura 4.5(b) não é limitada, pois o lugar p_3 acumula marcas se as transições t_1 e t_3 forem disparadas várias vezes.

4.2 - Propriedades Estruturais

Muitas vezes, o interesse está em observar as propriedades que estão apenas relacionadas com a estrutura do modelo. As propriedades estruturais são aquelas que refletem características do sistema modelado, independentes da marcação da rede de Petri que representa o sistema. Uma vez que as redes sejam puras pode-se representá-las pela sua matriz de incidência.

Para definir as propriedades estruturais serão consideradas apenas redes de Petri puras, nesta seção.

4.2.1 Limitação Estrutural

Uma rede de Petri é estruturalmente limitada (*structural bounded*) se é limitada para qualquer marcação inicial.

Definição 4.14 - *Rede de Petri estruturalmente limitada* : Seja $R = (P, T, I, O, \mu)$ uma rede de Petri, onde μ é a marcação inicial. Seja n a cardinalidade de P . Seja D a matriz de incidência de R . R é dita estruturalmente limitada se existir um vetor W de números inteiros com dimensão n tal que $W.D \leq 0$ [Mac-96].

4.2.2 - Conservação

A conservação é uma importante propriedade das redes de Petri permitindo, por exemplo, a verificação da não destruição de recursos através da simples conservação das marcas na rede.

Uma rede de Petri é dita conservativa quando ocorre o disparo de qualquer transição habilitada e a troca da marcação não altera o número de marcas na rede.

Definição 4.15 - *Rede de Petri Estritamente Conservativa* : Seja $R = (P, T, I, O, \mu)$ uma rede de Petri, onde μ é a marcação inicial. R é dita estritamente conservativa se $\sum \mu_j(p_i) = \sum \mu_0(p_i), \forall p_i \in P, \forall \mu_j \in R$ [Pet-81].

Para que a propriedade da **definição 4.15** seja observada é necessário que o número de arcos que ligam os lugares de entrada à transição seja igual ao número de arcos que ligam à transição aos lugares de saída, ou seja, a cardinalidade das pré-condições deve ser igual à das pós-condições de uma transição t_j , $\#I(t_j) = \#O(t_j)$. Assim quando t_j for disparada, o número de marcas na rede será conservado.

A Figura 4.6(a) representa uma rede que não é estritamente conservativa, pois se a transição t_1 for disparada, uma marca é retirada de p_1 e duas marcas são retiradas de p_5 , mas apenas uma marca é adicionada em



p_2 , não conservando o número de marcas na rede. Mas a Figura 4.6(b) é uma rede estritamente conservativa e foi transformada, a partir da Figura 4.6(a), para isto foi necessário alterar o número de arcos de saída das transições t_1 e t_2 para 3 e 2, respectivamente, assim como alterar os arcos de entrada das transições t_3 e t_4 para 3 e 2, respectivamente. Isto só é possível graças a restrição $\#I(t_i) = \#O(t_i)$ [Mac-96].

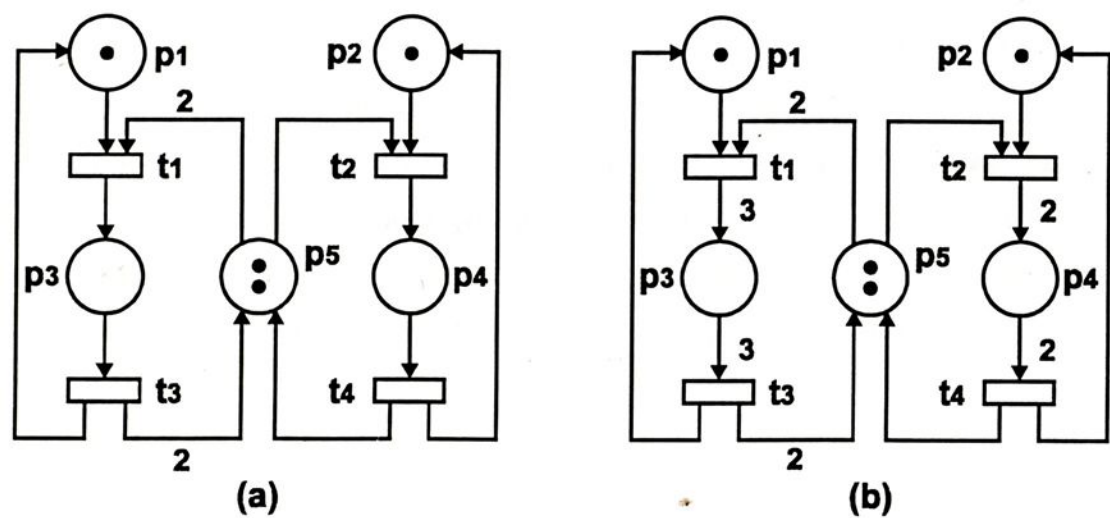


Figura 4.6 - (a) Representação de uma rede de Petri que não é estritamente conservativa. (b) Transformação da rede de Petri em uma rede estritamente conservativa [Mac-96].

A conservação nas redes de Petri não se restringe apenas a manter o somatório de marcas constantes. Depende da estrutura da rede que permite transformá-la de uma rede não conservativa para uma rede estritamente conservativa. Redes com esta característica, ou seja, que podem ser transformadas em estritamente conservativas são denominadas redes conservativas. Uma marca em um lugar pode ser usada para criar novas marcas em outros lugares após o disparo de uma transição. Mas para que possa valer a propriedade conservativa é necessário atribuir a cada lugar da rede um peso. Assim, tem-se um vetor peso $w = (w_1, w_2, w_3, ..., w_n)$ associado a cada um dos lugares da rede e define-se w_i como sendo o peso do lugar $p_i \in P$, onde n é a cardinalidade de P . Para as marcas que não são importantes é atribuído o peso 0, para as outras marcas é atribuído o peso 1, 2, 3, ..., ou qualquer outro valor inteiro positivo.

Definição 4.16 - *Rede de Petri conservativa* : Seja $R = (P, T, I, O, \mu)$ uma rede de Petri, onde μ é a marcação inicial. R é dita conservativa com relação ao vetor peso w , $w = (w_1, w_2, w_3, \dots, w_n)$, n a cardinalidade de P , $w_i > 0$, se $\forall \mu_j \in R$, $\sum w_i * \mu_j(p_i) = \sum w_i * \mu_0(p_i)$, $\forall p_i \in P$ [Pet-81].

Uma rede de Petri estritamente conservativa é conservativa com relação ao vetor peso $w = (1, 1, 1, \dots, 1)$. Observe que na definição 4.16 w_i é estritamente positivo, pois se fosse permitido usar o vetor nulo, como um vetor peso, para esta definição todas as redes de Petri seriam conservativas. Isto seria uma perturbação à definição de redes conservativas, pois uma rede só é conservativa em relação a algum vetor peso. Portanto, uma rede de Petri é conservativa se ela é conservativa com relação a algum vetor peso não nulo, $w > 0$ [Pet-81].



CAPÍTULO V

5 - Análise das redes de Petri

Ao executar uma rede de Petri, a partir de sua marcação inicial tem-se um conjunto de transições habilitadas. Disparando pelo menos uma transição, tem-se uma nova marcação e provavelmente novas transições habilitadas. Este processo pode se repetir infinitas vezes, para uma rede viva. Algumas das técnicas de análise de redes de Petri, usadas para verificar as suas propriedades, serão apresentadas neste capítulo.

5.1 - Métodos de análise das redes de Petri

Os métodos de análise estrutural das redes de Petri baseiam-se na representação matricial destas redes, a qual permite executá-las e determinar uma marcação qualquer alcançável a partir da marcação inicial. Essas técnicas se dividem em dois importantes grupos :

- Baseadas na árvore de alcançabilidade
- Baseadas na equação fundamental das redes de Petri.

5.1.1 - Árvore de alcançabilidade

A árvore de alcançabilidade representa um conjunto de marcações alcançáveis das redes de Petri, a partir de sua marcação inicial. A raiz de uma árvore de alcançabilidade é rotulada pela marcação inicial da rede. Os demais nós são rotulados pelas marcações alcançáveis. Os arcos são rotulados pela transição (ou seqüência de transições) cujo disparo gera a nova marcação .

Pensar em uma árvore de alcançabilidade finita é quase impossível, diante da possibilidade de se ter um conjunto infinito de marcações alcançáveis. Uma vez que, a rede de Petri pode não ser reiniciável. Sendo uma rede de Petri reiniciável, tem-se a possibilidade de uma repetição de marcações (ciclo), considerando todos os caminhos de disparo possíveis. É



bom lembrar que uma rede de Petri ilimitada também gera uma árvore de alcançabilidade infinita, pois a cada disparo uma nova marcação é gerada, diferente da anterior, pois ocorre o acúmulo de marcas em um determinado lugar.

A árvore de alcançabilidade é utilizada para representar graficamente um número infinito de marcações, de maneira finita. Uma forma para transformar uma árvore de alcançabilidade infinita em uma árvore de alcançabilidade finita é limitar o número de marcações na árvore, a cada nova marcação, atribuindo uma variável ω em cada marcação $\mu(p_i)$ que acumula marcas, no decorrer da construção da árvore, para $p_i \in P$, $\omega \in \mathbf{N}^+$ uma variável. Isto provoca um corte na árvore evitando que ela se torne infinita [Pet-81]. Veja um exemplo de árvore de alcançabilidade na Figura 5.2.

Definição 5.1 - *Árvore de alcançabilidade ou de cobertura ou de marcação*: Seja $R = (P, T, I, O, \mu_0)$ uma rede de Petri, onde μ_0 é a marcação inicial. Define-se um grafo $AC = (V, A)$ como sendo uma árvore de alcançabilidade, onde V é o multi-conjunto dos nós do grafo e A é multi-conjunto dos arcos que ligam os nós $v_i \in V$, $i = 1, 2, 3, \dots$. V é rotulado pelo conjunto de marcações alcançáveis, a partir de μ_0 e A é rotulado por $t_j \in T$, onde t_j é uma transição qualquer, potencialmente disparável [Rei-85].

Definição 5.2 - *Nó terminal*: Seja $R = (P, T, I, O, \mu_0)$ uma rede de Petri, onde μ_0 é a marcação inicial. Seja $AC = (V, A)$ uma árvore de alcançabilidade de R . Seja $v_i \in V$, $i = 1, 2, 3, \dots$, uma marcação alcançável de R , v_i é dito nó terminal se v_i é uma marcação que não habilita nenhuma transição $t_j \in T$, ou seja, t_j não é potencialmente disparável para v_i [Rei-85].

Definição 5.3 - *Nó duplicado*: Seja $R = (P, T, I, O, \mu_0)$ uma rede de Petri, onde μ_0 é a marcação inicial. Seja $AC = (V, A)$ uma árvore de alcançabilidade de R . Seja $v_i \in V$, $i = 1, 2, 3, \dots$, uma marcação alcançável de R , v_i é dito nó duplicado se $v_i = v_k$, para algum $v_k \in V$ [Pet-81].

Definição 5.4 - Nó interior : Seja $R = (P, T, I, O, \mu_0)$ uma rede de Petri, onde μ_0 é a marcação inicial. Seja $AC = (V, A)$ uma árvore de alcançabilidade de R . Seja $v_i \in V$, $i = 1, 2, 3, \dots$, uma marcação alcançável de R , v_i é dito nó interior se v_i é um nó de partida de arcos para novos nós [Gom-96].

Definição 5.5 - Nó fronteira : Seja $R = (P, T, I, O, \mu_0)$ uma rede de Petri, onde μ_0 é a marcação inicial. Seja $AC = (V, A)$ uma árvore de alcançabilidade de R . Seja $v_i \in V$, $i = 1, 2, 3, \dots$, uma marcação alcançável de R , v_i é dito nó fronteira se v_i é um nó criado para representar uma nova marcação inserida na árvore de alcançabilidade. Após o seu processamento, um nó fronteira pode ser classificado como nó terminal, nó duplicado ou nó interior [Gom-96].

5.1.2 - Algoritmo de Construção da Árvore de Cobertura

A árvore de alcançabilidade pode ser construída seguindo-se o algoritmo dado abaixo [Gom-96] :

1. A marcação inicial μ_0 representa a raiz da árvore
2. Enquanto houver transições potencialmente disparáveis para uma nova marcação faça
 - Selecione uma nova marcação μ_i
 - Se a marcação μ_i é idêntica a uma marcação no caminho desde a raiz até este momento, μ_i , então rotule-a como *atingida* e selecione outra marcação
 - Se nenhuma transição está habilitada para μ_i , rotule-a como marcação final (nó terminal)
 - Enquanto existir transições habilitadas para μ_i , faça :
 - para cada transição habilitada
 - Obtenha uma nova marcação μ_k que resulte do disparo de uma transição t_j habilitada para μ_i
 - Se no caminho da raiz para a marcação μ_i existe uma marcação μ_s tal que para todo lugar $p_i \in P$, $\mu_k(p_i) \geq \mu_s(p_i)$

e $\mu_k \neq \mu_s$ (μ_s cobre μ_k) então substitui-se $\mu_k(p_i)$ por ω para cada p_i tal que $\mu_i(p_i) > \mu_k(p_i)$, onde ω é uma variável inteira e positiva.

- Introduza μ_k como um nó e crie um arco rotulado com $t_j \in T$, de μ_i para μ_k e rotule a marcação μ_k como nova.

Veja a Figura 5.1 representando uma rede de Petri. Observe que as transições t_1 e t_2 estão habilitadas para serem disparadas para a marcação inicial $\mu_0 = (1, 0, 0)$, logo a árvore de alcançabilidade da rede de Petri dada na Figura 5.1, representada pela Figura 5.2, mostra as possíveis marcações quando t_1 ou t_2 são disparadas. Observe ainda que é possível disparar t_1 infinitas vezes, logo tem-se que p_2 acumula marcas e, conseqüentemente, observa-se que a rede dada é ilimitada e que a árvore de alcançabilidade é infinita. Uma solução para este problema é inserir um corte, atribuindo ao lugar ilimitado uma variável ω que indica uma marcação crescente para o lugar p_2 . Veja também que o lugar p_1 é seguro (*safe*), pois sempre possuirá uma marca ou nenhuma. Logo torna-se mais claro verificar a construção da árvore de alcançabilidade segundo o algoritmo dado anteriormente.

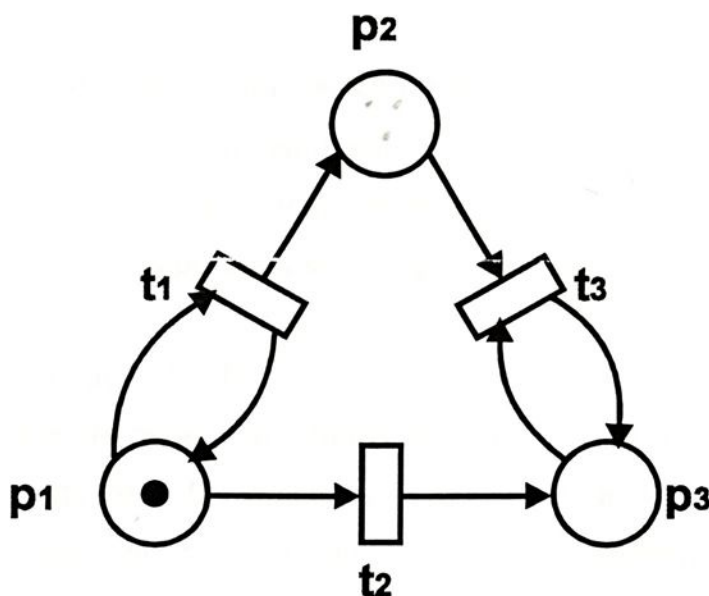


Figura 5.1 - Rede de Petri [Rei-85].

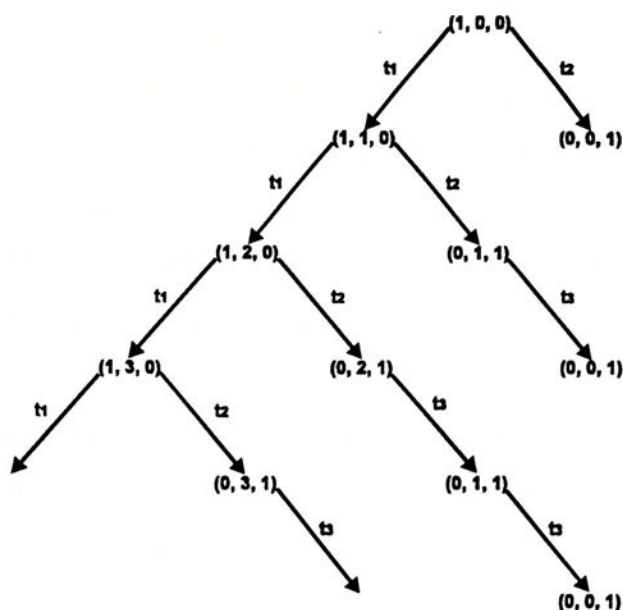


Figura 5.2 - Árvore de alcançabilidade da rede de Petri representada na Figura 5.1 [Rei-85].

Embora através da árvore de alcançabilidade seja possível observar as propriedades de limitação, segurança, reiniciabilidade e alcançabilidade, não é possível verificar as propriedades de vivacidade e alcançabilidade em redes não limitadas. Este problema está relacionado com a inserção da variável ω para indicar lugares ilimitados, pois não se sabe qual o seu valor real.

5.1.3 - Equação fundamental das redes de Petri

Utiliza-se a equação fundamental das redes de Petri para analisar a acessibilidade de marcações, bem como o número de vezes que uma transição necessita ser disparada para atingir tal marcação.

Definição 5.6 - *Equação fundamental das redes de Petri*: Seja $R = (P, T, I, O, \mu_0)$ uma rede de Petri, onde μ_0 é a marcação da inicial e I e O as matrizes de entrada e saída da rede R . Seja $p_i \in P$ um lugar qualquer. Seja μ_1 uma marcação alcançável com o disparo de uma transição habilitada $t_j \in T$, a partir de μ_0 , logo pela definição 2.13

$$\mu_1(p_i) = \mu_0(p_i) - \#(p_i, I(t_j)) + \#(p_i, O(t_j))$$

ou seja, $\mu_1 = \mu_0 - I + O$, que determina a próxima marcação da rede. A equação fundamental das redes de Petri é definida como

$$\mu_k(p_i) = \mu_0(p_i) + s^t * (\#(p_i, O(t_j)) - \#(p_i, I(t_j)))$$

onde s é um vetor característico cujos elementos s_i são números inteiros positivos e representam o número de vezes que cada transição t_j foi disparada para se atingir a marcação μ_k a partir de μ_0 .

Essa técnica vem sendo usada para verificar a propriedade de alcançabilidade, pois é possível verificar se uma determinada marcação μ_k pode ser obtida a partir da marcação inicial μ_0 .

5.2 - Cálculo de Invariantes

O comportamento de sistemas depende da estrutura da rede e de sua marcação inicial. A estrutura da rede pode ser analisada independentemente de algum processo dinâmico, como por exemplo sua execução realizando a troca de marcação [Lau-87].

Um invariante de uma rede de Petri é uma propriedade que garante cada uma das marcações alcançáveis da rede. Dada uma propriedade invariante de um sistema e uma marcação inicial, cada um dos invariantes asseguram todos os estados do sistema ou nenhum deles [Des-93].

5.2.1 - Invariantes de lugares das redes de Petri

O cálculo de invariantes de lugares de uma rede de Petri é um método de análise baseado na equação fundamental das redes de Petri usado para determinar componentes conservativos do sistema, sem a necessidade de se fazer uma observação exaustiva da árvore de alcançabilidade. Como foi visto nas definições 4.15 e 4.16 se uma rede tem componentes conservativos então existe uma soma ponderada das marcações dos lugares associados de tal forma que esta soma é constante. Usa-se a soma ponderada, dada na definição 4.16, e não simplesmente a soma, pois o disparo de uma transição



pode gerar mais ou menos marcas do que o número previamente existente, sem que ocorra um comprometimento da propriedade conservativa.

Definição 5.7 - *Invariante de lugar das redes de Petri* : Seja $R = (P, T, I, O, \mu_0)$ uma rede de Petri, onde μ_0 é a marcação inicial e I e O as matrizes de entrada (pré-condições) e saída (pós-condições) da rede R . Seja V um vetor de números inteiros positivos, com dimensão igual ao número de lugares P . V_p é chamado invariante de lugar (P-invariante) de uma rede de Petri R se $D * V_p^t = 0$, onde D é a matriz de incidência da rede R . A componente v_i de V_p é um invariante para o lugar $p_i \in P$ [Lau-87].

Definição 5.8 - *Combinação linear de P-invariante é um P-invariante* : Seja $R = (P, T, I, O, \mu_0)$ uma rede de Petri, onde μ_0 é a marcação inicial e I e O as matrizes de entrada (pré-condições) e saída (pós-condições) da rede R . Seja V_p um P-invariante da rede de Petri R , então toda combinação linear inteira positiva de V_p é um P-invariante de R , para $V_p > 0$ [Lau-87].

Definição 5.9 - *Lei da conservação das marcas* : Seja $R = (P, T, I, O, \mu_0)$ uma rede de Petri, onde μ_0 é a marcação inicial e I e O as matrizes de entrada (pré-condições) e saída (pós-condições) da rede R . Seja V_p um P-invariante da rede de Petri R . Seja μ_k uma marcação alcançável a partir de μ_0 , então se $\mu_0 * V_p^t = \mu_k * V_p^t$ a rede R é conservativa [Des-93].

Definição 5.10 - *Uma condição necessária para a vivacidade* : Seja $R = (P, T, I, O, \mu_0)$ uma rede de Petri, onde μ_0 é a marcação inicial e I e O as matrizes de entrada (pré-condições) e saída (pós-condições) da rede R . Seja V_p um P-invariante da rede de Petri R , se $V_p > 0$ então satisfaz $\mu_0 * V_p^t > 0$ [Des-93].



Definição 5.11 - Uma condição necessária para a limitação : Seja $R = (P,T, I, O, \mu_0)$ uma rede de Petri, onde μ_0 é a marcação inicial e I e O as matrizes de entrada (pré-condições) e saída (pós-condições) da rede R . Seja V_p um P-invariante da rede de Petri R , $V_p > 0$ então a rede é limitada [Des-93].

Considere a Figura 5.3 representando uma rede de Petri com marcação inicial $\mu_0 = (1,0,1,0,0,0,2)$ e matriz de incidência

$$D = \begin{bmatrix} p_1 & p_2 & p_3 & p_4 & p_5 & p_6 & p_7 \\ -1 & 1 & 0 & 0 & 0 & 0 & -1 \\ 0 & -1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & -1 & 1 & 0 & 0 & -1 \\ 0 & 0 & 0 & -1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 & 0 \end{bmatrix} \begin{matrix} t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \\ t_6 \end{matrix}$$

e o seguinte cálculo dos invariantes $D * V_p^t = 0$, que determina um conjunto de equações que após solucianá-las, usando algum método linear, tem-se o seguintes resultados $V_p = (v_1, 2v_1, v_3, v_1 + v_3, v_1, v_3, v_1)$. Atribuindo-se valores às componentes v_1 e v_3 de V_p , determina-se os P-invariantes da rede. Por exemplo, para $v_1 = 1$ e $v_3 = 1$, $V_p = (1, 2, 1, 2, 1, 1, 1)$. Observe que pela definição 5.9 a rede dada é conservativa.

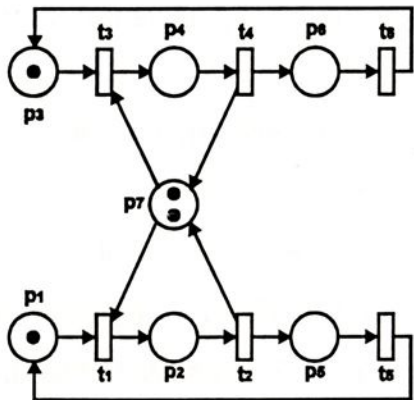


Figura 5.3 - Rede de Petri conservativa [Mac-96].

Desta forma, é condição necessária para que μ_i seja alcançável a partir de μ_0 que a solução V_p possua apenas valores inteiros não-negativos. O fato de não existir essa solução é condição suficiente para não acessibilidade de μ_i .

5.2.2 - Invariantes de transição das redes de Petri

A definição de invariantes de transição das redes de Petri é análoga à definição de invariantes de lugares das redes de Petri.

Se, em uma rede qualquer, a partir de uma marcação alcançável μ_k é possível disparar cada uma das transições n_i vezes e retornar à marcação inicial μ_0 diz-se que a rede tem componentes repetitivos estacionários. Este comportamento corresponde ao comportamento cíclico da rede, ou seja, a rede dada é reiniciável. Os componentes repetitivos estacionários são obtidos a partir da solução de um sistema do tipo $V * D = 0$, onde D é a matriz de incidência da rede e $V \geq 0$ é o vetor característico dado pela solução do sistema denominado invariante de transição (T-invariante) da rede.

Definição 5.12 - Invariante de transição das redes de Petri: Seja $R = (P, T, I, O, \mu_0)$ uma rede de Petri, onde μ_0 é a marcação inicial e I e O as matrizes de entrada (pré-condições) e saída (pós-condições) da rede R . Seja V um vetor de números inteiros positivos, com dimensão igual ao número de transições T . V_t é chamado invariante de transição (T-invariante) de uma rede de Petri R se $V_t * D = 0$, onde D é a matriz de incidência da rede R . A componente v_i de V_t é chamada de invariante para a transição $t_j \in T$ [Lau-87].

Definição 5.13 - Repetição de marcas: Seja $R = (P, T, I, O, \mu_0)$ uma rede de Petri, onde μ_0 é a marcação inicial e I e O as matrizes de entrada (pré-condições) e saída (pós-condições) da rede R . Seja μ_i uma marcação alcançável a partir da marcação inicial μ_0 . Seja σ uma seqüência de transições da rede R , habilitada em μ_i , então V_t é um T-invariante se após o disparo de σ a marcação atingida μ_k é igual a μ_i , ou seja, $\mu_i \xrightarrow{\sigma} \mu_k$ [Lau-87].

Considere a Figura 5.3 representando uma rede de Petri com marcação inicial $\mu_0 = (1,0,1,0,0,0,2)$ e matriz de incidência



$$D = \begin{bmatrix} p_1 & p_2 & p_3 & p_4 & p_5 & p_6 & p_7 \\ -1 & 1 & 0 & 0 & 0 & 0 & -1 \\ 0 & -1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & -1 & 1 & 0 & 0 & -1 \\ 0 & 0 & 0 & -1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 & 0 \end{bmatrix} \begin{matrix} t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \\ t_6 \end{matrix}$$

e o seguinte cálculo dos invariantes de transição $V_t * D = 0$, determina um conjunto de equações que após solucioná-las, usando algum método linear, tem-se os seguintes resultados $V_t = (v_1, v_1, v_3, v_3, v_1, v_3)$. Atribuindo-se valores a v_1 e v_3 de V_t , determina-se os T-invariantes da rede. Por exemplo, para $v_1 = 1$ e $v_3 = 0$, $V_t = (1, 1, 0, 0, 1, 0)$. Observe que pela definição 5.13 a rede dada é reiniciável, pois, após disparar todas as transições apenas uma vez a rede retorna à marcação inicial.

CAPÍTULO VI

6 - Redes de Petri Coloridas

As redes de Petri coloridas são os tipos de redes de alto nível com maior destaque. As redes de Petri coloridas, resumidamente CP-nets, foram propostas por Kurt Jensen por volta de 1981 [Jen-81].

As CP-nets são usadas para descrever sistemas concorrentes, os quais necessitam de tipos de redes mais poderosas. O principal objetivo das CP-nets é a redução do tamanho do modelo, permitindo que marcas individualizadas representem diferentes processos ou recursos em uma mesma sub-rede. O funcionamento das CP-nets baseia-se na atribuição de cores às marcas, isto é, uma marca passa a ter atributos associados (cor) e os arcos podem alterar esses atributos de acordo com a função associadas a eles. Sendo assim, a uma marca pode-se associar informações mais complexas.

Nas redes de Petri ordinárias existe apenas um tipo de marca, o que não permite a diferenciação de recursos em um lugar, sendo necessário lugares distintos para expressar recursos similares. As marcas das CP-nets são representadas por cores ou mesmo por padrões que possibilitam a distinção das marcas. Nesta classe de rede os arcos são rotulados com uma ou mais das possíveis cores ou ainda com variáveis.

6.1 - Principais motivos para a utilização das redes de Petri coloridas

Existem três diferentes motivos para se usar as CP-nets, os quais estão diretamente ligados. Primeiramente, uma CP-net é uma descrição de um sistema modelado e pode ser usada como uma especificação ou representação do modelo. Por exemplo, criando um modelo pode-se investigar um novo sistema, antes do sistema ser construído. Segundo, o comportamento do modelo pode ser analisado por simulação ou por métodos de análise mais



formais. Finalmente o processo de desenvolver a análise, geralmente indica ao modelador, as melhorias a serem feitas no sistema modelado.

6.2 - Funcionamento das redes de Petri coloridas

As redes de Petri coloridas se comportam de modo semelhante às ordinárias, com a vantagem de poder representar sistemas mais complexos de forma mais resumida. Para apresentar o funcionamento das CP-nets será usado um exemplo. Será utilizada uma rede que modela um sistema de manufatura de dois tipos de produtos distintos em duas linhas de produção, onde recursos, representando máquinas, são compartilhados nos processos de fabricação. Inicialmente, veja este sistema representado por uma rede de Petri Lugar/Transição mostrada na Figura 6.1.

Neste exemplo, o sistema é representado por cinco estados locais para os processos denominados $p(sp_1, sp_2, sp_3, sp_4, sp_5)$ e cinco estados locais para os processos denominados $q(sq_1, sq_2, sq_3, sq_4, sq_5)$; tem-se uma instância do processo q e duas do processo p representadas por uma marca no lugar sq_1 e pelas duas marcas no lugar sp_1 . Os recursos compartilhados são representados pelas marcas nos lugares r e s . Observe que se existisse um terceiro processo semelhante compartilhando os recursos r e s , seria necessário descrever todos os seus estados, ações e relações entre o novo processo e os recursos disponíveis. Assim fica claro que para um número finito, porém muito grande de processos, haveria uma variação na dimensão da rede.

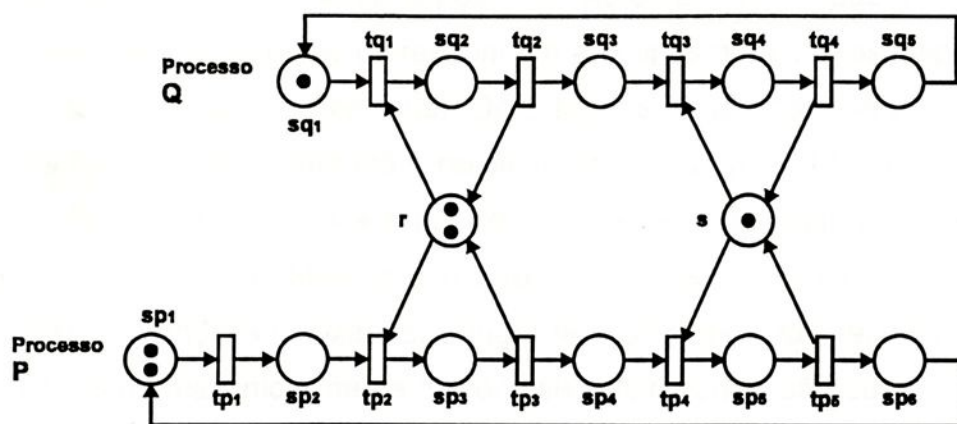


Figura 6.1 - Rede de Petri Lugar/Transição representando uma linha de manufatura [Mac-96].

A rede de Petri colorida a ser exemplificada possui três partes : estrutura, declaração e inscrição, como é apresentado na Figura 6.2.

A estrutura é um grafo dirigido composto de dois tipos de nós representados por círculos e retângulos para indicar os lugares e as transições, respectivamente. Cada lugar pode armazenar diversas marcas e cada marca representa um valor associado ao tipo de estrutura de dados. As declarações especificam o conjunto de cores e a declaração de variáveis. Cada lugar tem um conjunto de cores associados. Desta forma as marcas que são armazenadas nesses lugares devem ter a mesma cor do conjunto de cores associado à rede. Nesta classe de redes, cada lugar, transição e arco tem inscrições associadas. Os lugares têm três tipos de inscrição representando seu nome, conjunto de cores associado e uma expressão geradora da marcação inicial. As transições possuem dois tipos de inscrição representando o seu nome e uma expressão-guarda. Cada um dos arcos apresenta apenas um tipo de inscrição dado pelas respectivas expressões.

A marcação inicial de um lugar é representado por um multi-conjunto de cores. As expressões-guarda associadas às transições são expressões booleanas que devem ser satisfeitas para que se possa disparar uma transição. As expressões associadas aos arcos podem conter variáveis, constantes, funções e operações.

A rede da Figura 6.2 é formada por um conjunto de sete lugares e seis transições. O conjunto de cores que podem ser armazenadas nos lugares $S = (s_1, s_2, s_3, s_4, s_5)$ é dado pelo produto cartesiano de $Proc = \{p, q\}$ com o conjunto finito de números inteiros positivos. As cores associadas aos lugares que representam os recursos correspondem aos tipos de recursos disponíveis, no caso $rec = r$ ou $rec = s$ ($rec = rls$). Os lugares s_1 e s_2 e *recurso* possuem as seguintes funções de inicialização, respectivamente, $2'(p,0)$, $1'(q,0)$ e $2'r + 1's$, onde s_1 possui duas marcas e a cor do processo é p . O lugar s_2 possui uma marca onde a cor do processo é q , o zero, em ambos os lugares, corresponde ao número de iterações realizadas. O lugar *recurso* possui dois recursos r e um recurso s . As marcas inicialmente disponíveis nos recursos são duas da cor r e uma da cor s . Tem-se as transições t_1 e t_2 habilitadas. A transição t_2 pode ser

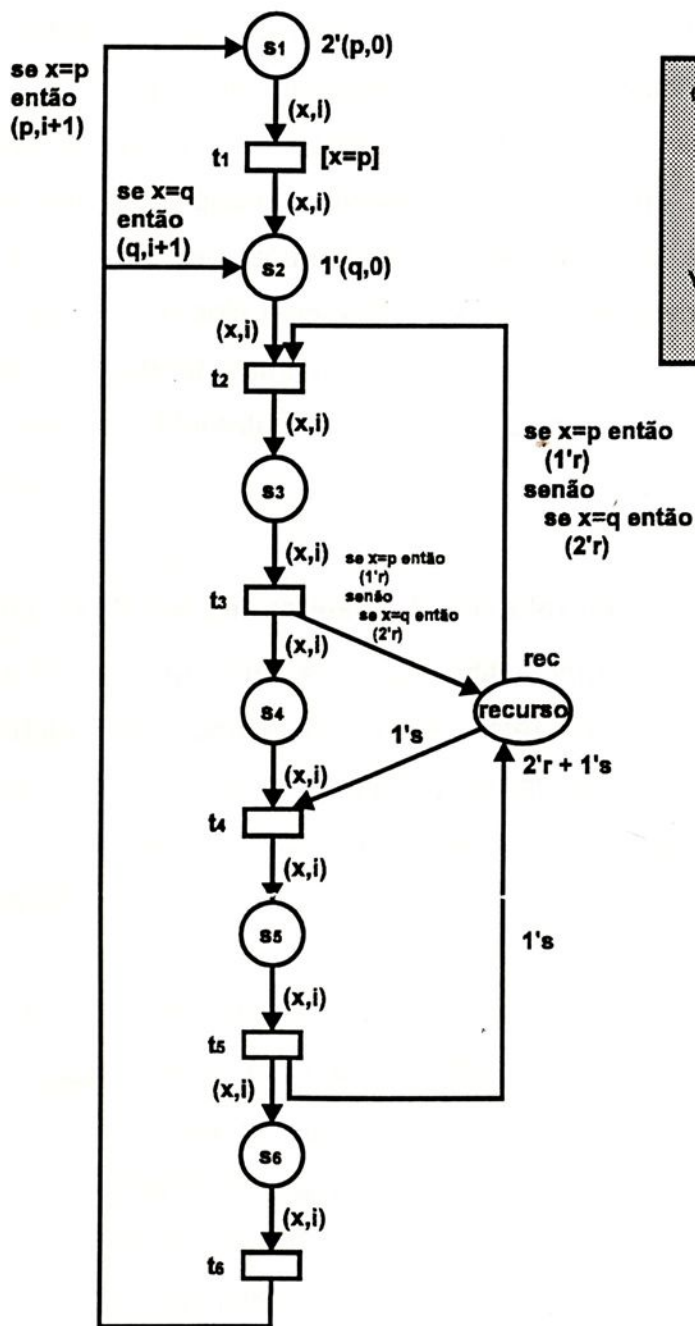



```

Cor
  I = Int;
  Proc = p|q;
  Rec = r|s;
  L = Proc * I;

Var
  x : Proc;
  i : I;

```



64

O sistema linha de manufatura está representado na Figura 6.1 por uma rede de Petri Lugar/Transição e na Figura 6.2 por uma rede de Petri colorida. Na Figura 6.1 tem-se duas redes representando cada um dos processos p e q modelados, os lugares r e s representando os recursos disponíveis no sistema. Na Figura 6.2 todos os processos compartilham a mesma estrutura da rede e todos os recursos são atribuídos a um único lugar. Tanto processos como recursos são identificados por cores.

A vantagem de se usar redes de Petri coloridas é que, caso seja necessário adicionar outro processo ao sistema adiciona-se uma nova cor à rede, estabelece-se uma nova rota para este processo, assim como, adapta-se as expressões-arco quando necessário. O mesmo procedimento ocorre quando da adição de recursos diferentes de r e s . Na rede de Petri Lugar/Transição caso seja necessário adicionar um novo processo ao sistema é necessário adicionar outra rede representando este novo processo. Para adicionar recursos diferentes de r e s teria que adicionar um lugar para cada novo recurso.

6.3 - Definição formal das redes de Petri coloridas (CP-nets)

Nas CP-nets, as marcas são definidas como tipos de dados e é possível efetuar operações complexas sobre estes dados. As marcações são representadas por cores em lugares individuais. Aos arcos, que ligam os lugares às transições, são associadas expressões, que servem para habilitar uma transição.

Definição 6.1 - *Redes de Petri coloridas (CP-nets)* : Uma CP-net é uma n-upla $CPN = (\Sigma, P, T, A, N, C, G, E, I)$, onde :

1. Σ é um conjunto de tipos não vazios, chamado de conjunto de cores
2. P é o conjunto de lugares associados à rede CPN
3. T é o conjunto de transições associados à rede CPN
4. P e T são conjuntos disjuntos, ou seja, $P \cap T = \emptyset$
5. A é o conjunto finito de todos os arcos da CPN



6. N é a função nó, é definida de A em $P \times T \cup T \times P$, permitindo múltiplos arcos entre dois nós, sendo um nó lugar e o outro nó transição.
7. C é uma função cor, é definida de P em Σ
8. G é uma função guarda, é definida de T em expressões tais que $\forall t \in T$ existe um arco $a \in A$ e uma expressão rotulada ao arco a tal que o tipo da variável $G(t) \subseteq \Sigma$
9. E é uma expressão arco, é definida de A em expressões tais que $\forall a \in A$ o tipo de $E(a) = C(p)$ e tipo da variável de $E(a) \subseteq \Sigma$, onde p é um lugar de $N(a)$
10. I é uma função de inicialização, é definida de P em expressões tais que $\forall p \in P$ o tipo de $I(p) = C(p)$

Para melhor entender uma rede de Petri colorida, veja as explicações para cada item da definição 6.1 [Jen-94] :

1. Σ é o conjunto de tipos que determina os valores dos dados, as operações e as funções que podem ser usadas nas expressões da rede, isto é, expressões arco, expressões guarda e expressões de inicialização. Considere que cada tipo tem pelo menos um elemento.
2. P , T , A são conjuntos que descrevem os lugares, as transições e os arcos da rede, respectivamente, os quais devem ser finitos e disjuntos entre si. A exigência dos conjuntos de lugares, transições e arcos serem finitos evita um número de problemas técnicos, tal como a possibilidade de um número infinito de arcos entre dois nós.
3. N é uma função nó que mapeia cada um dos arcos em um par ordenado onde o primeiro elemento é o nó origem e o segundo elemento é o nó destino. Os dois nós têm de ser de tipos distintos, ou seja, um deve ser um lugar enquanto o outro deve ser uma transição ou vice-versa. Permitindo que uma CP-net tenha vários arcos entre o mesmo par ordenado de nós.
4. A função cor C mapeia cada lugar $p \in P$ para um tipo $C(p)$, ou seja, cada marca em p deve ter um valor pertencente a $C(p)$.



5. A função guarda G mapeia cada transição $t \in T$ em uma expressão booleana, onde todas as variáveis têm tipo pertencente a Σ . Nas CP-nets as expressões guarda são omitidas, pois são consideradas como sendo sempre verdadeiras.
6. A expressão arco E mapeia cada arco em uma expressão do tipo $C(p)$. Isto significa que cada expressão arco deve ser avaliada para um multi-conjunto sobre o tipo dos lugares adjacentes p .
7. A função inicialização I mapeia cada lugar p em uma expressão que deve ser do tipo $C(p)$.

Definição 6.2 - *Lugar de entrada de uma CP-net* : Seja $CPN = (\Sigma, P, T, A, N, C, G, E, I)$ uma CP-net. Seja $p_i \in P$ um lugar qualquer de CPN. Seja $t_i \in T$ uma transição qualquer de CPN. p_i é um lugar de entrada de t_i se existir um arco $a \in A$, onde $a = (p_i, t_i)$, p_i é o nó origem e t_i é o nó destino [Jen-87].

Definição 6.3 - *Lugar de saída de uma CP-net* : Seja $CPN = (\Sigma, P, T, A, N, C, G, E, I)$ uma CP-net. Seja $p_i \in P$ um lugar qualquer de CPN. Seja $t_i \in T$ uma transição qualquer de CPN. p_i é um lugar de saída de t_i se existir um arco $a \in A$, onde $a = (t_i, p_i)$, t_i é o nó origem e p_i é o nó destino [Jen-87].

6.4 - Comportamento dinâmico das redes de Petri coloridas

A versão mais moderna das CP-nets usa as expressões dadas na definição 6.1 não somente quando o sistema está sendo descrito mas também quando está sendo analisado. Após definir a estrutura da rede será necessário considerar seu comportamento. Para isso deve-se considerar as seguintes notações para todo $t \in T$ e para todo par de nós $(x_1, x_2) \in (P \times T \cup T \times P)$:

- $A(t) = \{a \in A \mid N(a) \in P \times \{t\} \cup \{t\} \times P\}$
- $Var(t) = \{v \mid v \in Var(G(t)) \vee \exists a \in A(t) : v \in Var(E(a))\}$
- $A(x_1, x_2) = \{a \in A \mid N(a) = (x_1, x_2)\}$
- $E(x_1, x_2) = E(a)$

Definição 6.4 - Ligação entre nós das CP-nets : Seja $CPN = (\Sigma, P, T, A, N, C, G, E, I)$ uma CP-net. Uma ligação de uma transição t a um lugar p é uma função b definida em $Var(t)$, tal que :

1. $\forall v \in Var(t) : b(v) \in Tipo(v)$
2. $G(t) < b >$, denota a evolução da expressão guarda $G(t)$ na ligação b , ou seja, todas as ligações satisfazem a expressão guarda correspondente.

$B(t)$ denota o conjunto de todas as ligações para t [Jen-94].

Definição 6.5 - Marcação das CP-nets : Seja $CPN = (\Sigma, P, T, A, N, C, G, E, I)$ uma CP-net. Uma marca é um par do tipo (p, c) , onde $p \in P$ e $c \in C(p)$. Uma ligação é um par do tipo (t, b) , onde $t \in T$ e $b \in B(t)$. O conjunto de todas as marcas é denotado por TE e o conjunto de todas as ligações para todas as transições da rede é denotado por BE . Uma marcação de uma CP-net é um multi-conjunto sobre TE tal que $\forall (p, c) \in TE : \mu(p, c) = (I(p))(c)$ [Jen-94].

Assim, μ_0 representa a marcação inicial das CP-nets. Uma marcação μ de uma CP-net é uma função definida em P , tal que $\mu(p) \in C(p)$ para todo $p \in P$. Elementos de $C(p)$ são chamados cor [Jen-94].

Definição 6.6 - Habilitação de transição das CP-nets : Seja $CPN = (\Sigma, P, T, A, N, C, G, E, I)$ uma CP-net. Seja $t \in T$ uma transição qualquer de CPN. Seja μ uma marcação qualquer da rede. t é dita habilitada na marcação μ se para $\forall p \in P : \sum E(p, t) < b > \leq \mu(p)$ [Jen-94].

Definição 6.7 - Disparo de uma transição habilitada das CP-nets : Seja $CPN = (\Sigma, P, T, A, N, C, G, E, I, \mu_0)$ uma CP-net, onde μ_0 é a marcação inicial da rede. Seja $t \in T$ uma transição habilitada na marcação μ_0 de CPN. O disparo da transição t é definido por

$$\forall p \in P : \mu_1(p) = (\mu_0(p) - \sum E(p, t) < b >) + \sum E(t, p) < b >)$$



Isto representa a troca de marcação da CP-nets. Portanto, se existir uma transição t_j qualquer habilitada em uma marcação μ_i qualquer na rede, então o disparo da transição t_j representa uma nova marcação na rede, também chamado de novo estado da rede. Esta troca de marcação é dada por

$$\forall p \in P : \mu_k(p) = (\mu_i(p) - \sum E(p, t_j) \langle b \rangle) + \sum E(t_j, p) \langle b \rangle$$

onde μ_k é a nova marcação da rede. Diz-se μ_k que é diretamente alcançável a partir de μ_i [Jen-94].

Definição 6.8 - Seqüência de marcações das CP-nets : Seja $CPN = (\Sigma, P, T, A, N, C, G, E, I, \mu_0)$ uma CP-net, onde μ_0 é a marcação inicial da rede. Seja t_j uma transição habilitada na marcação μ_0 . Uma ocorrência de seqüência finita é uma seqüência de marcações dada por

$$\mu_0 [t_j > \mu_1 [t_{j1} > \mu_2 [t_{j2} > \dots \mu_n [t_{jn} > \mu_{n+1}$$

tal que $n \in \mathbb{N}^+$, e $\mu_i [t_{ji} > \mu_{i+1}$ para todo $i \in \{1, 2, \dots, n\}$ e μ_0 é a marcação inicial, μ_{n+1} é a marcação final e n é o tamanho da seqüência. O conjunto de marcações as quais são alcançáveis a partir de μ_0 é denotada por $[\mu_0 >$ [Jen-87].

Definição 6.9 - Marcação acessível das CP-nets : Seja $CPN = (\Sigma, P, T, A, N, C, G, E, I, \mu_0)$ uma CP-net, onde μ_0 é a marcação inicial da rede. Seja $t \in T$ uma transição habilitada em uma marcação qualquer $\mu_i \in [\mu_0 >$ da CPN. O disparo da transição t gera uma nova marcação dada por

$$\forall p \in P : \mu_k(p) = (\mu_i(p) - \sum E(p, t) \langle b \rangle) + \sum E(t, p) \langle b \rangle$$

onde μ_k é uma marcação alcançável (acessível) da CPN, a partir de μ_0 [Jen-94].

6.5 - Propriedades comportamentais das redes de Petri coloridas

As propriedades comportamentais caracterizam o comportamento individual das CP-nets. É através das propriedades comportamentais que se



verifica se é possível alcançar alguma marcação em um determinado passo da rede.

6.5.1 - Limitação

A propriedade limitação indica a quantidade de marcas de uma determinada cor em lugares individuais. Caso o número de marcas, em alguns lugares, se acumulem isto indica que a CP-net é ilimitada.

Definição 6.10 - *CP-nets limitada* : Seja $CPN = (\Sigma, P, T, A, N, C, G, E, I, \mu_0)$ uma CP-net, onde μ_0 é a marcação inicial da rede. Seja $p \in P$ um lugar qualquer da CPN, um número inteiro não negativo $n \in \mathbb{N}$ e um multi-conjunto $m \in C(p)$ diz-se que

1. n é um limite superior para p se $\forall \mu \in [\mu_0 > : |\mu(p)| \leq n$
2. m é um multi-conjunto limitado superiormente para p se $\forall \mu \in [\mu_0 > : |\mu(p)| \leq m$ [Jen-92].

Zero é um limite inferior para todo multi-conjunto.

6.5.2 - Vivacidade

A propriedade vivacidade indica que um conjunto de elementos de ligação permanecem ativos. Isto significa que em algum passo existe alguma transição habilitada para uma marcação qualquer da rede.

Definição 6.11 - *CP-net viva* : Seja $CPN = (\Sigma, P, T, A, N, C, G, E, I, \mu_0)$ uma CP-net, onde μ_0 é a marcação inicial da rede. Seja $\mu \in [\mu_0 >$ uma marcação. Seja $X \subseteq BE$ um conjunto de elementos de ligação, então

1. μ é morta se nenhum elemento de ligação está habilitado, isto é, se $\forall x \in BE : \neg \mu[x >$
2. X é morto em μ se nenhum elemento de X pode tornar-se habilitado, isto é, se $\forall \mu_i \in [\mu > e \forall x \in X : \neg \mu_i[x >$

3. X é vivo se não existe nenhuma marcação alcançável na qual X é morto, isto é, se $\forall \mu_i \in [\mu_0 > \exists \mu_k \in [\mu_i > \exists x \in X : \mu_k [x > [Jen-92]$.

A propriedade vivacidade exige que os elementos de X possam tornar-se habilitados. Para uma transição $t \in T$, usa-se $BE(t) \subseteq BE$ para denotar o conjunto de todos os elementos de ligação que contenham t . Diz-se que a transição t é viva ou morta se possuir a propriedade vivacidade. Também, diz-se que t é estritamente viva se $\{x\}$ é vivo para todo $x \in BE(t)$.

6.5.3 - Reiniciabilidade

A propriedade reiniciabilidade indica se é possível atingir a marcação inicial após o disparo de uma seqüência de transições habilitadas em uma seqüência de marcações. Mas, em algumas aplicações, é suficiente garantir o retorno a alguma marcação no multi-conjunto de marcações alcançáveis, denominado lugar *home*, desta forma a rede é considerada reversível [Jen-92].

Definição 6.12 - *Home-State* : Seja $CPN = (\Sigma, P, T, A, N, C, G, E, I, \mu_0)$ uma CP-net, onde μ_0 é a marcação inicial da rede. Seja $\mu \in [\mu_0 >$ uma marcação. Seja $X \subseteq [\mu_0 >$ um conjunto de marcações, então :

1. μ é uma marcação *home-state* se $\forall \mu_i \in [\mu_0 > : \mu \in [\mu_i >$;
2. X é um *home-space* se $\forall \mu_i \in [\mu_0 > : X \cap [\mu_i > \neq \emptyset$ [Jen-94].

Definição 6.13 - *Rede de Petri reiniciável* : Seja $CPN = (\Sigma, P, T, A, N, C, G, E, I, \mu_0)$ uma CP-net, onde μ_0 é a marcação inicial da rede. Seja $\forall \mu \in [\mu_0 >$ uma marcação de CPN. CPN é dita reiniciável se $\mu = \mu_0$ [Jen-92].

Nesta seção foram apresentadas as propriedades comportamentais mais importantes das redes de Petri e estas serão usadas nos próximos capítulos para a análise comportamental dos sistemas, assim como para a implementação do simulador que fará a análise.



6.6 - Análise das redes de Petri coloridas

A análise das CP-nets pode ser feita por meio de um grafo, onde tem-se todas as marcações alcançáveis, a partir da marcação inicial e considerando cada transição habilitada para cada uma das cores presentes nos lugares da rede. A construção deste grafo é muito parecida com a construção da árvore de alcançabilidade citada no capítulo V. Também faz-se a análise das CP-nets por meio de invariantes de lugar ou invariantes de transição.

6.6.1 - Árvore de alcançabilidade de uma rede de Petri colorida

A idéia básica de uma árvore de alcançabilidade é construir um grafo contendo um nó para cada marcação alcançável e um arco ligando cada elemento ocorrido, assim como definido no capítulo V, definição 5.1. Cada um dos arcos representa uma marcação ocorrida, ou seja, para uma marcação μ da rede, t uma transição habilitada em μ , o disparo de t gera uma nova marcação μ_i para a rede, sendo assim, tem-se $\mu[t > \mu_i$, μ_i é a marcação alcançável a partir de μ . Uma árvore de alcançabilidade pode tornar-se muito grande para uma CP-net muito pequena [Jen-94].

Definição 6.14 - *Grafo direcionado* : Seja $G = (V, A, N)$ um grafo direcionado tal que :

1. V é o conjunto de nós do grafo (vértices)
2. A é o conjunto de arcos tal que $A = \{\forall a_i \in A \mid a_i = (v, w) \rightarrow [v, w \in V \wedge v \neq w]\}$.
3. N é um nó função que é definido de A em $V \times V$

G é finito se A e V são finitos [Jen-94].

Note que, um grafo direcionado pode ter vários arcos entre o mesmo par ordenado de nós. Um arco a com $N(a) = (v_1, v_2)$ tem origem em v_1 e destino em v_2 .



Definição 6.15 - *Árvore de alcançabilidade de uma CP-net* : Seja $CPN = (\Sigma, P, T, A, N, C, G, E, I, \mu_0)$ uma CP-net, onde μ_0 é a marcação inicial da rede. Seja $OG = (V, A, N)$ um grafo direcionado. OG é uma árvore de alcançabilidade de CPN, tal que:

1. $V = [\mu_0>$, ou seja é o conjunto de todas as marcações alcançáveis a partir de μ_0 .
2. $A = \{(\mu_1, t, \mu_2) \in V \times BE \times V \mid \mu_1 [t > \mu_2\}$
3. $\forall a = (\mu_1, t, \mu_2) \in A : N(a) = (\mu_1, \mu_2)$ [Jen-92].

No desenho de uma árvore de alcançabilidade coloca-se um texto em cada nó para descrever a marcação que o nó representa. Assim como nos arcos coloca-se um texto para indicar qual transição ocorreu, a partir do nó (marcação) origem e gerou o nó destino (nova marcação).

A análise da árvore de alcançabilidade torna-se desagradável para grafo com muitos nós.

6.6.2 - Invariantes de lugar de uma rede de Petri colorida

A idéia básica de invariantes de lugar consiste, primeiramente, em formular algumas equações que devem ser satisfeitas independentemente dos passos que ocorram na rede. Então deve-se provar que estas equações são efetivamente satisfeitas. Finalmente, tais equações são usadas para demonstrar propriedades comportamentais do sistema modelado [Jen-81].

A análise dos invariantes de lugar de uma CP-net tem diversos aspectos muito atrativos. Em primeiro lugar, ela pode ser usada para verificar uma série de propriedades comportamentais, tais como : alcançabilidade, limitabilidade, vivacidade, eqüitatividade. Em segundo lugar, no caso de uso de redes hierárquicas, é possível obter os invariantes da rede total pela composição dos invariantes de cada nível de hierarquia, o que reduz a complexidade da análise em relação à apresentada quando do uso de árvore de alcançabilidade. Em terceiro lugar, pode-se usar invariantes de lugar sem a necessidade de fixar todos os parâmetros do sistema modelado, o que possibilita a demonstração de certas propriedades genéricas do sistema, independentemente do valor



desses parâmetros. Em último lugar, pode-se estabelecer os invariantes de lugar durante o projeto de um sistema o que, em geral, resulta em um projeto melhorado.

A principal desvantagem da análise por meio de invariantes de lugar é que ela requer uma capacitação maior e um conhecimento matemático mais profundo por parte do analista do que os outros métodos de análise.



CAPÍTULO VII

7 - Um Programa para Análise de Modelos de Sistemas usando Redes de Petri Coloridas

Neste capítulo será descrita a implementação de um programa para analisar modelos de sistemas usando redes de Petri Coloridas. Esta implementação originou um simulador *CPNETVMV*, escrito em Pascal que simula as CP-nets e analisa as propriedades de vivacidade, reiniciabilidade e limitação. Também, é construída uma lista de todas as marcações alcançáveis a partir da marcação inicial. Esta lista de marcação é considerada a árvore de alcançabilidade da rede de Petri. Caso seja verificado que a árvore de alcançabilidade venha a se tornar infinita, tem-se um nível de corte atribuído com base no algoritmo [Gom-96], citado no capítulo V, para interromper a construção desta árvore.

O simulador, após construir a árvore de alcançabilidade, faz a análise das propriedades comportamentais das redes de Petri coloridas.

Para a construção deste programa, foram definidas algumas estruturas de dados que armazenam as marcações da rede, as matrizes de entrada e saída, a matriz de incidência e as matrizes de disparo. Toda construção do programa baseia-se no uso de matrizes para representar as redes de Petri coloridas. Tais estruturas de dados serão apresentadas neste capítulo.

7.1 - Definição das estruturas construídas

Foi construída uma estrutura dinâmica para armazenar a lista de marcações alcançáveis, que tem como objetivo gerar a árvore de alcançabilidade da rede de Petri colorida que será analisada. Também construiu-se uma estrutura dinâmica para as matrizes de entrada, de saída, de incidência, de disparo e as marcações alcançáveis. O armazenamento dessas informações serve para calcular as novas marcações, a partir da marcação que



está sendo analisada, percorrendo a lista de marcações citada acima, e também para se ter todas as marcações geradas, pois serão usadas para analisar as principais propriedades comportamentais das redes de Petri coloridas. Assim como foi construída uma estrutura dinâmica para armazenar cada uma das expressões arco, para os arcos de entrada e saída das transições. Estas informações servem para verificar se é possível disparar ou não uma transição, ou seja, se uma transição é potencialmente disparável para cada uma das cores presente na rede.

7.1.1 - Estrutura dinâmica das matrizes

Tem-se dois tipos de estruturas para as matrizes do sistema, uma estrutura que armazenam valores numéricos como, por exemplo, matrizes de entrada, de saída, de incidência, de disparo e as marcações, e outra para as matrizes que armazenam informações *strings* para as expressões arco.

- Estrutura dinâmica das matrizes numéricas :

- *no = record*

rela : integer;

col, row : ^ponteiro_no;

Veja, na Figura 7.1, a representação desta estrutura, que é usada para fazer a alocação dinâmica de memória. O ponteiro *p* representa um ponteiro que está indicando o começo da estrutura, ou seja, aponta para o elemento da linha 1, coluna 1.

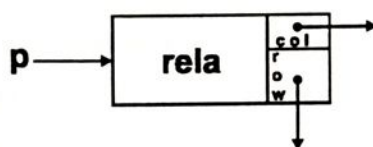


Figura 7.1 - Representação de um elemento da matriz numérica.

Para acessar cada um dos elementos desta estrutura usa-se um ponteiro que controla a linha que está sendo utilizada e outro ponteiro que controla a coluna.

As matrizes numéricas têm um campo para armazenar o número (*rela*) e dois campos que ligam cada elemento das matrizes, ou seja, o campo *col* liga o elemento à próxima coluna, assim é possível acessar a próxima coluna e o campo *row* liga o elemento à próxima linha. Nas matrizes de entrada e saída o campo *rela* armazena o valor 1, se existir um arco ligando o lugar a transição e a transição ao lugar, respectivamente, ou o valor 0 se não existir arco unindo o lugar à transição e vice-versa. Para a matriz de incidência o campo *rela* armazena 1 se for um lugar de saída da transição, ou -1 se for um lugar de entrada e 0 se não existir arco unindo os nós. Observe que os arcos não possuem pesos, estes serão atribuídos às cores e o disparo de uma transição será controlado pelas expressões arco, que será explicado na próxima seção.

Com esta estrutura dinâmica é possível construir matrizes de tamanhos variados sem ter a preocupação com a limitação de espaço na memória de seu computador, pois a limitação da memória depende somente da quantidade de memória disponível na máquina utilizada. Caso fosse usada uma estrutura estática o programa estaria com uma limitação, pois dependeria do espaço reservado. Sendo assim o simulador suporta matrizes com tamanhos variados e pode trabalhar com sistemas cujos modelos apresentam um grande número de nós.

As definições de matrizes de entrada, saída e incidência encontram-se no capítulo I. Para cada uma destas matrizes, as linhas representam as transições e as colunas representam os lugares das redes de Petri coloridas.

Para as marcações das redes não foi possível usar um vetor, como definido no capítulo II. Pois, para as CP-nets, uma marcação é representada por um multi-conjunto de cores em cada um dos lugares. Portanto, as marcações das CP-nets são representadas por matrizes, onde as linhas representam as cores usadas e as colunas representam os lugares. Sendo assim, cada coluna representa a quantidade de cada uma das cores nos respectivos lugares. A dimensão desta matriz é $\mu_{m \times n}$, onde m é a cardinalidade do conjunto de cores e n é a cardinalidade do conjunto de lugares. Para compreender melhor a construção da matriz que representa uma marcação veja a matriz μ .



$$\mu = \begin{bmatrix} p_1 & p_2 & p_3 & p_4 \\ 1 & 0 & 1 & 3 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 2 & 0 & 1 \end{bmatrix} \begin{matrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{matrix}$$

O elemento $\mu_{1,1} = 1$ indica que no lugar p_1 existe uma marca da cor 1, assim como o elemento $\mu_{4,2} = 2$ indica que no lugar p_2 existem duas marcas da cor 4. Todas as marcações da rede usam a estrutura representada na Figura 7.1.

- Estrutura dinâmica das matrizes *strings*

- *expressão* = *string*[40];

- *estrutura* = *record*

express : *expressão*;

col, row : *^ponteiro_estrutura*;

Veja, na Figura 7.2, a representação desta estrutura que é usada para fazer a alocação dinâmica de memória. O ponteiro *q* representa um ponteiro que está indicando o começo da estrutura, ou seja, aponta para o elemento da linha 1, coluna 1.



Figura 7.2 - Representação de um elemento da matriz *string*.

Para acessar cada um dos elementos desta estrutura usa-se um ponteiro que controla a linha que está sendo utilizada e outro ponteiro que controla a coluna.

As matrizes *strings* têm um campo para armazenar uma *string* (*express*) e dois campos que ligam cada elemento das matrizes, ou seja, o campo *col* liga o elemento à próxima coluna e o campo *row* liga o elemento à próxima linha.

O campo *express* é usado para armazenar cada uma das expressões arco presentes no modelo simulado. Estas expressões arco, como definido no capítulo VI, são responsáveis pela habilitação das transições, quando existirem expressões nos arcos.

As expressões arco podem ser de vários tipos, mas neste simulador foram avaliadas expressões que representam estruturas de decisão simples, ou seja, *if - then - else* e expressões que representam a pertinência de cores nos lugares.

Uma expressão do tipo *if - then - else* é avaliada da seguinte maneira

- a condição para o *if* informa quais os lugares que devem ser analisados para algum conjunto de cores indicado, onde cada cor tem a sua quantidade (peso) especificada na expressão. A condição para o *if* é analisada e, se seu resultado for verdadeiro, executam-se as atribuições para o *then*, mas se a condição for falsa executam-se as atribuições para o *else*, como é feito em programação estruturada. Expressões ligadas aos arcos de entrada indicam a retirada de cores dos lugares de entrada da transição enquanto que, expressões ligadas aos arcos de saída indicam a adição de novas marcas aos lugares de saída.

Uma expressão do tipo pertinência de cores é analisada da seguinte forma :

- As expressões arco do tipo pertinência de cores informam quais cores devem estar presentes nos lugares de entrada, que são origens dos arcos, para habilitar uma transição. A análise das expressões arco do tipo pertinência de cores para habilitar uma transição é feita verificando se as cores especificadas na expressão estão presentes nos lugares de entrada e se cada um dos lugares possuem a quantidade de cores especificada na expressão. Caso uma transição seja



habilitada pelas expressões arco do tipo pertinência de cores, então as cores são retiradas dos lugares de entrada, de acordo com a quantidade especificada, e novas cores são adicionadas nos lugares de saída da transição, de acordo com as cores e quantidades especificadas nas expressões dos arcos de saída.

Quando todos os arcos de entrada de uma transição têm expressões arco do tipo pertinência de cores, para que esta transição seja habilitada é necessário que as expressões arco presentes nos lugares de entrada sejam todas verdadeiras.

Mas pode-se ter pelo menos um arco de entrada de uma transição que não possui expressão arco. Neste caso, para uma transição estar habilitada, não basta analisar as expressões arco, é necessário que a transição esteja habilitada independentemente das expressões, ou seja, deve estar habilitada de acordo com a **definição 2.12** dada no capítulo II. Para casos como este foram construídas algumas matrizes, chamadas de matrizes de disparo, usando a mesma estrutura para as matrizes de entrada, saída, etc. Nas matrizes de disparo, as linha representam as cores e as colunas representam as transições. Essas matrizes são formadas de 0's e 1's que irão verificar se uma transição está habilitada para cada uma das cores presentes na rede. Quando uma transição é disparável para uma determinada cor, a posição referente a esta transição e cor recebe o valor 1 caso contrário recebe o valor 0. Assim, tem-se uma matriz com valores iguais a 1 para todas as cores que a transição é disparável. Este processo se repete para todas as transições que possuem arcos sem expressão e para cada transição é gerada uma nova matriz de disparo. Por exemplo, a matriz de disparo μ a seguir, indica que a transição t_1 é disparável para as cores 1 e 2.

$$\mu = \begin{bmatrix} t_1 & t_2 & t_3 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{matrix} c_1 \\ c_2 \end{matrix}$$

7.1.2 - Estrutura dinâmica da lista de marcações alcançáveis

A estrutura dinâmica da lista de marcações alcançáveis foi construída para armazenar todas as marcações geradas com o disparo das transições habilitadas, a partir da marcação inicial.

- Estrutura dinâmica da lista de marcações alcançáveis

- *estrlista = record*

trans : integer;

marcat : ponteiro_no;

próx : ^ponteiro_estrlista;

Veja na Figura 7.3 a representação desta estrutura que é usada para fazer a alocação dinâmica de memória. O ponteiro *lista* está indicando o começo da lista de marcações, ou seja, aponta para o primeiro elemento da lista. Este elemento armazena a marcação inicial da rede e, a partir dela, serão geradas novas marcações, que serão os próximos elementos da lista.

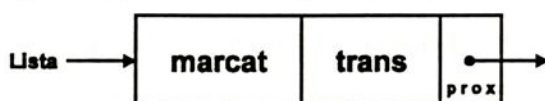


Figura 7.3 - Representação de um elemento da lista de marcações alcançáveis

Esta estrutura possui um campo para armazenar a transição que foi disparada, *trans*, um campo para armazenar a matriz que representa a marcação gerada com o disparo da transição, *marcat*, e um campo para indicar a próxima marcação, *próx*, que é outro elemento igual a este. O primeiro elemento da lista armazena a marcação inicial no campo *marcat* e a transição disparada para gerar a marcação inicial é nula, pois não existe esta transição.

É importante armazenar a seqüência de marcações e as transições que as geraram, pois a representação matricial das redes de Petri não armazena a seqüência de disparos das transições.

Para verificar as propriedades vivacidade, limitação e reiniciabilidade precisa-se da seqüência de disparos para realizar a análise.

7.2 - Organização do programa

O programa inicia-se com a descrição da rede através das seguintes leituras

- do número de lugares da rede
- do número de transições
- do número de cores
- da matriz de entrada
- da matriz de saída
- da marcação inicial
- da matriz de expressão para os arcos de entrada
- da matriz de expressão para os arcos de saída

Com estas informações tem-se a representação matricial da CP-net que será simulada.

Após o armazenamento dinâmico desta informações inicia-se a simulação. Nesta simulação usa-se a estrutura das matrizes que armazenam as expressões arco para localizar as transições disparáveis a partir da marcação inicial. Verifica-se o tipo de expressão, *if - then - else* ou pertinência de cores. Se a transição for disparável gera-se a nova marcação. Mas se existir algum arco sem expressão, usam-se as matrizes de disparo para localizar as transições disparáveis a partir da marcação inicial. Este processo se repete para todas as transições da rede, usando a marcação inicial. As novas marcações vão sendo enfileiradas na lista, ou seja, a cada nova marcação é inserido, no final da lista, um novo nó do tipo da estrutura usada para lista (*estrlista*). Após terem sido criadas todas as marcações alcançáveis, a partir da marcação inicial, um ponteiro utilizado para percorrer a lista é deslocado para o próximo nó, imediatamente após à marcação inicial. O processo se repete para todas as transições a partir da marcação armazenada no campo *marcat*, do nó que está sendo apontado atualmente. Após gerar todas as novas marcações, novamente o ponteiro utilizado para percorrer a lista é deslocado para o próximo nó da lista. Este processo se repete até que seja encontrada a marcação inicial, provando a propriedade de reiniciabilidade,



ou então, é verificado se algum lugar está acumulando alguma marca de uma das cores usadas, se isso acontecer, para um número finito de vezes, o processo é interrompido e considera-se a rede ilimitada, provocando um corte, como foi definido no capítulo V, seção 5.1.1. Mas se a rede não retorna à marcação inicial e nenhum lugar acumula marcas então o simulador ficará repetindo o processo até que não tenha mais nenhuma transição disparável, provando a propriedade vivacidade. Mas se a rede é viva, após a repetição do processo por um número determinado de vezes encerra-se o programa. Este número foi indicado aleatoriamente, supondo que para este trabalho, os exemplos usados para teste são relativamente pequenos. Nada impede que este número seja aumentado para sistemas maiores.

Mas se o simulador for interrompido após atingir o limite de execução, a lista é percorrida para verificar se todas as transições foram disparadas, caso isso aconteça, a rede é viva.



CAPÍTULO VIII

8 - Estudos de casos em redes de Petri coloridas

Neste capítulo serão analisados alguns exemplos que foram testados usando o simulador desenvolvido. Inicialmente, será analisado o modelo do jantar dos filósofos, sistema clássico na literatura científica computacional. Na sequência será feita a análise do modelo de um sistema de manufatura baseado na filosofia *kanban* usando dois cartões. Os objetivos deste capítulo são testar o simulador desenvolvido e mostrar a aplicabilidade das redes de Petri coloridas em diversas áreas, verificando os resultados obtidos após a simulação.

8.1 - Análise do modelo do sistema do jantar dos filósofos usando redes de Petri colorida

O funcionamento do jantar dos filósofos já foi descrito no capítulo II, mas para que se possa fazer a sua análise, o sistema será novamente apresentado.

Seja $CPN = (\Sigma, P, T, A, N, C, G, E, I)$ representando a rede de Petri Colorida que modela o problema do jantar dos filósofos proposto por Dijkstra. Suponha a existência de cinco filósofos que podem estar comendo ou pensando. Os filósofos estão sentados em volta de uma mesa, cada filósofo tem um prato à sua frente, um garfo à sua direita e outro garfo à sua esquerda, mas seus garfos são compartilhados com os filósofos que estão a seu lado. Assim quando um filósofo estiver comendo, os filósofos que se encontram do seu lado direito e esquerdo não podem estar comendo, mas os outros podem estar pensando ou comendo.

Veja a Figura 8.1 onde os lugares Filósofos, Comendo e Pensando representam os estados dos processos e o lugar Garfos representam os recursos que devem ser compartilhados, ou seja, os garfos dos filósofos. O



disparo da transição t_1 pode ser efetuado segundo as cinco atribuições possíveis às variáveis x e i , ou seja, a expressão de inicialização do lugar Filósofos quando avaliada resulta em cinco marcas de cores distintas, representando os filósofos e o número de iterações. Caso atribua-se à variável x a cor $f_1(t_1, x=f_1)$, a expressão do arco que liga o lugar Garfo à transição t_1 é avaliada para $1'g_1 + 1'g_2$. O disparo de t_1 remove uma marca $1'(f_1, 0)$ do lugar Filósofos e as marcas $1'g_1 + 1'g_2$ do lugar Garfo e adiciona uma marca $1'(f_1, 0)$ no lugar Comendo. Caso a atribuição seja outra, por exemplo $x = f_2$, a expressão avaliada será $1'g_2 + 1'g_3$ e o disparo da transição gera uma outra marcação.

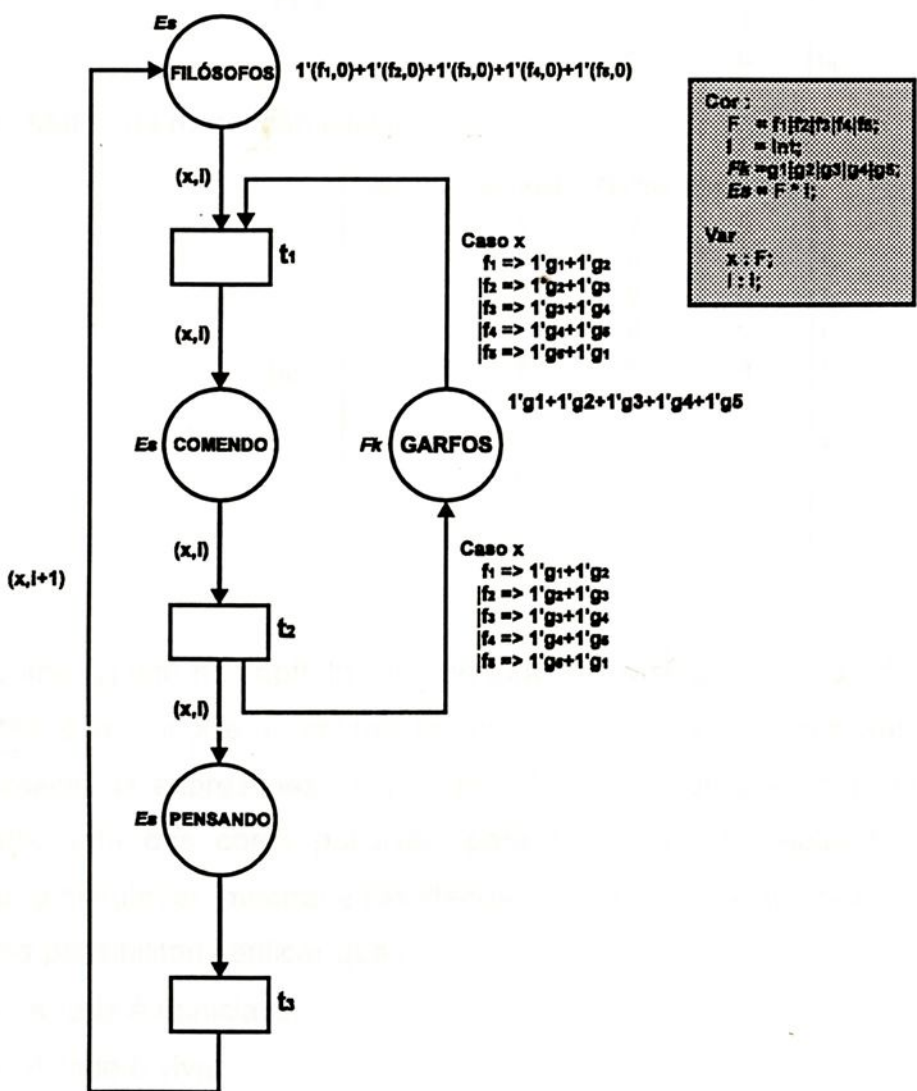


Figura 8.1 - Jantar dos filósofos modelado usando rede de Petri colorida [Mac-96].

Após a explicação do funcionamento da rede representando o jantar do filósofos por uma CP-net, veja como se comportou o simulador para as seguintes informações:

- Matriz de entrada

$$I = \begin{bmatrix} \text{Filosofos} & \text{Comendo} & \text{Garfos} & \text{Pensando} \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{matrix} t_1 \\ t_2 \\ t_3 \end{matrix}$$

- Matriz de saída

$$O = \begin{bmatrix} \text{Filosofos} & \text{Comendo} & \text{Garfos} & \text{Pensando} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{matrix} t_1 \\ t_2 \\ t_3 \end{matrix}$$

- Matriz da marcação inicial

$$\mu_0 = \begin{bmatrix} \text{Filosofos} & \text{Comendo} & \text{Garfos} & \text{Pensando} \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{matrix} t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \\ g_1 \\ g_2 \\ g_3 \\ g_4 \\ g_5 \end{matrix}$$

Como foi dito no capítulo VII, não foram previstas estruturas de decisão aninhadas, como ocorre na expressão arco do lugar Garfo para a transição t_1 . Sendo assim, as expressões arco foram informadas, uma a uma, de acordo com cada uma das cores possíveis para habilitar a transição t_1 . A cada inserção, o simulador, mostrava resultados individuais mas que unidos com os anteriores possibilitou verificar que :

- A rede é reiniciável
- A rede é viva
- A rede é limitada

Com estes resultados conclui-se que o sistema está interagindo de acordo com o esperado, pois para a propriedade vivacidade tem-se que nenhum processo fica esperando infinitamente para usar os recursos oferecidos pelo sistema. A propriedade reiniciabilidade indica que todos os processos estão usando os recursos oferecidos e liberando-os para que outro processo possa usar também, retornando para o estado inicial. Assim, a propriedade limitação indica que nenhum processo usa infinitamente o mesmo recurso.

8.2 - Análise do modelo de um sistema de manufatura baseado na filosofia kanban usando dois cartões

Kanban é o termo japonês que significa *cartão*. Este cartão age como disparador da produção de centros produtivos em estágios anteriores do processo produtivo, coordenando a produção de todos os itens de acordo com a demanda de produtos finais. O sistema *kanban* mais difundido atualmente é o sistema de dois cartões. Este sistema consiste da utilização de dois cartões, um deles denominado ordem de produção *kanban* (*POK*) e o outro *kanban* de transporte (*WLK*).

A ordem de produção *kanban* dispara a produção de um pequeno lote de peças de determinado tipo, em um determinado centro de produção da fábrica. Nenhuma operação de produção é normalmente executada sem que haja uma ordem de produção *kanban* autorizada. Veja na Figura 8.2 um modelo de uma ordem de produção *kanban*.

POK - Ordem de Produção Kanban

Nº da Peça :
Descrição da Peça :
Lote de Fabricação :
Centro de Produção Responsável :
Local de Armazenamento :

Figura 8.2 - Modelo da ordem de produção *kanban* [Cor-93]

O *kanban* de transporte autoriza a movimentação do material pela fábrica, do centro de produção que gera determinado componente para o



centro de produção que necessita deste componente. Nenhuma atividade de movimentação é executada sem que haja um *kanban* de transporte autorizado. Veja um modelo de um *kanban* de transporte na Figura 8.3.

WLK - Kanban de Transporte

Nº da Peça :
Descrição da Peça :
Lote de Fabricação :
Centro de Produção Origem :
Centro de Produção Destino :

Figura 8.3 - Modelo do *kanban* de transporte [Cor-93].

No sistema *kanban* considerado neste trabalho possui uma estação de montagem que é alimentada por uma célula de manufatura. Neste sistema, a célula de manufatura envia um pedido de peça através de uma ordem de produção *kanban* (*POK*). Isto é, a produção é puxada da célula de manufatura por um *cart* cheio (*cart* é uma carreta para o transporte de peças). Quando a célula de montagem usa a peça fornecida pela célula de manufatura, outro cartão, *kanban* de transporte (*WLK*), é enviado com o *cart* vazio para a célula de manufatura [Man-91].

8.2.1 - Componentes do sistema *kanban* de dois cartões

- Duas estações de uma linha de montagem : uma para a peça A outra para a peça B
- Uma célula de manufatura, a qual pega a peça C e produz a peça A ou B
- Um reconhecimento do ponto do estoque onde as chegadas do *cart* C são armazenadas
- Um ponto de estoque de montagem A para armazenar o *cart* cheio A para a célula de montagem A
- Um ponto de estoque de montagem B para armazenar o *cart* cheio B para a célula de montagem B
- Um ponto de estoque de manufatura A para armazenamento do *cart* A cheio ou vazio e do *cart* C cheio ou vazio

- Um ponto de estoque de manufatura B para armazenamento do *cart* A cheio ou vazio [Sch-85].

Veja a Figura 8.4 para entender o Sistema *kanban* de dois cartões.

8.2.2 - Fluxo do sistema *kanban* de dois cartões

1. O início é a célula de montagem B. O *kanban* de transporte é retirado do *cart* B cheio e colocado em uma caixa de coleta. Peças B são removidas, quando necessárias, do *cart* B e usadas na operação de montagem na célula de montagem B;
2. Se um *cart* B é colocado na célula de montagem B, um *kanban* de transporte é enviado e este volta para o ponto de estoque de manufatura B. Se um *cart* B cheio está disponível, o *kanban* de transporte é enviado e o *cart* B cheio volta para o ponto de estoque de montagem B. Antes do *cart* B cheio removido, a ordem de produção *kanban* (*POK*) é removida e colocada em uma caixa de coleta. Se um *cart* B cheio não está disponível, o *kanban* de transporte fica com o *cart* até ser produzido um *cart* cheio;
3. A existência do *POK* na caixa de coleta na célula de manufatura é um sinal para fabricar outro *cart* de peças B. Depois que um *cart* cheio de peças B é fabricado, o *cart* é colocado no ponto de estoque de manufatura B com o *POK* enviado;
4. A manufatura de peças B consiste de peças C como entrada. Antes de alguma peça B for usada, o *kanban* de transporte é retirado do *cart* cheio e colocado na caixa de coleta. Peças C são removidas, quando necessárias, do *cart* e usadas para fabricar uma peça A ou B;
5. Se um *cart* C vazio está disponível, um *kanban* de transporte é enviado e este volta para o ponto de estoque de recebimento. Se um *cart* C cheio está disponível, o *kanban* de transporte é enviado e o *cart* C cheio volta para o ponto de estoque de manufatura A. Se um



cart C cheio não está disponível, o *kanban* de transporte fica com o *cart* até ser recebido uma *cart* cheio;

6. A mesma seqüência de passos é repetida para a célula de montagem A, com a finalidade de se produzir a peça A.

O fluxo do sistema *kanban* apresentado na Figura 8.4 foi resumido para a peça B, pois anotações semelhantes são válidas para a peça A e para a peça C.

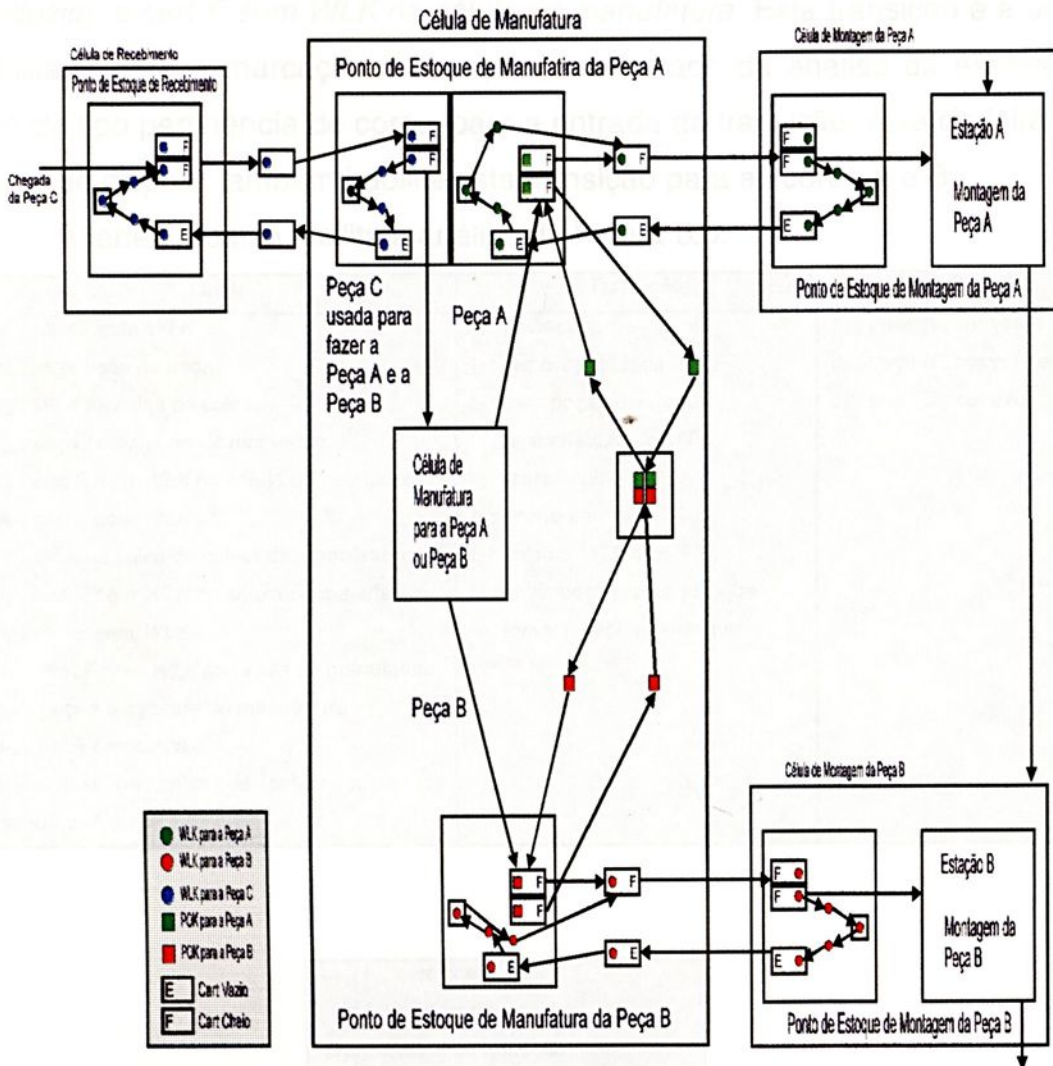


Figura 8.4 - Fluxo do Sistema *kanban* de dois cartões [Sch-85].

8.2.3 - Representação do sistema *kanban* por CP-net

A Figura 8.5 descreve o modelo para o sistema *kanban* de dois cartões usando CP-nets.

Seja $CPN = (\Sigma, P, T, A, N, C, G, E, I)$ representando a rede de Petri Colorida que modela o sistema *kanban* de dois cartões. Inicialmente tem-se marcas nos lugares *WLK na caixa de coleta*, *peça no estoque de montagem* e *peça no estoque de manufatura*, as marcas das cores verde e vermelha representam as peças A e B respectivamente. Quando a transição *usar peça do estoque* é disparada, as marcas no lugar *peça no estoque de montagem* são removidas e marcas do tipo A e B são adicionadas nos lugares *peça produzida* e *cart E sem WLK na célula de manufatura*. Esta transição é a única habilitada para a marcação inicial, pois o resultado da análise da expressão arco do tipo pertinência de cores, para a entrada da transição, é verdadeira e a matriz de disparo também habilita esta transição para as cores A e B.

A tabela abaixo facilita a análise da Figura 8.5.

Lugar	Transição	Cor
p ₁ : <i>Cart F com WLK</i>	t ₁ : <i>pedir peça</i>	c ₁ : <i>peça A : cor verde</i>
p ₂ : <i>requisição de peça</i>	t ₂ : <i>remover WLK de cart F</i>	c ₂ : <i>peça B : cor vermelha</i>
p ₃ : <i>WLK na caixa de coleta</i>	t ₃ : <i>usar peça do estoque</i>	c ₃ : <i>peça C : cor azul</i>
p ₄ : <i>peça no estoque de montagem</i>	t ₄ : <i>anexar WLK a cart F</i>	
p ₅ : <i>cart E sem WLK na célula de montagem</i>	t ₅ : <i>retira WLK</i>	
p ₆ : <i>cart E com WLK</i>	t ₆ : <i>enche cart E</i>	
p ₇ : <i>WLK na caixa de coleta de manufatura</i>	t ₇ : <i>colocar WLK cart F</i>	
p ₈ : <i>cart E sem WLK na célula de manufatura</i>	t ₈ : <i>envia peça para o estoque</i>	
p ₉ : <i>cart F sem WLK</i>	t ₉ : <i>fornece peça ao estoque</i>	
p ₁₀ : <i>cart F com WLK na célula de manufatura</i>		
p ₁₁ : <i>peça no estoque de manufatura</i>		
p ₁₂ : <i>peça produzida</i>		
p ₁₃ : <i>pok na caixa de coleta célula de manufatura</i>		

cart F : cart cheio
cart E : cart vazio
WLK : cartão de remoção
pok : ordem de produção *kanban*

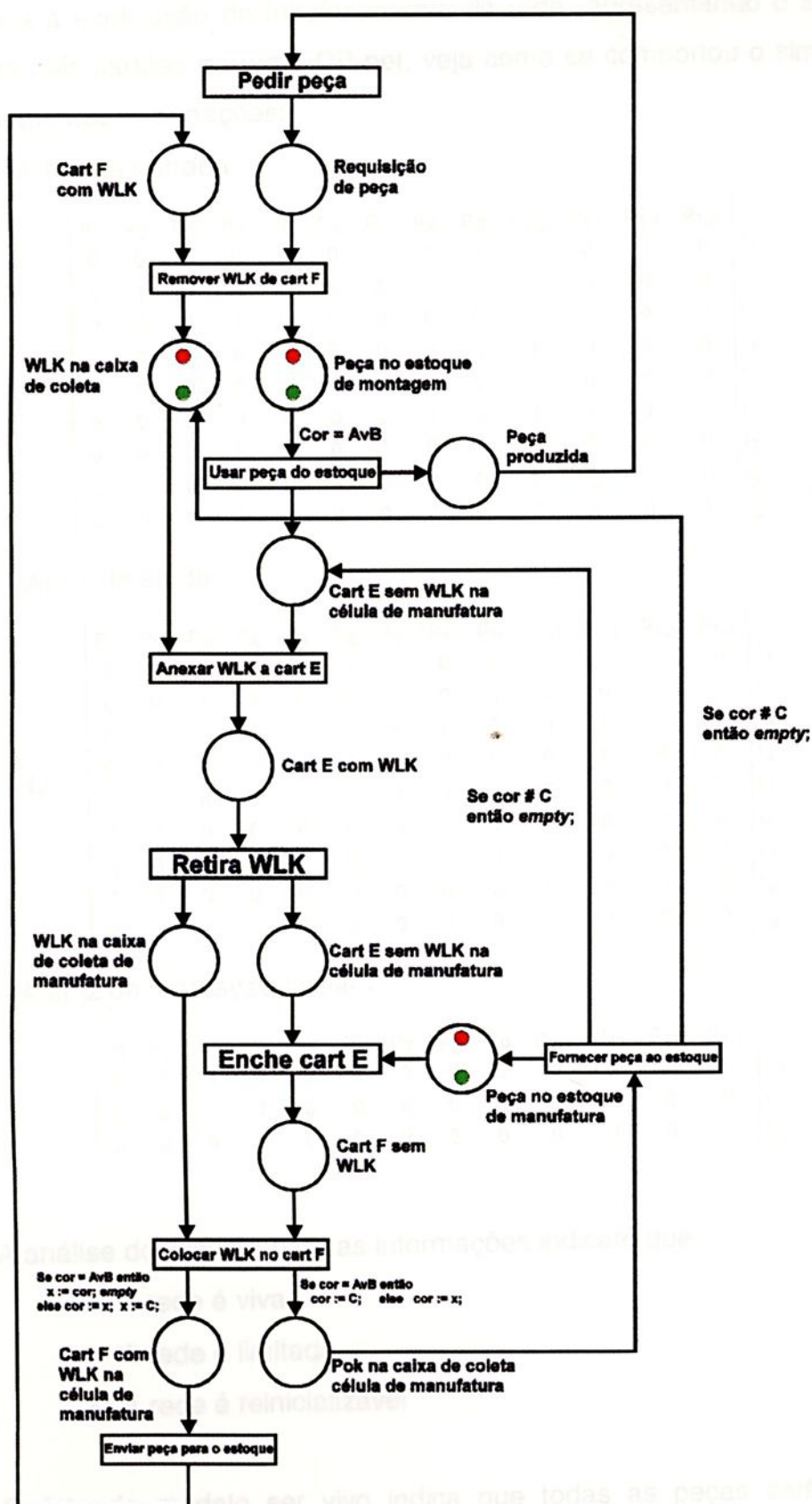


Figura 8.5 - Sistema *kanban* de dois cartões representado por redes de Petri colorida

Após a explicação do funcionamento da rede representando o sistema *kanban* de dois cartões por uma CP-net, veja como se comportou o simulador para as seguintes informações:

- Matriz de entrada

$$I = \begin{bmatrix} p_1 & p_2 & p_3 & p_4 & p_5 & p_6 & p_7 & p_8 & p_9 & p_{10} & p_{11} & p_{12} & p_{13} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{matrix} t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \\ t_6 \\ t_7 \\ t_8 \\ t_9 \end{matrix}$$

- Matriz de saída

$$O = \begin{bmatrix} p_1 & p_2 & p_3 & p_4 & p_5 & p_6 & p_7 & p_8 & p_9 & p_{10} & p_{11} & p_{12} & p_{13} \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{matrix} t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \\ t_6 \\ t_7 \\ t_8 \\ t_9 \end{matrix}$$

- Matriz da marcação inicial

$$\mu_0 = \begin{bmatrix} p_1 & p_2 & p_3 & p_4 & p_5 & p_6 & p_7 & p_8 & p_9 & p_{10} & p_{11} & p_{12} & p_{13} \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{matrix} c_1 \\ c_2 \\ c_3 \end{matrix}$$

A análise do sistema para as informações indicam que :

- A rede é viva
- A rede é limitada
- A rede é reinicializável

O fato do modelo ser vivo indica que todas as peças estão sendo produzidas de acordo com a demanda. A propriedade reiniciabilidade

representa, para o sistema *kanban*, que suas partes estão interagindo e nenhum ponto do sistema está permanentemente inativo. Sendo limitado, o modelo indica que o sistema não está acumulando peças em nenhum ponto de armazenamento, ou seja, não há estoque no sistema. Esta é uma característica desejável dentro da filosofia *kanban*, pois as peças só devem ser produzidas quando existir a necessidade.



CAPÍTULO IX

9 - Conclusão

O desenvolvimento deste trabalho foi baseado no estudo de redes de Petri, destacando as redes de alto nível em especial as redes coloridas.

A teoria das redes de Petri mostrou-se interessante para a simulação de vários sistemas em diversas áreas de estudo. As redes de Petri coloridas permitem representar sistemas complexos por modelos reduzidos, se comparados com as redes Lugar/Transição, pois várias sub-redes idênticas podem ser representadas pela alocação de recursos concentradas em um lugar da rede, como foi mostrado no sistema do jantar dos filósofos.

O simulador desenvolvido faz a análise das redes de Petri coloridas com algumas restrições, ou seja, ainda existem alguns problemas a serem resolvidos. Para as redes com expressões arco dos tipos pertinência de cores e estruturas de decisão simples, ou seja, *if - then - else*, o simulador analisa a rede sem problemas, como foi feito no sistema *kanban*, mas para expressões arco com estruturas de decisão aninhadas, como no exemplo do jantar dos filósofos, deve-se tomar um certo cuidado, pois podem ocorrer situações inesperadas, ou seja, resultados indesejados, como por exemplo se dor informado um *if - then - else* aninhado, sistema não reconhece esta expressão arco e informa que existe um erro na identificação da expressão. Para esta simulação foi necessário inserir cada uma das possíveis expressões individualmente, ou seja, para $x = f_1$, a expressão do arco que liga o lugar Garfo à transição t_1 tem que ser $1'g_1 + 1'g_2$, e assim repete-se para os outros filósofos, um de cada vez, ou seja, a cada simulação. Este problema aparece porque, no decorrer do desenvolvimento deste trabalho, não havia a previsão de analisar estruturas de decisão aninhadas, mas quando foi feita a seleção dos exemplos para avaliar o simulador verificou-se que existia a necessidade deste tipo de expressão.



O simulador foi desenvolvido para ambiente MS-DOS usando a linguagem de programação Pascal, a alocação dinâmica de espaços de memória ajuda a não limitar o sistema para aplicações de tamanhos fixos, ou seja, se fosse usada uma estrutura estática, por exemplo *array*, o simulador ficaria restrito ao tamanho máximo desta estrutura, mas as estruturas dinâmicas são limitadas pela capacidade da memória de trabalho da máquina utilizada, o que permite simular sistemas vários tamanhos.

Diante do problema encontrado, qual seja a impossibilidade de analisar expressões arco com estrutura de decisão aninhadas, uma sugestão é a implementação da análise destas estruturas, possibilitando analisar sistemas com expressões arco mais complexas. Outra sugestão é a implementação deste simulador usando uma linguagem de programação para ambiente *WINDOWS*, por exemplo, a linguagem *Visual Basic* ou então a linguagem *Delphi* pois, tem-se uma interface mais agradável, as expressões podem ser inseridas usando a tabela de símbolos, para o caracteres *ASCII*, do tipo : sinal de pertence, sinal de diferença, etc. Assim como, tem-se condições de melhor aproveitar a capacidade de memória de trabalho (*RAM*) da máquina, não ficando preso aos 640Kb tratados pelo sistema operacional MS-DOS. Também seria interessante a implementação das expressões arco por meio de alguma técnica de compilação.

[Lau-87] Lautenbach, K. - *Linear Algebra*

Notes - Lecture Notes in Computer Science, vol. 44, pp. 1-10, Springer-Verlag, 1987.

Verlag, 1984.

[Lau-87] Lautenbach, K. - *Linear Algebra*

Notes - Lecture Notes in Computer Science, vol. 44, pp. 1-10, Springer-Verlag, 1987.

Verlag, 1984.

[Mac-96] Maciel, P. R. M., Lins, R. D., Cunha, P. R. F. - *Introdução ao Teorema*

de Petri e Aplicações - 10ª Escola de Computação - Unicamp, 1996.



REFERÊNCIAS BIBLIOGRÁFICAS

[Cor-93] Corrêa, H. L., Giansesi, I. G. N. - *Just in time, MRP II e OPT - Um Enfoque Estratégico* - Ed. Atlas S.A., 91 - 99 pp., São Paulo-SP, 1993.

[Des-93] Desel, J. - *P/T-Systems, Invariants, and Free-Choice Systems* - Informatik-Bericht Nr. 26, Humboldt-Universität zu Berlin, 49 pp., 1993.

[Gom-96] Gomes, L. - *Redes de Petri - aplicação a sistemas de automação* - Notas da I Escola Regional de Informática - Unesp - Ibilce, 5 - 43 pp., 1996.

[Jen-81] Jensen, K. - *Coloured Petri Nets and the Invariant-Method* - Theoretical Computer Science, vol. 14, pp.317-336, North-Holland Rep. Co., 1981.

[Jen-87] Jensen, K. - *Coloured Petri Nets* - Lecture Notes in Computer Science, vol. 254, pp. 248-299, Springer-Verlag, 1987.

[Jen-92] Jensen, K. - *"Coloured Petri Nets - Basic Concepts"* - Springer-Verlag, 1992.

[Jen-94] Jensen, K. - *An Introduction to the Theoretical Aspects of Coloured Petri Nets* - Lecture Notes in Computer Science, vol. 803, pp. 230-272, Springer-Verlag, 1994.

[Lau-87] Lautenbach, K. - *Linear Algebraic Techniques for Place/Transition Nets* - Lecture Notes in Computer Science, vol. 254, pp. 142-167, Springer-Verlag, 1987.

[Mac-96] Maciel, P. R. M., Lins, R. D., Cunha, P. R. F. - *Introdução às Redes de Petri e Aplicações* - 10ª Escola de Computação - Unicamp - 1996.



[Man-91] Manacero, A. Jr. - *Computação em Sistemas de Manufatura* - Unesp
- Ibilce - Notas de Aula, 1991.

[Pet-81] Peterson, J. L. - *Petri Nets Theory and the Modeling of Systems* -
Prentice-Hall - 1981.

[Rei-85] Reisig, W. - *Petri Nets - An Introduction* - Springer-Verlag - Berlin
Heidelberg - 1985.

[Sch-85] Schroer, B. J., Black, J. T., Zhang, S. X. - *Just-in-time with Kanban
Manufacturing System Simulation on a Computer Simulation* - 45(2)62-70,
August, 1985.

[Smi-95] Smith, E. - *A Primer on High-Level Petri-Net Theory* - Informatik-
Bericht Nr. 25, Humboldt-Universität zu Berlin, 27 pp., 1995.



