

IMPLEMENTAÇÃO DE REDES DE PETRI EM UMA LINHA DE PRODUÇÃO PARALELA COM ESTAÇÃO OCIOSA

HENRIQUE J. PINHEIRO* GIOVANI B. MERLIN* ALEX TREVISIO*

**Departamento de Sistemas Elétricos de Automação e Energia
Universidade Federal do Rio Grande do Sul*

Av. Osvaldo Aranha, 99 - Centro Histórico, Porto Alegre - RS, CEP: 90035-190

Email: henriquejpo@hotmail.com, giovani.merlin@ufrgs.br, alex.treviso@ufrgs.br

Abstract— This work aims at the use of Petri nets for the modeling and control of a manufacturing plant that implements a production line optimization through the use of the line's idle time for transport of non-priority parts. The discrete event system was implemented and simulated in the graphic tool software TINA.

Keywords— Petri nets, Discrete event system, TINA.

Resumo— Este trabalho visa a utilização de redes de Petri para a modelagem e controle de uma planta de manufatura que implementa uma otimização de linha de produção através do aproveitamento de tempo ocioso da linha para transportes de peças não prioritárias. O sistema a eventos discretos foi implementado e simulado graficamente com a utilização do software TINA.

Palavras-chave— Redes de Petri, Sistema a eventos discretos, TINA

1 Introdução

A planta e o sistema de controle descritos neste estudo serão modelados utilizando-se a teoria de Redes de Petri aplicado a um Sistema a Eventos Discretos (SED). A planta consiste de duas linhas de produção que devem ter os seus produtos escoados através de uma única esteira transportadora. O controlador deve funcionar de forma a permitir o movimento da linha parada quando não houverem peças sendo transportadas na linha em funcionamento ou quando esta não possuir nenhuma peça processada ou com o processo em andamento.

A modelagem e a simulação da rede foram realizadas utilizando o software TINA, respeitando e atendendo os requisitos impostos no presente trabalho quanto as boas propriedades, ou seja, não ser bloqueante, ser reiniciável e limitada.

2 Revisão Bibliográfica

2.1 Funcionamento da Rede de Petri

A rede de Petri é um modelo matemático com representação gráfica que vem sendo amplamente utilizado há mais de 30 anos em vários domínios de atuação entre os quais destacam-se os sistemas de manufatura, de comunicação, de transporte, de informação, logísticos e, de forma geral, todos os sistemas a eventos discretos (CARDOSO e VALETTE, 1997). Dentre as motivações para seu uso encontram-se a possibilidade de analisar o comportamento lógico, avaliar o desempenho e implementar sistemas a eventos discretos.

Três elementos são responsáveis por caracterizar uma rede de Petri, sendo eles:

- **Lugar:** Representado por um círculo. Pode

ser interpretado como um estado parcial, como por exemplo, uma máquina livre, peça em espera, etc;

- **Transição:** Representada por barra ou retângulo. É associada a um evento que ocorre no sistema, como por exemplo, iniciar a operação;
- **Ficha:** Representada por um ponto dentro de um lugar. É um indicador significando que a condição associada ao lugar é verificada. Por exemplo, uma ficha no lugar *máquina livre* indica que a máquina está livre. Se este lugar não possuir fichas, a máquina não está livre.

A Figura 1 representa graficamente uma rede de Petri e ilustra o funcionamento de uma máquina onde, a partir da ocorrência da transição t , as fichas presentes nos lugares *máquina livre* e *peça em espera* são removidas e uma ficha é adicionada ao lugar *máquina em operação*.

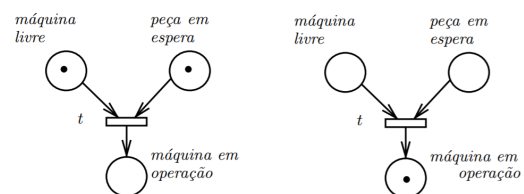


Figura 1: Exemplo de uma Rede de Petri.

Uma transição só pode ser disparada após a mesma estar sensibilizada. Uma transição é dita sensibilizada (ou habilitada) se e somente se cada posição da qual esta retira fichas possuir o número mínimo de fichas necessários para que isto ocorra. Os arcos que interligam os lugares e as transições

indicam a quantidade de fichas que devem ser removidas ou adicionadas, sendo esta quantidade o referido peso do arco. Quando o peso não é indicado, a transição possui peso unitário.

2.2 Propriedades da Rede de Petri

Uma rede de Petri pode ser definida quanto a sua vivacidade, limitabilidade e reiniciabilidade (CARDOSO e VALETTE, 1997).

Uma rede é dita k -limitada se e somente se todos os seus lugares forem k -limitados, ou seja, se todos os lugares que a compõe tiverem uma limitação máxima de k fichas. Caso haja um lugar ao qual o número de fichas possa aumentar indefinidamente, a rede é então denominada ilimitada.

A propriedade de vivacidade está associada as transições t da rede que podem ser classificadas em diferentes níveis, sendo eles:

- L0-viva (ou morta), onde t nunca pode ser disparada;
- L1-viva, onde a partir da marcação inicial, existe uma sequência que pode disparar t ao menos uma vez;
- L2-viva, onde a partir da marcação inicial, t pode ser disparada um número k arbitrário de vezes;
- L3-viva, onde a partir da marcação inicial existe uma sequência em que t pode disparar um número infinito de vezes;
- L4-viva (ou viva), onde a partir de qualquer marcação, existe uma sequência que pode disparar t ;

Vale ressaltar que uma transição que é dita L_j -viva também é L_{j-1} -viva para $j = 2, 3$ e 4 . Também, uma rede de Petri é dita L_j -viva se toda transição t é L_j -viva. Uma rede de Petri dita viva garante que nenhum bloqueio poderá ser causada pela sua estrutura.

Já a propriedade de reiniciabilidade nos diz respeito quanto a possibilidade de que a partir de qualquer marcação acessível seja possível retornar à marcação inicial. Caso isso ocorra, a rede é dita reiniciável. Caso contrário, a rede é dita não reiniciável.

3 Planta a ser Controlada

O sistema a ser tratado foi simulado por meio do software *TINA*, o qual permite a modelagem gráfica e análise de sistemas baseados em redes de Petri. O sistema pode ser visualizado na Figura 2.

Tal sistema consiste de duas linhas de produção. A primeira é representada por um alimentador *Stack Feeder 1* (SF1), uma máquina de processamento *Processing Machine 1* (PM1), uma esteira transportadora *Conveyor Belt* (CB1) e um elemento que transfere peças de uma esteira para outra *Distribution System* (DS). A segunda é repre-

sentada por um alimentador *Stack Feeder 2* (SF2), uma máquina de processamento *Processing Machine 2* (PM2), uma esteira rotatória *Rotary Table* (RB) que funcionado como elemento de transporte entre múltiplas linhas e, por fim, uma rampa de saída do sistema *Exit Slide* (XS). Os componentes citados são descritos na Tabela 1. O funcionamento normal do sistema consiste na movimentação simples da linha de produção inferior, desde o alimentador até a rampa de saída. Neste caso a linha de produção superior não tem por onde escoar a sua produção. A concepção do controle explanado posteriormente visa, justamente, permitir que a produção da linha superior seja movida para fora do sistema pela esteira transportadora da linha inferior somente quando não houverem peças processadas na linha inferior, a qual deve ter prioridade na eliminação de peças do sistema.

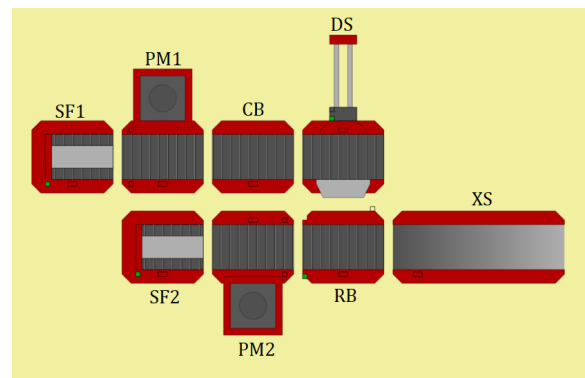


Figura 2: Ilustração do sistema concebido.

Tabela 1: Componentes do sistema e suas funções.

| Componente | Nome | Função |
|-----------------------------|------|---|
| <i>Stack Feeder 1</i> | SF1 | Alimentar a linha de produção superior com uma nova peça para processamento |
| <i>Stack Feeder 2</i> | SF2 | Alimentar a linha de produção inferior com uma nova peça para processamento |
| <i>Processing Machine 1</i> | PM1 | Realizar um processo sobre a peça em questão |
| <i>Processing Machine 2</i> | PM2 | Realizar um processo sobre a peça em questão |
| <i>Conveyor Belt</i> | CB | Transportar peças entre os componentes funcionando como um <i>buffer</i> |
| <i>Distribution System</i> | DS | Mover uma peça da linha de produção superior para a saída |
| <i>Rotary Table</i> | RB | Fazer o direcionamento das peças de ambas as linhas para a rampa de saída |
| <i>Exit Slide</i> | XS | Permitir a saída das peças produzidas nas linhas |

4 Modelagem da planta

A planta analisada é composta por oito elementos modelados com o auxílio do software *TINA*, que possibilita a análise gráfica do sistema. Nesta seção é apresentada a modelagem de cada

componente, bem como suas respectivas representações na rede de Petri implementada e interações gerais que a rede apresenta.

A modelagem desenvolvida e implementada levou em conta aspectos gerais de funcionamento da planta e também questões como limitabilidade, vivacidade e reiniciabilidade, propriedades de interesse devido às especificações de projeto impostas. Pode-se observar que algumas transições relativas à um determinado componente influenciam diretamente o componente vizinho, de modo que a abordagem utilizada para modelar o sistema foi, em um primeiro momento, modelar os componentes individuais para então unir as redes tendo em mente as restrições impostas.

4.1 Stack Feeder 1

Como citado na seção anterior, o *Stack Feeder 1* (SF1) tem como função alimentar a linha de produção superior com peças. A parte da rede de Petri modelada que descreve o funcionamento do *Stack Feeder 1* pode ser visualizada na Figura 3 e a descrição das transições na Tabela 2. A quantidade de fichas na posição Pilha SF1 indica quantas peças estão aguardando na pilha do *Stack Feeder 1*. Após a ocorrência da transição *sf1_wpar* uma ficha é transferida para a posição *peça aguardando*. Assim que o lugar *atuador pronto* também possuir uma ficha e a esteira de PM1 estiver ligada, a transição *sf1_fdon* é sensibilizada, podendo então ocorrer. A transição *sf1_fdon* é responsável por iniciar o ciclo de operação do *Stack Feeder 1*, sendo a peça colocada em movimento e enviada a *Processing Machine 1* através da sequência de transições *sf1_fdon*, *sf1_wplv*, *sf1_fdoff* e *sf1_fdhome*, que indica que o atuador retornou a posição de origem e está pronto para o envio de uma nova peça, momento esse em que uma nova ficha é adicionada ao lugar *atuador pronto*. As fichas retornam para o lugar *Pilha SF1* através da transição de retiradas de peças do *Exit Slide*, fazendo com que nossa rede seja reinicializável e também mantendo a rede limitada.

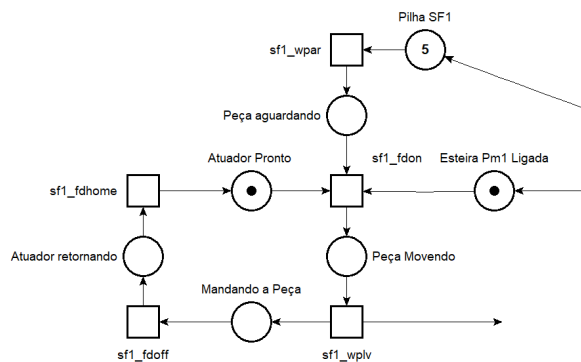


Figura 3: Parte da rede de Petri implementada que descreve o comportamento do *Stack Feeder 1*.

Tabela 2: Transições de SF1 e suas descrições.

| Transição | Descrição |
|------------|--|
| sf1_fdon | Acionamento do motor do <i>Stack Feeder 1</i> |
| sf1_fdoff | Desligamento do motor do <i>Stack Feeder 1</i> |
| sf1_fdhome | Sinal do sensor que indica que o atuador encontra-se na posição de origem |
| sf1_wpar | Sinal do sensor que indica chegada de uma peça |
| sf1_wplv | Sinal do sensor que indica que a peça foi direcionada para a linha de produção |

4.2 Stack Feeder 2

O mesmo processo descrito para o *Stack Feeder 1* (SF1) ocorre para o *Stack Feeder 2* (SF2), apenas alterando as conexões de entrada e saída de fichas do sistema que serão vistas mais a frente na integração de todos os subsistemas. A rede resultante pode ser vista na Figura 4 e as transições na Tabela 3.

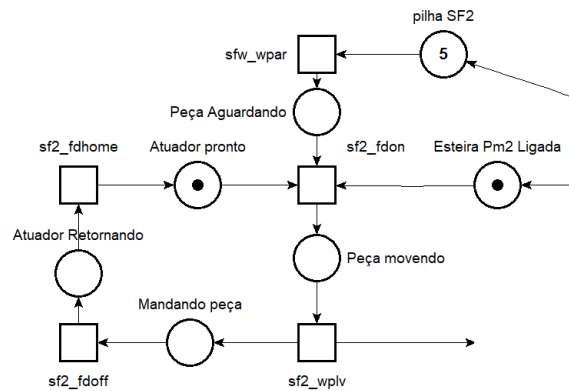


Figura 4: Parte da rede de Petri implementada que descreve o comportamento do *Stack Feeder 2*.

Tabela 3: Transições de SF2 e suas descrições.

| Transição | Descrição |
|------------|--|
| sf2_fdon | Acionamento do motor do <i>Stack Feeder 2</i> |
| sf2_fdoff | Desligamento do motor do <i>Stack Feeder 2</i> |
| sf2_fdhome | Sinal do sensor que indica que o atuador encontra-se na posição de origem |
| sf2_wpar | Sinal do sensor que indica chegada de uma peça |
| sf2_wplv | Sinal do sensor que indica que a peça foi direcionada para a linha de produção |

4.3 Processing Machine 1

A modelagem da parte da rede de Petri que descreve o comportamento da *Processing Machine 1* (PM1) foi desenvolvida visando descrever um ciclo de trabalho completo da mesma, incluindo não só a entrada e saída de peças do sistema mas também todas as operações internas existentes na máquina. A rede que descreve este ciclo é apresentada na Figura 5 e o seu funcionamento está atrelado aos componentes SF1 e CB. Ao receber

a peça de SF1, a transição *pm1_wpar* é disparado e então a esteira PM1 é desligada, para assim iniciar o processo de fabricação da peça. Ao disparar a transição *pm1_ps+*, atinge-se o lugar *Começar Processo* com duas fichas, sendo que, para poder disparar *pm1_poff* é necessário também ficha do lugar *Finalizar Processo*, tal estrutura foi criada para poder executar o compartilhamento da transição *pm1_poff*, pois tal transição ocorre ao iniciar a fabricação da peça e ao terminar a fabricação da mesma. Ao iniciar o ciclo, chega-se ao estado *Começa ou Finaliza* que nesta etapa só pode disparar a transição *pm1_mon*, pois *pm1_bm+* está bloqueado por não existir peça em *Peça Pronta Pm1*, ao seguir o ciclo de processo $p11 \rightarrow p12 \rightarrow p15 \rightarrow p16$ chega-se á transição *pm1_ps-*, que indica a finalização do processo, tal transição libera uma ficha para *Peça Pronta Pm1* e libera duas fichas para *Finalizar Processo*, assim habilitando novamente a transição *pm1_poff*, e retornando *Finalizar Processo* ao seu estado inicial, mas obrigando o lugar *Começa Ou Finaliza* disparar a transição *pm1_bm+*, pois não há mais ficha em *Começar Processo*. Caso a maquina CB esteja ligada, é disparada a transição *pm1_bm+*, trazendo de volta a maquina para o seu estado inicial e mandando uma ficha para *Peça Vindo pra Cb*.

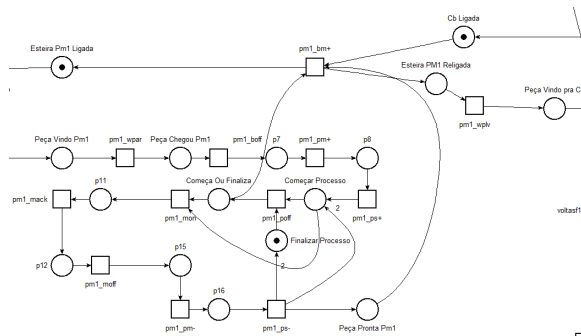


Figura 5: Parte da rede de Petri implementada que descreve o comportamento da *Processing Machine 1*.

4.4 Processing Machine 2

Analogamente a *Processing Machine 1* (PM1), a *Processing Machine 2* (PM2) foi modelada visando representar um ciclo completo de produção. Visto que PM2 encontra-se inversa a PM1, o acionamento da esteira que direciona as peças é feito no sentido contrário, por *pm2_bm-*. A rede que descreve este ciclo de funcionamento é apresentada na Figura 6 e possui como diferença em relação à PM1 a necessidade que CB esteja ligada e que RB esteja na horizontal ligada.

Tabela 4: Transições de PM1 e suas descrições.

| Transição | Descrição |
|-----------|--|
| pm1_bm+ | Liga a esteira da máquina no sentido oeste-leste |
| pm1_boff | Desliga a esteira da máquina |
| pm1_wpar | Sinal do sensor que indica chegada de uma peça |
| pm1_pm+ | Avanço da ferramenta para posição de trabalho |
| pm1_pm- | Recuo da ferramenta para posição de origem |
| pm1_ps- | Sinal do sensor que indica que a ferramenta retornou a posição de origem |
| pm1_mrqu | Sinal do sensor que indica que a máquina está pronta para iniciar o processo |
| pm1_mack | Sinal do sensor que indica que o processo foi finalizado |
| pm1_mon | Acionamento da máquina dando início ao processo |

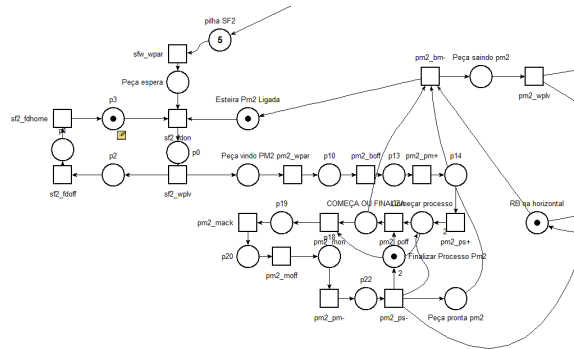


Figura 6: Parte da rede de Petri implementada que descreve o comportamento da *Processing Machine 2*.

Tabela 5: Transições de PM2 e suas descrições.

| Transição | Descrição |
|-----------|--|
| pm2_bm- | Liga a esteira da máquina no sentido leste-oeste |
| pm2_boff | Desliga a esteira da máquina |
| pm2_wpar | Sinal do sensor que indica chegada de uma peça |
| pm2_pm+ | Avanço da ferramenta para posição de trabalho |
| pm2_pm- | Recuo da ferramenta para posição de origem |
| pm2_ps- | Sinal do sensor que indica que a ferramenta retornou a posição de origem |
| pm2_mrqu | Sinal do sensor que indica que a máquina está pronta para iniciar o processo |
| pm2_mack | Sinal do sensor que indica que o processo foi finalizado |
| pm2_mon | Acionamento da máquina dando início ao processo |

Figura 8: Parte da rede de Petri implementada que descreve o comportamento do *Distribution System*.

Tabela 7: Transições de DS e suas descrições.

| Transição | Descrição |
|-----------|--|
| ds_bm+ | Acionamento da esteira no sentido oeste-leste |
| ds_boff | Desligamento da esteira |
| ds_plwpar | Borda de subida do sinal do sensor indicando chegada de peça |
| ds_plm+ | Avanço da ferramenta para deslocamento da peça |
| ds_plm- | Recuo da ferramenta para posição de origem |
| ds_pls+ | Sinal do sensor que indica que a ferramenta encontra-se em posição de avanço |
| ds_pls- | Sinal do sensor que indica que a ferramenta encontra-se na posição de origem |

4.7 Rotary Table

O componente *Rotary Table* (RB) desempenha papel fundamental no funcionamento da planta, sendo o responsável pelo direcionamento de peças de ambas as linhas de produção para a saída XS. O lugar *Peça Aguardando em Ds* é responsável por sensibilizar a transição *rb_rcw*, que depende também do lugar *RB na horizontal*. Seguindo com a transição *rb_scw* e os lugares a ela ligados tem-se a RB posicionada verticalmente e a transição *ds_plm+* pode ocorrer dando prosseguimento ao envio da peça que estava aguardando no *Distribution System*. A transição *ds_ps1* leva a dois lugares:

- *Peça chegando em RB*, lugar compartilhado por ambas as linhas sempre que uma peça chega na *Rotary Table*.
- *Origem vertical*, usado para indicar a necessidade de realizar a rotação da mesa para a posição horizontal através das transições *rt_rccw* e *rtz_sccw* e para que ambas as linhas possam compartilhar a transição *rt_bm+* que liga a esteira de RB.

O processo apresentado anteriormente garante que a *Rotary Table* está na posição horizontal independente da origem da peça (linha superior ou inferior). Portanto, tudo que resta é destinar a peça ao *Exit Slide* através das transições *rt_bm+* e *rt_wplv*.

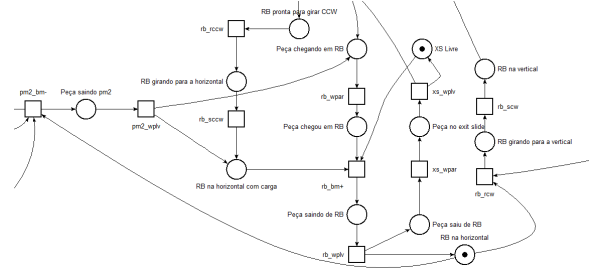


Figura 9: Parte da rede de Petri implementada que descreve o comportamento da *Rotary Table* e do *Exit Slide*.

Tabela 8: Transições de RB e suas descrições.

| Transição | Descrição |
|-----------|--|
| rb_bm+ | Acionamento da esteira no sentido oeste-leste |
| rb_boff | Desligamento da esteira |
| rb_rcw | Acionamento de giro no sentido horário |
| rb_rccw | Acionamento de giro no sentido anti-horário |
| rb_roff | Desligamento do giro da esteira |
| rb_scw | Sinal do sensor que indica que a esteira encontra-se na posição vertical |
| rb_sscw | Sinal do sensor que indica que a esteira encontra-se na posição horizontal |
| rb_wpar | Borda de subida do sinal do sensor indicando chegada de peça |

4.8 Exit Slide

Por fim, o componente *Exit Slide* (XS) permite a saída deslizante do produto finalizado. Possui como transições apenas dois sinais de um mesmo sensor, que detecta através da borda de subida *xs_wpar* a chegada de uma peça e através da borda de descida *xs_wplv* a partida de uma peça. O autômato foi modelado através de dois estados A e B, e suas transições ocorrem de acordo com as transições citadas anteriormente. A implementação de XS pode ser observada na Figura 9 e as transições, bem como suas descrições, na Tabela 9.

Tabela 9: Transições de XS e suas descrições.

| Transição | Descrição |
|-----------|---|
| xs_wpar | Borda de subida do sinal do sensor indicando chegada de peça |
| xs_wplv | Borda de descida do sinal do sensor indicando partida de peça |

Para poder tornar a rede reinicializável foi modelado um sistema de reposição da peça processada na sua pilha de origem, como mostra a figura 10. Ao disparar a transição *xs_wplv*, indicando

saída da peça da planta, é ativada a transição *Voltasf1* ou *Voltasf2*, e sua escolha depende do processamento da planta de cima, indicado por *ds_p1s-* ou da planta de baixo, indicado por *pm2_ps-*.

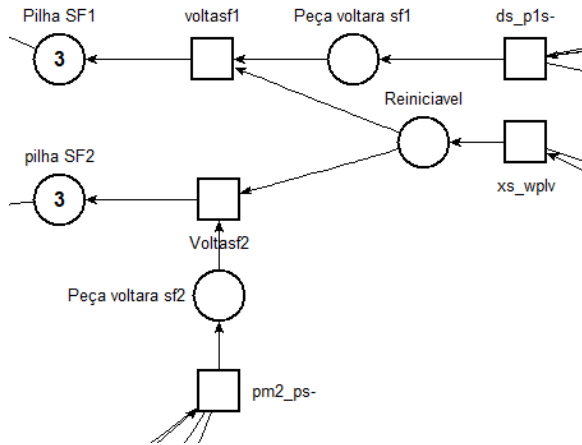


Figura 10: Reinicialização da planta.

5 Análise da rede gerada

Após implementados todos os componentes descritos anteriormente, constatou-se que a planta gerada possui 69 lugares e 56 transições. Sob análise Estrutural, verificou-se que esta é consistente e invariante. Para a análise Conceitual, foram realizadas 6 simulações para condições iniciais diferentes, gerando os resultados apresentados na Tabela 10 para os diferentes valores de fichas nas pilhas SF1 ou SF2.

Tabela 10: Analise Conceitual

| Pilha SF1 | Pilha SF2 | Viva | Marcações | Transições | Transições Mortas |
|-----------|-----------|------|-----------|------------|-------------------|
| 3 | 2 | Sim | 4000203 | 24105948 | 0 |
| 2 | 3 | Sim | 2426606 | 14675872 | 0 |
| 2 | 2 | Sim | 835174 | 4510095 | 0 |
| 1 | 2 | Não | 88430 | 415343 | 2 |
| 2 | 1 | Sim | 155982 | 714285 | 0 |
| 1 | 1 | Não | 13040 | 49411 | 2 |

Não foi possível processar para condições iniciais maiores em cada pilha, pois há um crescimento acelerado de transições e marcações para cada incremento de ficha inicial, requerendo assim um grande esforço computacional. Como pode-se observar, para o caso em que há apenas uma ficha em SF1 a rede se torna morta, pois há duas transições nunca disparadas, *cb_bm+* e *cb_boff*, pois como nesses casos não é possível ter uma peça esperando em CB, a esteira nunca pode ser desligada e então re-ligada. Para todos os casos a rede é reiniciável, limitada (por $\max(PilhaSF1, PilhaSF2, 2)$) e não possui marcações mortas.

6 Conclusão

A modelagem de por Redes de Petri mostrou ser de fácil aplicação, afinal, segue um fluxo lógico semelhante a operação real do sistema que se deseja controlar. Relações de interdependência e travamento, consideravelmente complexas em outros métodos de modelagem, são facilmente implementáveis e o sistema de fichas da Rede de Petri torna claro o funcionamento do sistema em ambiente de simulação. A utilização de transições por mais de um lugar é um dos pontos de maior complexidade encontrado na modelagem da integração dos sistemas pois a sua análise não é tão intuitiva. O software utilizado *TINA* é intuitivo e o modo como implementa a rede é muito semelhante ao que se estuda na teoria, fazendo a transição de teoria para prática um processo rápido. Por fim, o modelo final de rede de Petri implementado resultou em um total de 69 lugares e 56 transições. Também apresentou as propriedades de reiniciabilidade, não ser bloqueante e ser limitado, cumprindo os critérios de projeto estipulados para este estudo.

Referências

- [1] CARDOSO J. VALETTE R. *Redes de Petri*, 1997.