

# MODELAGEM DE SISTEMAS JOBSHOP PARA TOMADA DE DECISÃO EM TEMPO REAL

OCTÁVIO FRANCISCO P. S. R. DOS SANTOS\* , LUIZ E. DE SOUZA\* , GUILHERME S. BASTOS\*

\**Instituto de Engenharia de Sistemas e Tecnologias da Informação, Universidade Federal de Itajubá  
Avenida Bps, 1303  
Itajubá, MG, Brasil*

Email: octaviopaschoal@gmail.com, edival@unifei.edu.br, sousa@unifei.edu.br

**Abstract**— The industrial revolution changed the way that man has treated its production. The development of production processes endures to the present days, when many tools are used in the search for manufacturing lines optimization. Artificial intelligence techniques and decision making methods have been widely used in order to increase productivity and to reduce costs. In addition, processes modeling and simulation are very important to check the response of these systems before its physical implementation. This paper proposes the modeling of flexible manufacturing systems (FMS) with colored and limited Petri nets and decision-making through the MDP (Markov decision processes). After the determination of the policy of actions that will maximize the total reward of a FMS, some examples of this kind of system were simulated and the results for different conceptions of reward and manual production heuristics were checked.

**Keywords**— flexible manufacturing systems, limited colored Petri nets, Markov decision processes.

**Resumo**— A revolução industrial mudou a forma com que o homem passou a tratar sua produção. O desenvolvimento de processos produtivos perdura até os dias de hoje, quando muitas ferramentas são utilizadas na busca de otimização das linhas de manufatura. Técnicas de inteligência artificial e métodos de tomada de decisão vêm sendo utilizados extensamente de forma a se conseguir aumento de produtividade e diminuição de custos. Além disso, modelagem e simulação de processos são muito importantes para verificar a resposta desses sistemas antes da sua implementação física. Este trabalho propõe a modelagem de sistemas flexíveis de manufatura (FMS) por redes de Petri coloridas limitadas e tomadas de decisão através do MDP (processos decisórios de Markov). Após ser determinada uma política de ações que irá maximizar a recompensa total de um FMS, pôde-se simular alguns exemplos deste tipo de sistema e verificar os resultados para diferentes concepções de recompensa e heurísticas de produção manuais.

**Palavras-chave**— sistemas flexíveis de manufatura, redes de Petri coloridas limitadas, processos de decisão markovianos.

## 1 Introdução

A revolução industrial mudou a forma com que o homem passou a tratar sua produção. A invenção de máquinas a vapor e teares mecânicos no século XVIII foi um marco na história, pois estes equipamentos permitiram a implantação das primeiras manufaturas e usinas. Mais tarde, no início do século XX, Ford e Taylor propõem o modelo de linhas de produção, que representam um grande avanço de eficiência no sistema de produção em massa. O *fordismo* prega a economia de tempo mediante aumento da velocidade com que materiais são trabalhados. A produção em massa, no entanto, gera um elevado grau de desperdício com superprodução, estoques e produção de produtos defeituosos.

Este modelo de fabricação teve seu declínio durante a crise do petróleo na década de 1970. Com várias economias em recessão, a empresa automobilística japonesa Toyota se destacou devido a um novo modelo que estava em desenvolvimento desde o término da segunda guerra mundial. O sistema Toyota de produção, ou *toyotismo*, foi concebido para eliminação completa de desperdício, diminuição de custos e produção apenas do necessário. Embasado no modelo *Just-in-Time* e *autonomação*, este modelo entra em cena apro-

ximadamente na mesma época que a automação e robótica ganham destaque em vários setores industriais.

O desenvolvimento da inteligência artificial fez com um novo modelo de linhas de manufatura fosse criado: os sistemas flexíveis de manufatura (FMS). Agora, além da eliminação de desperdícios, um novo agente é estabelecido na produção, um sistema tomador de decisão. A tomada de decisão é responsável por definir o que produzir e quando produzir, de forma a otimizar a produção (Silva and Valette, 1990).

Este trabalho propõe a modelagem de sistemas do tipo *jobshop* por redes de Petri coloridas limitadas (Murata, 1989; Jensen, 1997; Bata et al., 2008) e sua transformação direta em MDP (Bastos, 2010) para tomada de decisão. Esta transformação é a grande contribuição deste trabalho. O *jobshop* é uma classe de sistema de manufatura muito encontrada em arranjos de produção. Após serem calculadas as políticas de ações do MDP, é feita a simulação do sistema no pacote *SimEvents*<sup>®</sup> do software matemático *MatLab*<sup>®</sup>. Os resultados mostram as recompensas totais do sistema para diferentes concepções de recompensas do MDP e heurística manual de produção.

## 2 Sistemas Flexíveis de Manufatura

Um sistema de produção (*workflow*) é formado por máquinas dispostas em determinado arranjo (*layout*), pelas quais os produtos são fabricados seguindo uma rota pré-determinada de produção. Dentre diversas formas de classificação de um sistema de produção, pode-se citar a classificação segundo complexidade, que leva em consideração a sequência de fabricação dos produtos numa determinada linha. As três principais classificações segundo complexidade são (Reis, 2006):

- *openshop* ou ambiente aberto: caracteriza-se pela ausência de ordem para execução das operações;
- *flowshop* ou ambiente contínuo: sistema totalmente sequencial, onde todos os produtos têm a mesma sequência de produção, iniciando e terminando na mesma máquina;
- *jobshop* ou ambiente intermitente: existe sequência de operação para todos os produtos da linha, porém as sequências são diferentes entre as famílias de produtos.

O arranjo do tipo *openshop* dificulta a modelagem do problema, devido à ausência de rotas de produção pré-estabelecidas. O *flowshop* é um problema cíclico, o que pode gerar resultados pouco interessantes para análise das recompensas. Já o *jobshop*, mesmo sendo formado por vários arranjos cíclicos agrupados, permite mais flexibilidade na determinação das rotas de produção. A figura 1 mostra o diagrama de blocos de um sistema tipo *jobshop*. Neste exemplo a fabricação do produto tipo A inicia-se na máquina *M2* e é finalizada em *M3*. O produto C é iniciado na máquina *M1* e finalizado em *M2*. Já o produto B é iniciado em *M1*, processado em *M2* e finalizado em *M3*.

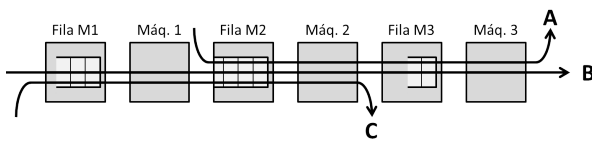


Figura 1: Diagrama de blocos de um sistema *jobshop*

### 2.1 Sistemas produtivos e o FMS

Em sistemas produtivos existe um conflito entre eficiência e flexibilidade. O conceito de linha de produção trouxe eficiência ao processo, devido a economia de tempo mediante aumento da velocidade de produção. No entanto, a produção em larga escala necessita de grandes lotes de matéria prima e grandes *buffers* de armazenamento, ou seja, garante pouca flexibilidade. Já o modelo

tipo *jobshop*, permite que produtos de diferentes famílias sejam produzidos ao mesmo tempo sem a necessidade de grandes estoques e matéria prima. Porém sua eficiência é baixa, visto que máquinas podem ficar ociosas devido a dependência entre as operações. Na tentativa de unir a eficiência das linhas de produção com a flexibilidade do sistema *jobshop* surgem os sistemas flexíveis de manufatura. O FMS é composto basicamente por (Silva and Valette, 1990):

- Conjunto de máquinas flexíveis;
- Sistema automático de transporte;
- Sistema de tomada de decisão.

O foco deste trabalho está no sistema de tomada de decisão e os resultados obtidos para o sistema tipo *jobshop*.

## 3 Redes de Petri

Os sistemas de manufatura muitas vezes são chamados de sistemas de produção discretos (Silva and Valette, 1990), pois trabalham com lotes e unidades de material. Um sistema a eventos discretos (SED) (Ramadge and Wonham, 1989) é um sistema dinâmico cuja mudança de estados ocorre em pontos discretos do tempo. Uma ferramenta que possibilita a modelagem de SEDs é a rede de Petri (RdP) (Murata, 1989), que tem representações gráfica e algébrica intuitivas.

Informalmente, as RdPs são grafos direcionados, bipartidos, compostos basicamente por quatro tipos de elementos: lugares, transições, marcas (*tokens*) e arcos orientados. A figura 2 mostra uma RdP simples e seus elementos.

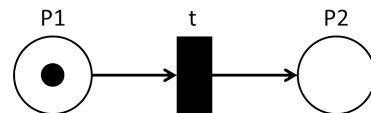


Figura 2: Rede de Petri com representação dos 4 elementos: lugares (círculos), transições (barra), arco orientado (seta) e *token* (ponto)

A definição formal de um RdP é a quintúpla  $R = (P, T, F, W, M_0)$ , onde:

- $P = \{p_1, p_2, \dots, p_n\}$  é um conjunto finito de lugares;
- $T = \{t_1, t_2, \dots, t_m\}$  é um conjunto finito de transições;
- $F \subseteq (P \times T) \cup (T \times P)$  é o conjunto dos arcos orientados;
- $W : F \rightarrow \{1, 2, 3, \dots\}$  é a função peso do arco;
- $M_0 : P \rightarrow \{0, 1, 2, 3, \dots\}$  é a marcação inicial da rede;

- $P \cap T = 0$  e  $P \cup T \neq 0$ .

De um ponto de vista real, os lugares representam os recursos, as transições são os eventos, os arcos orientados definem as relações entre estes elementos e os *tokens* representam a quantidade de recursos disponíveis em cada lugar. Uma marcação é um vetor que relaciona cada lugar da rede com a quantidade de *tokens* em cada um deles.

Os lugares de entrada de uma transição são os lugares pré-condição e os lugares de saída são os lugares pós-condição. A dinâmica do sistema é descrita pelo disparo das transições, retirando a quantidade de *tokens* estabelecida pelo peso dos arcos que ligam seus lugares pré-condição e depositando nos lugares pós-condição a quantidade de *tokens* estabelecida nos arcos que ligam a transição a eles. Uma transição está habilitada para uma marcação  $M$  se a equação 1 for verdadeira:

$$\forall p_i \in I(t_j) \rightarrow M(p_i) \geq \#(p_i, I(t_j)) \quad (1)$$

ou seja, uma transição está habilitada apenas se a marcação dos lugares de entrada for maior ou igual ao peso dos arcos de entrada na transição. A próxima marcação do lugar é dada por:

$$M'(p_i) = M(p_i) - \#(p_i, I(t_j)) + \#(p_i, O(t_j)) \quad (2)$$

que significa que a próxima marcação de um lugar  $p_i$  no disparo de uma transição  $t_j$  é igual à marcação atual menos a quantidade de *tokens* retirada pelo arco que liga  $p_i$  a  $t_j$ , mais a quantidade de *tokens* inserida pelo arco que liga  $t_j$  a  $p_i$ .

O grafo que mostra todas as marcações atingidas por uma RdP é chamada de árvore de alcançabilidade e sua determinação é importante neste trabalho, pois cada marcação da árvore corresponde a um estado na cadeia de *Markov* associada (de Sales, 2002). Isto se faz necessário para a determinação do MDP que irá calcular a política de ações a ser tomadas no FMS.

### 3.1 Redes de Petri Coloridas

As RdPs possuem grande poder de representação de SEDs. No entanto, seus *tokens* são indistinguíveis, representando apenas um tipo de informação. Na representação de uma linha de manufatura isto se torna um problema, pois a modelagem de um sistema se torna muito grande, havendo a necessidade de mais lugares e transições para representação completa do modelo e seus diferentes tipos de produtos.

Uma das contribuições das redes de Petri coloridas (Jensen, 1997) é a possibilidade de agregar informações aos *tokens* de forma que estes sejam individualizados e representem diferentes recursos do mesmo tipo em uma rede. Do ponto de vista gráfico, a RdP colorida continua sendo representada da mesma forma, porém os lugares são classificados de acordo com um conjunto de cores (ou

*colour set*). Esta característica é perfeita para o uso da modelagem de FMS, pois um lugar que representa uma máquina pode conter um conjunto de cores PEÇAS, onde cada *token* representa um tipo diferente de produto.

### 3.2 Redes de Petri Limitadas

Outro aspecto importante a ser considerado na representação por RdP é a capacidade de cada lugar, que pode ser finita ou infinita (Murata, 1989). Não é factível afirmar que a capacidade dos recursos de um FMS seja infinita, portanto é necessário que se limite os lugares da RdP de acordo com suas capacidades.

As máquinas de um FMS geralmente representam um servidor simples, ou seja, processam apenas uma peça de cada vez. Já os *buffers* devem ser caracterizados de acordo com sua capacidade. Para determinação da capacidade dos *buffers* do sistema faz-se necessário o uso das RdPs limitadas, que são definidas pela tupla  $N = (R, k)$  (Bata et al., 2008), onde  $R$  é a definição da RdP e:

$$\exists k \in \mathbb{N} \forall M : M(p) \leq k \quad (3)$$

Isto é importante pois cada estado da RdP deve levar em consideração o que está presente nos *buffers*, e conforme se aumenta a capacidade destes, maior o número de estados atingíveis pelo sistema. Isto tem impacto direto na geração da árvore de alcançabilidade do FMS.

## 4 Processos decisórios de Markov

Com a RdP colorida limitada, é possível que todos os estados do sistema sejam determinados. De forma a tomar a decisão de qual ação executar num determinado estado, foi usada a técnica dos Processos Decisórios de Markov (*Markov Decision Processes* ou MDP).

A definição do MDP é uma quádrupla  $(S, A, T, R)$  (Bastos, 2010), onde:

- $S$  é o conjunto finito de estados do sistema;
- $A$  é o conjunto finito de ações disponíveis ao agente tomador de decisão;
- $T : S \times A \times S \rightarrow [0, 1]$  é a função que dá a probabilidade de o sistema passar para um estado  $s' \in S$ , a partir de um estado  $s \in S$ , se tomada uma ação  $a \in A$ . É denotada por  $T(s'|s, a)$ ;
- $R : S \times A \rightarrow R$  é uma função que dá a recompensa por tomar uma decisão  $a \in A$  quando o processo se encontra em um estado  $s \in S$ .

Os MDPs tem esse nome, pois seguem a propriedade de Markov (Bause and Kritzing, 2002): o efeito de uma ação depende apenas da ação tomada e do estado onde o sistema se encontra.

O agente observador verifica em qual estado o sistema está e a partir disso decide qual ação deve ser tomada. O intervalo de tempo em que as decisões são tomadas é chamado de *época de decisão*  $k$  e o mapeamento das ações para todos os estados do sistema é a política ( $\pi$ ).

Para uma política  $\pi$ , dado que o sistema se encontra em um estado  $s \in S$ , pode-se definir por  $Q^\pi(s, a)$  o valor de uma ação  $a \in A$  no estado  $s$  considerando a recompensa imediata de  $a$  e as recompensas esperadas após sua execução, desde que as ações tomadas após a sejam determinadas pela política  $\pi$ :

$$Q^\pi(s, a) = R(s, a) + \gamma \sum_{s' \in S} T(s'|s, a) V^\pi(s') \quad (4)$$

onde  $R(s, a)$  é a função recompensa,  $T(s'|s, a)$  é a função probabilidade e  $V^\pi(s')$  é a função valor. Para uma política ótima ( $\pi^*$ ),  $Q^*$  é definida por:

$$Q^*(s, a) = R(s, a) + \gamma \sum_{s' \in S} T(s'|s, a) V^*(s') \quad (5)$$

O máximo valor obtido em  $Q^*(s)$  é definido pela função  $V^*(s)$ . Quando a recompensa total esperada pra cada estado é máxima, a política é ótima ( $\pi^*$ ).

$$V^*(s) = \max_{a \in A} [Q^*(s, a)] \quad (6)$$

A política ótima  $\pi^*$  que mapeia as ações que retornam máximo valor de  $Q$  para cada estado é definida por:

$$\pi^*(s) = \arg \max_{a \in A} Q^*(s, a) \quad (7)$$

O fator de desconto  $\gamma \in ]0, 1[$  define a importância das decisões tomadas em épocas futuras. O valor zero não dá importância para recompensas futuras (*greedy behavior*) (Bastos, 2010), enquanto o valor um não gera descontos na recompensa total acumulada. A variação deste valor no cálculo da política vai gerar um dos parâmetros de comparação para o resultado final. Como recompensas para o sistema estudado (FMS), foi considerada uma função de utilidade para cada máquina (Bastos, 2010). Essas funções geram um valor de recompensa de acordo com o tempo de processamento de cada produto nas máquinas, sendo maior seu valor quanto menor o tempo de processamento.

## 5 Modelagem e Simulação do FMS

A modelagem do FMS foi feita a partir de algoritmos no software matemático *MatLab*<sup>®</sup>. Estes algoritmos recebem como parâmetros a notação matricial que representa a RdP do *jobshop* e geram sua árvore de alcançabilidade. Cada transição disparada da RdP corresponde a uma ação

do MDP para a qual pode ser definido ou não um valor de recompensa. A princípio, as transições que recebem recompensa são aquelas que representam a retirada de peças das máquinas. As transições fonte da RdP são responsáveis por inserir novos produtos na linha de produção e as transições dreno responsáveis pela retirada de produto pronto.

Alguns trabalhos relacionando RdP, MDP e FMS serviram de base para este desenvolvimento. Agerwala (1979) e Van der Aalst (1994) introduzem a RdP como ferramenta de modelagem de sistemas industriais. Eboli (2010) apresenta a transformação de RdP coloridas em MDP com probabilidades imprecisas. Já focado em FMS, Silva (2007) aplica a teoria de controle supervisorio (TCS) no sistema de rotas de um FMS, enquanto Molina (2007) propõe o controle de um FMS através da TCS para um FMS modelado por RdP colorida.

Com base no disparo das transições para geração da árvore de alcançabilidade e o valor de recompensa recebido para cada transição disparada, é possível que sejam criadas duas matrizes que são base do cálculo do MDP: a matriz de transições  $P(s, s', a)$  e matriz de recompensas  $R(s, a)$ . Assim é possível calcular a política do MDP a partir de um fator de desconto ( $\gamma$ ). O primeiro resultado deste trabalho é a comparação da recompensa total do sistema para doze valores diferentes de  $\gamma$ .

Num primeiro momento verificou-se que as políticas calculadas previam a ação de inserção de novos produtos na linha para poucos estados do sistema. A análise de alguns estados de interesse mostrou que em muitos casos, mesmo com disponibilidade de máquinas no FMS, o MDP não previa lançamento de novas peças na linha. Isto é devido ao comportamento cíclico do FMS e do horizonte infinito do MDP. De forma a induzir o lançamento de novas peças, foram adicionadas recompensas também para disparo das transições fontes. Com isso tem-se dois parâmetros de comparação: o *jobshop* sem recompensa na fonte e o *jobshop* com recompensa na fonte.

Além das simulações feitas através das políticas calculadas, é possível atribuir heurísticas ao FMS, isto é, uma estratégia manual de produção baseada no conhecimento empírico do sistema. A heurística aplicada a este exemplo é mostrada na tabela 1.

Os resultados deste trabalho foram obtidos para estes três parâmetros através de um modelo de *jobshop* feito no pacote *SimEvents*<sup>®</sup> do *MatLab*<sup>®</sup>.

## 6 Resultados

O FMS escolhido para realização da simulação é um *jobshop* composto de quatro máquinas e três tipos de produtos. A figura 3 mostra o diagrama

Tabela 1: Heurística de produção

Estado	Ação
0 0 0 0 0 0 0	Ação 1 (Insere A)
0 A 0 0 0 0 0	Ação 3 (Insere C)
0 0 0 A 0 C 0 0	Ação 2 (Insere B)
0 0 0 A 0 0 0 C	Ação 2 (Insere B)
0 0 0 0 0 C 0 A	Ação 2 (Insere B)
0 B 0 A 0 0 0 0	Ação 3 (Insere C)
0 B 0 A 0 0 0 C	Ação 3 (Insere C)
0 0 0 B 0 0 0 0	Ação 1 (Insere A)
0 0 0 B 0 A 0 0	Ação 1 (Insere A)
0 0 0 B 0 A 0 C	Ação 1 (Insere A)
0 A 0 B 0 0 0 0	Ação 3 (Insere C)
0 A 0 B 0 0 0 C	Ação 3 (Insere C)
0 B 0 A 0 0 0 0	Ação 3 (Insere C)
0 B 0 A 0 0 0 C	Ação 3 (Insere C)

de blocos do sistema. O produto A tem sequência de produção  $M1 \rightarrow M2 \rightarrow M4$ . O produto B tem sequência  $M1 \rightarrow M3 \rightarrow M2$ . O produto C é produzido por  $M3 \rightarrow M4$ .

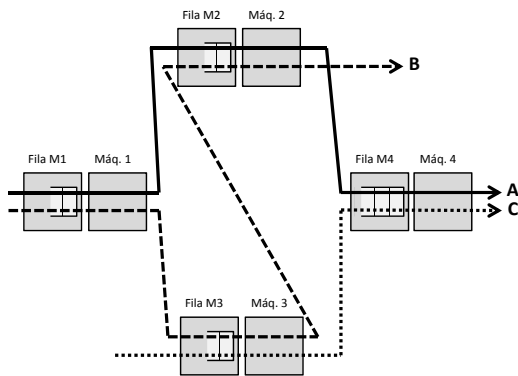


Figura 3: Diagrama de blocos do modelo *jobshop*

A figura 4 mostra a modelagem deste sistema em RdP e a tabela 2 mostra os tempos de produção de cada produto em cada máquina.

Tabela 2: Tempos de produção

Produto	Maq 1	Maq 2	Maq 3	Maq 4
A	20	5	—	10
B	5	10	20	—
C	—	—	15	5

A figura 5 mostra a variação da recompensa total para as três simulações realizadas. Este gráfico mostra que o *jobshop* com recompensa na fonte obteve uma recompensa total maior para todos os valores de  $\gamma$ . Ao se analisar a política, de todos estados possíveis neste sistema, a inserção de novas peças ocorre em cerca de 55% dos estados, enquanto o mesmo ocorre para apenas cerca de 7% do total de estados quando não há recompensa na fonte.

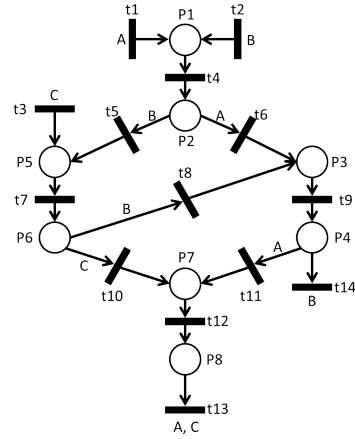


Figura 4: Rede de Petri do modelo *jobshop*

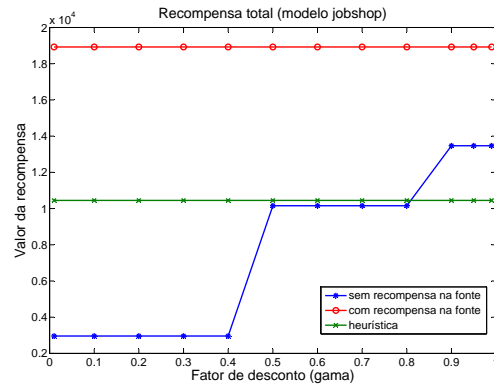


Figura 5: Variação das recompensas totais

O valor da recompensa total da heurística manual não varia pois é calculado apenas uma vez segundo as ações definidas na tabela 1. A figura 6 mostra a quantidade produzida pela heurística e pelos dois MDPs (para  $\gamma = 0.99$ ). A simulação feita para sistema com recompensa na fonte também mostra a produção do dobro da quantidade de peças em relação aos outros dois sistemas. Como a inserção de peças A e C não é concorrente (visto na figura 4), o MDP opta pela produção alternada entre elas, maximizando a recompensa total, ao contrário do sistema sem recompensa na fonte.

## 7 Conclusões e trabalhos futuros

Neste trabalho foi feito um estudo sobre tomadas de decisão em sistemas flexíveis de manufatura. Foi desenvolvida uma metodologia de tratamento do FMS, com modelagem por redes de Petri coloridas limitadas e tomada de decisão através de processos decisórios de *Markov*. Tendo sido calculada a política do MDP, são feitas simulações para três casos distintos: *jobshop* sem recompensa em transições fontes, *jobshop* com recompensa em transições fontes e heurística manual de produção.

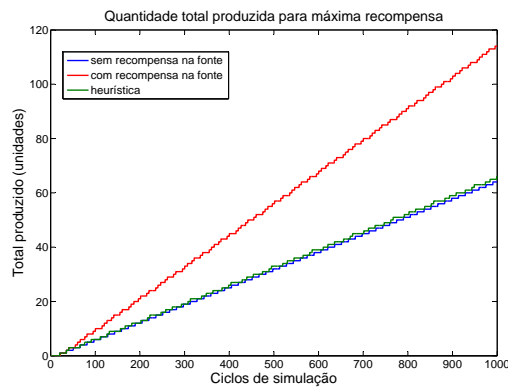


Figura 6: Quantidades totais produzidas

Os resultados mostram que o MDP com recompensa nas fontes apresentou maior recompensa total com relação ao MDP sem recompensa nestas transições e à heurística manual de produção. Este resultado é esperado pois ao se induzir a produção, o MDP mapeia as ações de disparo de peças para todos os estados para os quais estas ações estão habilitadas. Assim existe maior probabilidade do sistema se encontrar num destes estados e conseqüentemente mais peças são introduzidas na linha de manufatura.

Além disso, a recompensa total do MDP sem recompensa na fonte aumenta conforme se aumenta o fator de desconto. Isto é reflexo da recompensa acumulada, que para valores maiores de  $\gamma$ , dá mais importância para decisões tomadas em épocas futuras. A recompensa calculada para a heurística não varia pois não tem influência do valor de  $\gamma$ .

O principal objetivo do trabalho foi atingido, uma vez que as matrizes que definem o MDP são convertidas diretamente da modelagem do FMS em RdP, através da árvore de alcançabilidade.

Espera-se como trabalhos futuros:

- inserção de tempo no modelo não apenas considerando recompensas dependentes, mas através de redes de Petri temporizadas ou TiMDP (Processos Decisórios de Markov dependentes do Tempo);
- modelagem do sistema através da Teoria de Controle Supervisório para comparação entre os métodos de tomadas de decisão;
- repetição da metodologia apresentada em modelos diversos e comparação de resultados.

### Agradecimentos

Os autores agradecem à CAPES e à Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG) pelo apoio e suporte financeiro recebidos para o desenvolvimento deste trabalho.

### Referências

- Agerwala, T. (1979). Special feature: Putting petri nets to work, *Computer* **12**(12): 85–94.
- Bastos, G. S. (2010). *Methods for Truck Dispatching in Open-Pit Mining*, Thesis of doctor in science, Technological Institute of Aeronautics, São José dos Campos.
- Bata, R., Obršálová, I., Volek, J. and Jordão, T. C. (2008). Petri nets application for management of biodegradable components of municipal waste, **4**: 1057–1066.
- Bause, F. and Kritzinger, P. S. (2002). *Stochastic Petri Nets*, Springer.
- de Sales, A. H. C. (2002). Um estudo sobre redes de petri estocásticas generalizadas.
- Eboli, M. G. (2010). *Transformação de redes de petri coloridas em processos de decisão markovianos com probabilidades imprecisas*, Dissertação de mestrado, Universidade de São Paulo.
- Jensen, K. (1997). A brief introduction to coloured petri nets, *Tools and Algorithms for the Construction and Analysis of Systems*, Springer, pp. 203–208.
- Molina, J. R. C. (2007). *Uma abordagem híbrida para o controle de sistemas de manufatura baseada na teoria de controle supervisorio e nas redes de petri coloridas*, Master's thesis, PUC-PR.
- Murata, T. (1989). Petri nets: Properties, analysis and applications, *Proceedings of the IEEE* **77**(4): 541–580.
- Ramadge, P. J. and Wonham, W. M. (1989). The control of discrete event systems, *Proceedings of the IEEE* **77**(1): 81–98.
- Reis, J. (2006). Uma introdução ao scheduling, *REIS*.
- Silva, D. B. (2007). *Aplicação da teoria de controle supervisorio no projeto de controladores para sistemas de rota variável centrado em robô ppgeps*, Master's thesis, PUC-PR.
- Silva, M. and Valette, R. (1990). Petri nets and flexible manufacturing, *Advances in Petri nets 1989*, Springer, pp. 374–417.
- Van der Aalst, W. (1994). Putting high-level petri nets to work in industry, *Computers in Industry* **25**(1): 45–54.